

IBM

Technical Handbook of Systems Engineering



TABLE OF CONTENTS

Section	Area
00 - General	00 Table of Contents, Index, and Index Supplement 01 Introduction to the Technical Handbook 02 Role of the Systems Engineer in IBM
01 - Functions	01 Study and Design 02 Implementation Guidance 03 Operation Support
02 - Study and Design Techniques	01 Display Techniques for Reports 02 Study Documentation Forms 03 Sequential File Run Design 04 Optimum Blocking for Unbuffered Systems
03 - Implementation and Operation Techniques	01 Planning a Data Processing Organization 02 Program Testing Check List 03 Preinstallation Check List 04 Decision Tables 05 Run Documentation Check List
04 - Industry	01 Banking 02 Brokerage 03 State and Local Government 04 Medical Services 05 Chain, Wholesale and Retail 06 Schools and Universities 07 Motor Freight 08 Gas and Electric Utilities 09 Newspaper Publishing 10 Life Insurance 11 Fire and Casualty Insurance 12 Manufacturing Industries 13 Aerospace 14 Service Industries
05 - Application Techniques	01 Statistics 02 Validation and Controls 03 Queuing Theory 04 Matrix Methods 05 Linear Programming

Section	Area	Page
00	00	02

TABLE OF CONTENTS (CONTINUED)

Section	Area
06 - Program Techniques	<ul style="list-style-type: none"> 01 Modular Programming 02 COBOL Reference Description 03 Available FORTRAN Statements 04 Programming Systems Concepts 05 Data Tables and Table Searching 06 Systems Efficiency 07 Fixed and Floating Point Notation
07 - Real-Time Systems	<ul style="list-style-type: none"> 01 IBM Communications Products 02 Glossary 03 Real-Time Systems Design 04 Real-Time Systems Design Criteria 05 Real-Time Systems Requirements Checklist
08 - Tables and References	<ul style="list-style-type: none"> 01 Selected Reference Books 02 Selected Periodicals 03 Selected Professional Societies 04 Number Base Conversion 05 Identification of Mathematical Errors 06 Basic Mathematical Terms 07 Data Representation Codes 08 Boolean Algebra 51 Statistical Tables 52 Mathematical Tables
09 - Equipment	<ul style="list-style-type: none"> 01 Digital Computer Organization 11 IBM Systems Configurators 21 IBM Systems Reference Cards

INDEX

A		Applications development	02:02
		BIBLIOGRAPHY	
ACCOUNTING		Books	08:01
Appropriation	04:03	Periodicals	08:02
Dividend	04:02	BID evaluation problem	05:05
Employee	04:03	BILLING	
Margin	04:02	Hospitals	04:04
Patient billing	04:04	BINARY	
Purchase	04:02	Floating point number	06:07
Revenue	04:03	Number conversion	08:04
Sales Clearance	04:02	Search	06:05
ACCOUNTING controls	01:02	BINOMIAL coefficients	
ACCOUNTS payable		Tables	08:52
Utility industries	04:08	BINOMIAL distribution	
ACTIVITY		Tables	08:51
Analysis problems	05:05	BIOMEDICINE	
Boundaries	01:01	Industry guide	04:04
Formulation	01:01	BIT trigger	09:01
Sheet	02:02	BLENDING problem	05:05
Tape	02:05	BLOCK angular system	05:05
ACTUARIAL calculations		BLOCK diagrams	
Insurance industry	04:10	see FLOW charts	
ADDER	09:01	BLOCKING factor	02:03
ADDRESS	09:01	Unbuffered systems	02:04
Calculation	06:05	BOARD wiring	03:03
Direct	06:05	BOOLEAN algebra	08:08
Intervals	06:05	BOOKS	08:01
ADVERTISING accounting		BROKERAGE	
Newspaper	04:09	Industry guide	04:02
AGENCY statistics		BUDGETS	
Insurance industry	04:10	School industry guide	04:06
AIR CONDITIONING		BY pass procedures	01:02
Requirements	03:03		
ALGEBRA		C	
Boolean	08:08	CABLE requirements	03:03
ALLOCATION problem	05:05	CALLING sequence	06:01
ALTERNATING direction method	05:04	CANONICAL expansion	08:08
ALTERNATING file		CAPACITATED transportation problem	05:05
Run design	02:05	CARD systems	05:02
ALU		Codes	08:07
see ARITHMETIC & logical unit		Comment cards	06:01
ANALYSIS of variance	05:01	Programs	
APPENDIX		Testing	03:02
Study and design	01:01	Reference	09:21
APPLICATION flow charts	01:02	CASH posting	
	03:03	Utilities industries	04:08
APPROPRIATION accounting		CASUALTY insurance	
Government industry	04:03	Industry guide	04:11
ARGUMENT	06:05	CATERERS problem	05:05
Definition	08:06	CENTRAL processing unit	09:01
ARITHMETIC		CENTRAL tendency measures	05:01
Double-precision	08:05	CHAIN stores	
Progression	08:06	Industry guide	04:05
Terminology	08:06	CHANGE tape	
ARITHMETIC and logical unit	09:01	Run design	02:05
ARITHMETIC scale	02:01	CHARACTER codes	08:07
ARRAYS	08:06	CHARACTERISTIC	
ASSEMBLY program	06:04	Floating point	06:07
ASSIGNMENT problem	05:05	CHARTS	
AUDIT trails	01:01	Application	
AVERAGE deviation	05:01	Development	01:02
		Flow	03:03
B		Bar	02:01
BANKING			03:03
Industry guide	04:01	Manpower loading	01:02
BAR graphs	02:01	Organization	03:01
	03:03	Physical plan	03:03

Program		CONTROL matrix	09:01
Flow	02:03	CONTROLS	09:01
Plan	03:03	EAM	03:03
System plan	03:03	Magnetic tape	01:03
Work	03:03	Runs	03:03
CHECK		Tape redundancy	05:02
see also ERRORS		Verification	05:02
Batch	05:02	CONVERGENT iteration	05:02
Code	04:06	CONVERSION procedures	03:03
	05:02	Parallel operation	01:02
Completeness	05:02	Pilot operation	01:02
Consistency	05:02	Planning	01:02
Date	05:02	CONVEX set	05:05
Level	05:02	COPY logic	09:01
Limit	05:02	CORPORATE trust accounting	
Restart procedures	01:02	Banking industry	04:01
Tape	05:02	CORRELATION	05:01
CHI-SQUARE distribution	05:01	Coefficient	05:01
Tables	08:51	COST analysis	03:03
CIRCULATION accounting		Operations	01:01
Newspapers	04:09	Runs	01:01
CITY government		Study and design	01:01
Industry guide	04:03	COUNTY GOVERNMENT	
	04:04	Industry guide	04:03
CIVIL engineering		CURVE fitting	
Government guide	04:03	Polynomial	04:01
CLINICS (Medical)		CURVE graph	02:01
Industry guide	04:04	CUSTOMER accounting	
CLOCKS	09:01	Utilities industries	04:08
CLUB accounting		CUSTOMER inquiries	
Banking industry	04:01	Utilities industries	04:08
COBOL	06:02	CUSTOMER ledger	
CODES	03:03	Brokerage industry	04:02
Character	08:07	CUSTOMER management	
Composite simplex	05:05	Distribution industries	04:05
Primal-dual	05:05	CUSTOMER organization	03:03
COEFFICIENTS			
Binomial	08:52	D	
Cost	05:05		
Multiple correlation	05:01	DATA	
Right-hand side	05:05	Analysis	
COLLATERAL reading		Medical research	04:04
Books	08:01	Codes	08:07
Periodicals	08:02	Control	05:02
COLLEGE industry guide	04:06	Gathering	01:01
COLUMN graphs	02:01		04:06
COMBINATORIAL analysis	05:01	Paths	09:01
COMMENTS cards	06:01	Processing	
COMMERCIAL banks		Functions	03:01
Industry guide	04:01	Organization	03:01
COMMERCIAL loan accounting		Reject	05:02
Banking industry	04:01	DeMORGAN'S theorems	08:08
COMMISSION analysis		DEBUGGING	
Brokerage industry	04:02	Program	06:01
Insurance industry	04:10		03:02
COMMON notation	08:04	DECIMALS	
COMMUNICATION		Floating point number	06:07
Visual		Conversion	08:04
Display techniques	02:01	DECISION tables	03:04
COMPILER program	06:04	DECODERS	09:01
COMPUTERS		DECOMPOSITION algorithm	05:05
see DIGITAL computers		DEFLATION method	05:04
CONFIGURATORS	09:11	DEMAND deposit accounting	
CONSECUTIVE search	06:05	Banking industry	04:01
CONSOLE procedures	03:03	DENSITY function	05:01
	02:03	DENTAL schools	
Messages	05:02	Industry guide	04:04
Operation	01:03	DENTISTS	
Restart	02:03	Industry guide	04:04
Run book	01:03	DESCRIPTIVE statistics	
Switch settings	02:03	Definition	05:01
CONTINUOUS variable	05:01	DESIGN hints	02:05
CONTRIBUTIONS control			
Government industry	04:03		

DESK checking
 Program logic 01:02
 DETAILED narrative 06:01
 DEVICES 09:01
 Teaching 04:06
 DIAGNOSIS 04:04
 Medical industry guide
 DIAGONAL matrix 05:04
 DIET problem 05:05
 DIGITAL computer 09:01
 Organization 06:05
 DIRECT lookup 05:01
 DISCRETE variable 05:01
 DISKS 09:01
 Magnetic 02:03
 Techniques 05:01
 DISPERSION measures 03:03
 DISPLAY techniques 02:01
 Charts 02:01
 Graphs 05:01
 Histograms 05:01
 DISPERSION measures 04:05
 DISTRIBUTION industry 04:05
 Industry guide
 DISTRIBUTIONS 05:01
 Binomial 05:01
 Chi-square 05:01
 F 05:01
 Functions 08:51
 Tables 05:01
 Hypergeometric 05:01
 Multinomial 05:01
 Normal 05:01
 Poisson 05:01
 Students t 05:01
 DIVIDEND accounting 04:02
 Brokerage industry 04:10
 Insurance industry 09:01
 DIVISION 04:04
 DOCTORS 02:02
 Industry guide 02:03
 DOCUMENTATION 02:03
 Forms 01:03
 History log 03:01
 Input 02:03
 Operation 03:02
 Organization chart 02:03
 Output 03:02
 Program test 02:03
 Run 02:03
 Storage map 01:01
 Study and design 02:03
 Switch listing 02:03
 DRIVERS license control 04:03
 Industry guide
 DRUGS 04:04
 Control
 DRUMS 09:01
 Magnetic
 DUAL simplex algorithm 05:05
 DUALITY theorem 05:05

 E

 E
 see EXECUTION
 EAM 03:03
 Control log 03:03
 Desk check
 EDUCATION 04:06
 Industry guide

Section	Area	Page
	00	05
EIGENVALUES		05:04
EIGENVECTORS		05:04
ELECTRIC utility		
Industry guide		04:08
EMPLOYEE accounting		
Government		04:03
EMPLOYMENT security wage benefit		
Industry guide		04:03
ENGINEERING		
Utilities industries		04:08
EQUATIONS		
Normal		05:01
EQUIPMENT accounting		
Trucking industry		04:07
EQUIPMENT selection		01:01
ERLANG		
Distribution		05:03
Process		05:03
ERRORS		
see also CHECK		
Absolute		08:05
Initial		08:05
Percent		08:05
Propagated		08:05
Relative		08:05
Rounding		08:05
Truncation		08:05
ESTIMATING		
Statistics		05:01
Structural data		01:01
EXECUTION		
Controls		09:01
Time		09:01
EXPONENT laws		08:06
EXTREMAL program		05:05
EXTREME point		05:05
F		
F distribution		
Tables		08:51
FACILITIES		
Structure date		01:01
FACTORIAL		08:06
FEASIBLE solutions		
Basic		05:05
Minimum		05:05
Nondegenerate		05:05
FILE		
Maintenance runs		02:05
Study and design		01:01
FILE sheet		02:02
FIRE insurance		
Industry guide		04:11
FIXED point		06:07
FLEXIBLE scheduling		
School applications		04:06
FLIP-FLOPS		09:01
FLOATING point notation		06:07
FLOW charts		
Application		03:03
Logic		02:03
Modular		06:01
Program		02:03
System		06:01
Design		03:03
Implementation		01:02
FOOD industry guide		04:05
FORMAL table		02:01
FORMS		
MO 8 sheets		02:02
I/O		03:03
FORTRAN		06:03
		06:07

Section	Area	Page
---------	------	------

00	00	06
----	----	----

FRACTIONS			Freight	
Conversion	08:04		Motor	04:07
FREIGHT industry guide	04:07		Gas utility	04:08
FREQUENCY			Government	04:03
Analysis			Insurance	
Study and design	01:01		Casualty	04:11
Function	05:01		Fire	04:11
FUNCTIONS			Life	04:10
Definition	08:06		Medical	04:04
Density	05:01		Motor freight	04:07
Frequency	05:01		Newspaper	04:09
Multiple	06:05		Publishing	
Tables	08:51		Newspaper	04:09
G			Retail chains	04:05
			Schools	04:06
			Truck	04:07
			Universities	04:06
GANTT chart	02:01		INFERENTIAL statistics	
GAS utility			Definition	05:01
Industry guide	04:08		IN-PATIENT accounting	
GATES	09:01		Hospitals	04:04
GAUSSIAN elimination	05:04		INPUT-OUTPUT	
	05:05		Control system	06:04
GUASS-SEIDEL method	05:04		Design hint	02:05
GENERAL ledger			Documentation	02:03
Brokerage industry	04:02		Limited programs	06:06
GENERAL preinstallation schedules	01:02		Logic	08:08
GEOMETRIC				09:01
Mean	05:01		Presorted	06:05
Progression	08:06		Records	03:03
GIVEN'S method	05:04		Register	09:01
GOVERNMENT			Study and design	
Industry guide	04:03		Characteristics	01:01
	04:04		INQUIRE response	
GRAPHIC presentation	02:01		Time	03:05
GRAPHS			INSTALLATION	
see also FLOW charts			Cycle	01:02
Bar	02:01		Requirements	03:03
	03:03		INSTALLMENT loan accounting	
Column	02:01		Banking industry	04:01
Curve	02:01		INSURANCE	
Line	02:01		Industry guides	
Mathematical	02:01		Casualty	04:11
Pie	02:01		Fire	04:11
Scales	02:01		Health	04:04
Statistical	02:01		Life	04:10
Surface	02:01		INTEGER programming	05:05
GREEK alphabet	08:52		INTERACTIONS	
H			Statistics	05:01
			INTERNAL floating point format	06:07
HALT listing	01:02		INTERVAL estimation	05:01
HARDWARE industry guide	04:03		INTERVAL scan	06:05
HARMONIC mean	05:01		INVENTORIES	
HEEP	04:03		Control	
HIGHWAY engineering			Design hint	02:05
Industry guide	04:03		Distribution industries	04:05
HISTOGRAM	02:01		Structure data	01:01
HISTORY log	03:05		INVERSE power method	05:04
HOSPITALS			INVESTMENTS	
Industry guide	04:04		Banking industry	04:01
HOUSEHOLDER'S method	05:04		Brokerage industry	04:02
HYPERGEOMETRIC distribution	05:01		Insurance industry	04:10
I			I-O	
			see INPUT-OUTPUT	
IDEALIZED network	05:03		J	
INDEX register	03:05			
INDEX work listing	01:02		JACKSON'S theorem	05:03
INDUSTRIES			JACOBI method	05:04
Banking	04:01		JOB descriptions	
Brokerage	04:02		Consultants	03:01
Chain stores	04:05		Data processing	03:01
Electric utility	04:08		Manager	03:01

Librarians	03:01	Record block	02:05
Specialists	03:01	Redundancy test	05:02
Systems and programming		MAIN line	
Manager	03:01	Programming	06:01
Supervisor	03:01	MANAGEMENT abstract	
JORDAN complete elimination	05:04	Study and design	01:01
JUNIOR colleges		MANTISSA	06:07
Industry guide	04:06	MARGIN accounting	
		Brokerage industry	04:02
K		MASS screening	
		Medicine	04:04
		Psychology	04:04
KRON'S method	05:04	MASTER record	
KURTOSIS measures	05:01	Design hint	02:05
		MATHEMATICAL graphs	02:01
L		MATHEMATICS	
		Number conversion	08:04
LABELS		Symbols	08:52
Tape	01:03	Tables	08:52
	06:04	Terminology	08:06
LATCHES	09:01	MATRICES	
LAW of large numbers	05:01	Algebra	05:04
LEAST squares	05:01	Control	09:01
LEDGERS		Inversion	05:04
Brokerage industry	04:02	Notation	05:05
LIBRARIAN	03:03	MBR	
Job description	03:01	see MEMORY buffer register	
LIBRARY operations	03:03	MEAN	05:01
LIFE insurance		MEDIAN	05:01
Industry guide	04:10	MEDICAL schools	
LINE graph	02:01	Industry guide	04:04
LINEAR constraints	05:05	MEDICINE	
LINEAR programming		Industry guide	04:04
see also MODULAR programming		MEMORY	
PROGRAM		Organization	03:03
Definitions	05:05	MEMORY buffer register	09:01
Simplex method	05:05	MERGING records	06:04
Theory	05:05	MESSAGE sheet	02:02
LINEAR relationships	05:05	MICR	
LOAN accounting		see MAGNETIC ink character recognition	
Banking industry	04:01	MODE	05:01
Insurance industry	04:10	METER reading	
LOGARITHMIC scale	02:01	Utilities industries	04:08
Definition	08:06	MIDPOINT inspection	06:05
LOGIC		MIXED numbers	
Desk checking	01:02	Conversion	08:04
Input-Output	09:01	MODULAR flowcharts	06:01
Operations	01:01	MODULAR programming	06:01
Program	06:01	see also LINEAR programming	
Symbolic	08:08	PROGRAM	
LOOKUP	06:05	MORTGAGE loan accounting	
	06:06	Banking industry	04:01
	09:01	Insurance industry	04:10
LOOP controls	06:06	MOTOR vehicles	
		Freight industry guide	04:07
M		Registration control	04:03
		MULTIPLE correlation coefficient	05:01
MACHINE assignment problem	05:05	MULTIPLE regression	05:01
MACHINE room		MULTIPLICATION	09:01
Design	03:03	Scalar	05:04
Operations	03:03	MUNICIPAL government	
MACHINE-loading problem	05:05	Industry guide	04:03
MAGAZINES	08:02		04:04
MAGNETIC core	09:01	N	
MAGNETIC disks	09:01		
MAGNETIC drums	09:01	NASA	
MAGNETIC ink character recognition		see NATIONAL Aeronautics and Space	
Banking industry	04:01	Administration	
MAGNETIC tape	05:02	NATIONAL Aeronautics and Space	
	09:01	Administration	
Care	01:03	Industry guide	04:04
Control	01:03	NETWORK flow problem	05:05
Labels	05:02		

Section	Area	Page
---------	------	------

00	00	08
----	----	----

NEWSPAPER industry guide	04:09	PERSONNEL	
NOISE record bypass	05:02	Job descriptions	03:01
NON-NEGATIVITY constraints	05:05	Operations	01:03
NORMAL distribution function	05:01	Console	03:03
Tables	08:51	Librarian	03:03
NORMAL equations	05:01	Project director	03:03
NULL matrix	05:04	Tape	03:03
NUMBERS		Requirements	01:01
Base conversion	08:04	Selection	01:01
Distributions	05:03	Structure data	01:01
Large		Training	01:01
Law	05:01	PERT	
NURSES		Implementation usage	01:02
Visiting		PHARMACY control	
Industry guide	04:04	Industry guide	04:04
NURSING homes		PHYSICAL plan	
Industry guide	04:04	Air conditioning	03:03
O		Chart	03:03
		Machine rooms	03:03
		Power requirements	03:03
		Space requirements	03:03
OBJECT deck	03:02	PIE graphs	02:01
OBJECT program	06:04	PILOT operation	01:02
OP DEC		PIVOT element	05:05
see OPERATION decoder		PLANT accounting	
OP REG		Utility industries	04:08
see OPERATION register		POINT estimation	05:01
OPERATION	01:03	POISSON	
Costs	01:01	Arrivals	05:03
Decoder	09:01	Distributions	05:03
Instructions	01:02	Tables	08:51
	02:03	Output	05:03
Personnel	03:03	Service	05:03
Job descriptions	03:01	Service channel	05:03
Register	09:01	POLICY issue	
Sheet	02:02	Insurance industry guide	04:10
Console settings	02:03	POLYNOMIAL curve fitting	05:01
OPERATIONAL analysis		POWER method	05:04
Study and design	01:01	POWER requirements	03:03
OPERATOR messages	05:02	PRECISIONS (numbers)	06:07
OPTIMUM solution	05:05	PRE-INSTALLATION procedure	03:03
ORGANIZATION chart	03:01	PREMIUM accounting	
	03:03	Insurance application	04:10
	05:04		04:11
ORTHOGONAL transformation		PRESENTATION	
OUT-PATIENT accounting		Study and design	01:01
Medical industry	04:04	PRIME numbers	
OUTPUT		Tables	08:52
Documentation	02:03	PRINTING industry guide	
Logic	09:01	Newspapers	04:09
OVERLAY	03:02	PRIORITY processing	09:01
Cards	03:03	PROBABILITY	
Program	02:03	Concepts	05:01
Sheets	03:03	Queuing theory	05:03
P		Statistical	05:01
		PROBLEMS	
		Activity analysis	05:05
PATCH		Allocation	05:05
see OVERLAY		Assignment	05:05
PAYROLL		Bid evaluation	05:05
Schools	04:06	Blending	05:05
Utilities industry	04:08	Caterers	05:05
PEARSON'S coefficients	05:01	Diet	05:05
PATIENT billing		Machine assignment	05:05
Industry guide	04:04	Production scheduling	05:05
PERFORMANCE measures		Smooth production	05:05
Factors	01:01	Tanker routing	05:05
Study and design	01:01	Transshipment	05:05
PERIODICALS	08:02	Transportation	05:05
PERIPHERAL operations	01:03	Warehouse	05:05
PERMUTATIONS		POSITION	06:05
Statistics	05:01	PROCESSING log	05:02
PERSONAL trust accounting		PRODUCT moment correlation	
Banking industry	04:01	Coefficient	05:01

Section	Area	Page
00	00	09

PRODUCTION scheduling problem	05:05	Study and design	01:01
PROFESSIONAL societies	08:03	RESOURCE usage sheet	02:02
PROGRAM plan chart	03:03	RESTART procedures	01:02
	02:03	RETAIL business	
PROGRAM		Industry guide	04:05
see also LINEAR programming		REVENUE accounting	
MODULAR programming		Government industry	04:03
Blocking symbols	03:03	Trucking industry	04:07
Coding	03:03	RINGS	09:01
Control	01:03	ROUND-OFF	05:02
Extremal	05:05	ROUTINES	
Halts	06:01	Program	06:01
Integer	05:05	RUN book	02:03
Modification	06:04	RUNNING time	01:01
Operation	01:03		02:03
Overlays	03:03	RUNS	
Process-limited	06:06	Control	03:03
Selection	06:04	Definition	01:01
Sort	01:03	Development	01:02
Systems	01:01	File maintenance	02:05
Testing	01:02	Limited	02:05
	03:02	Number	03:03
Timing sheets	03:03	Sequential file	02:05
Utility	01:03		
PROGRAMMED learning		S	
Industry guide	04:06		
PROGRAMMING staff	03:03	SALES records	
PROGRAMMING systems	06:04	Brokerage industry	04:02
PROJECT director	03:03	Trucking industry	04:07
PROPOSALS		SAMPLING	01:01
System	01:01	Theory	05:01
PSYCHOLOGICAL testing		SAVINGS accounting	
Industry guide	04:04	Banking industry	04:01
PUBLIC Health Service		SAVINGS and loan associations	
Industry guide	04:04	Industry guide	04:01
PUBLISHING		SCALAR multiplication	05:04
Newspaper industry guide	04:09	SCALE factors	06:07
PURCHASE records		SCANNING search	06:05
Brokerage industry	04:02	SCHEDULES	
Chain store industry	04:05	Nurses	
Schools	04:06	Industry guide	04:04
		Operation	01:03
Q		Pre-installation	01:02
QUEUING theory	05:03		03:03
		SCHOOLS	
R		Industry guide	04:06
		Medical	
RADIX	08:04	Industry guide	04:04
RANDOM processing		Savings programs	
Design hint	02:05	Banking industry	04:01
RANDOM units		SEARCH	06:05
Tables	08:51	SECURITY ledgers	
RANGE	05:01	Brokerage industry	04:02
RATING		SEIDEL method	05:04
Insurance industry guide	04:11	SENSITIVITY analysis	05:05
READOUT	09:01	SEQUENTIAL processing	
RECIPROCALs		Design hint	02:05
Tables	08:52	SERVICE orders	
REDUNDANT calculation	05:02	Utilities industry	04:08
REFERENCE cards	09:21	SIGNIFICANT digits	08:05
REGISTERS	09:01	SIMILARITY transformation	05:04
Voters	04:03	SIMPLEX method	05:05
REGRESSION		Revised	05:05
Multiple	05:01	SIMPLEX tableau	05:05
REGULAR		SIMULATION	
Distribution	05:03	Biological system	04:04
Service	05:03	Study and design use	01:01
REJECTED data	05:02	SIMULTANEOUS equations method	05:04
Absolute	05:02	SKEWNESS measures	05:01
REPORTS		SMOOTH production patterns	
Display techniques	02:01	Problem	05:05
Generators	06:04	SORTING	06:04
		SOURCE language	06:04

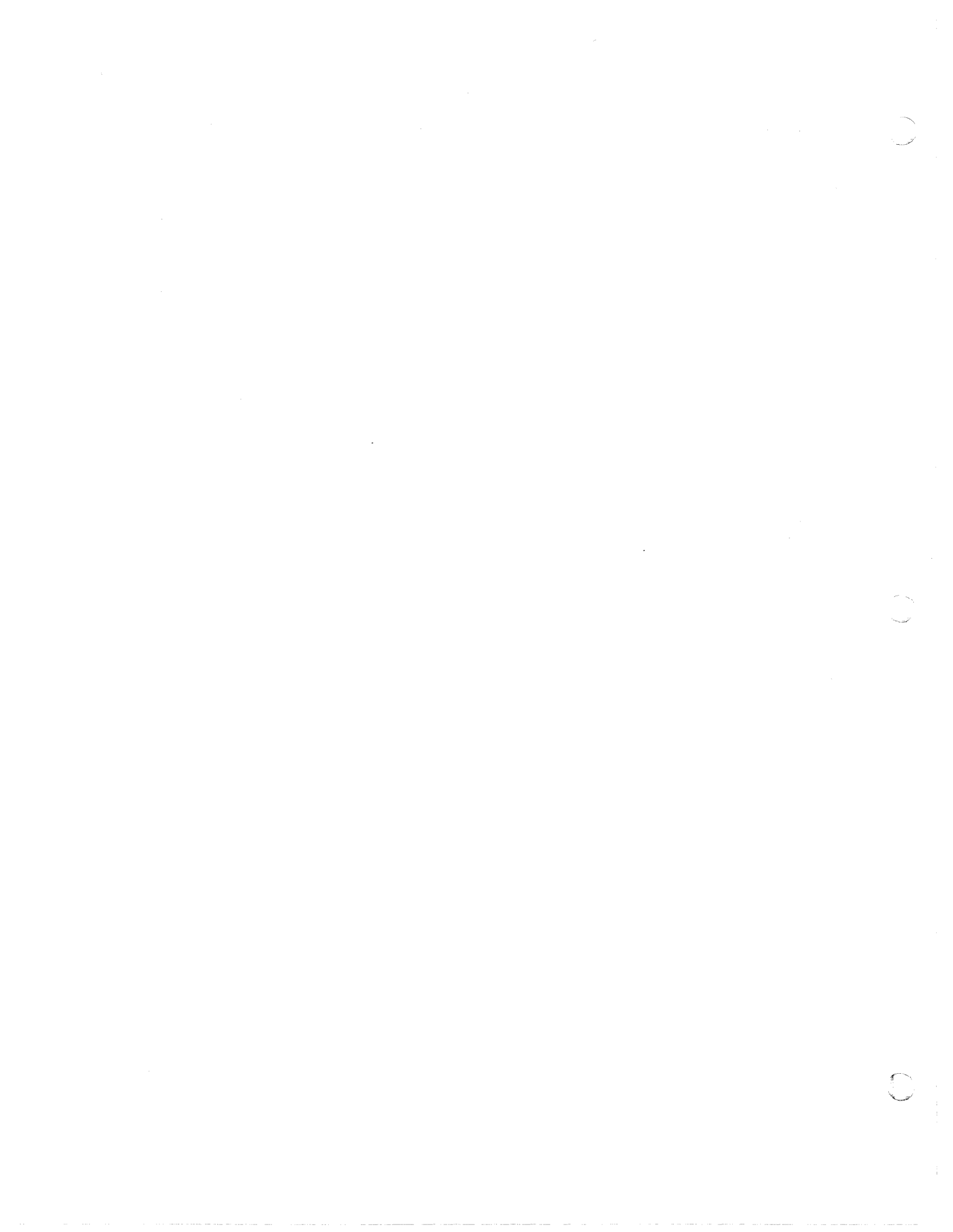
Section	Area	Page
---------	------	------

00	00	10
----	----	----

SPACE requirements	03:03	Time sheets	03:03
SPECIFICATIONS		Work chart	03:03
System design	03:03	SYSTEMS configurators	
SPLIT file		Configurators	09:11
Run design	02:05	Reference cards	09:21
SQUARE matrix	05:04	SYSTEMS engineers	
SQUARE root tables	08:52	Characteristics	00:02
STANDARD deviation	05:01	Professional development	00:02
STANDARDIZATION manual	01:02	Society memberships	08:03
STAR		Reading matter	08:01
see STORAGE address register			08:02
STATE government		Responsibility	
Industry guide	04:03	General	00:02
	04:04	Managerial	00:02
		Promotional	00:02
STATISTICS		Technical	00:02
Definitions	05:01		
Graphs	02:01	T	
Tables	08:51		
	02:01		
STEPPING-STONE method	05:05	TABLE lookup	06:05
STEPWISE multiple regression	05:01		06:06
STOCK brokerage			09:01
Industry guide	04:02	TABLES	
STRAIGHT-LINE relationship	05:05	Argument	06:05
STORAGE		Closed	03:04
Address	09:01	Components	06:05
Register	09:01	Decision	03:04
File		Extended entry	03:04
Blocking factor	02:03	Function	06:05
Description	02:03	Limited entry	03:04
Disk techniques	02:03	Mathematical	08:52
Magnetic core	09:01	Mixed entry	03:04
Magnetic drum	09:01	Open	03:04
Map	02:03	Statistical	02:01
STUDENTS			08:51
Distribution tables	08:51	Structural parts	02:01
Industry guide		TANKER routing problem	05:05
Grade reporting	04:06	TAPE	
Pupil attendance	04:06	Activity	02:05
Student registration	04:06	Availability sheet	01:03
STUDY and design	01:01	Blocking	06:04
SUBSCRIPTS	08:06	Care	01:03
SUBTRACTION	09:01	Change	02:05
SUPERMARKETS		Codes	08:07
Industry guide	04:05	Files	01:03
SUPPLIES system design	03:03		03:03
SURFACE graph	02:01	Header	01:03
SWITCHES		Labels	01:03
Listing	01:02		05:02
	02:03		06:04
Settings	02:03	Library	01:03
SYMBOLIC logic	08:08	Magnetic	09:01
SYMBOLS		Program testing	03:02
Mathematics	08:52	Redundancy tests	05:02
SYSTEM design	01:01	Trailer	01:03
Alternatives	01:01	TAXES	
Blocking symbols	03:03	Industry guide	04:03
Coding	03:03	TEACHING devices	
Cost analysis	03:03	Industry guide	04:06
EAM control	03:03	TESTS	
Efficiency	06:06	Data forms	03:03
File numbering	03:03	Procedures	03:03
Flow charts	03:03	Program	01:02
	06:01	Run book	03:03
Implementation	01:02	Scoring	04:06
I/O records	03:03	THIRD moment measure	05:01
Memory	03:03	TIME analysis	
Plan chart	03:03	Study and design	01:01
Implementation	01:02	TIME distributions	05:03
Runs control	03:03	TIME sheets	
	06:06	Program	03:03
Specifications	03:03	TRAINING	
Supplies	03:03	Personnel	01:02
Tape file	03:03		

Section	Area	Page
00	00	11

TRAINING aids		VALUATION	
Documentation	02:03	Insurance application	04:10
TRANSACTION file		VARIABLES	
Run design	02:05	Boolean	08:08
TRANSFORMATIONS		Definition	08:06
Orthogonal	05:04	Statistics	05:01
Similarity	05:04	VARIANCE	05:01
TRANSIT		VECTORS	05:04
Banking industry	04:01	VEHICLE registration	
TRANSPORTATION problem		Industry guide	04:03
Algorithm	05:05	VERIFICATION	05:02
Capacitated	05:05	VETERANS administration	
TRANSSHIPMENT problem	05:05	Industry guide	04:04
TRIANGULAR matrix	05:04	VISUAL communication	
TRIGGERS	09:01	Formal table	02:01
TRUCKING industry guide	04:07	Graphs	02:01
TRUNCATION	05:02	Histogram	02:01
Error	08:05	VOTERS registration	
TRUST accounting		Industry guide	04:03
Banking industry	04:01	VOLUME analysis	
TRUTH tables	08:08	Study and design	01:01
U		W	
UNBUFFERED systems		WAITING-LINE theory	05:03
Optimum blocking	02:04	WAREHOUSE	
UNIT matrix	05:04	Inventory control	04:05
UNIVERSITY industry guide	04:06	Problem	05:05
UNSCRAMBLING data	05:02	WHOLESALE industry guide	04:05
UTILITIES		WORK chart	03:03
Industry guide		Y	
Electric	04:08	YOUNG method	05:04
Gas	04:08	Z	
UTILITY programs	06:04	ZERO matrix	05:04
UTILIZATION coefficient	05:03		
V			
VALIDATION	05:02		



INDEX, Supplement

A

C (Cont'd)

ACCOUNTING
 Centralized 04:14
 Club 04:14
 Expense 04:14
 Guest 04:14
 Revenue 04:14
 Route 04:14
ACCOUNTS payable
 Manufacturing industries 04:12
 Process industries 04:12
 Service industries 04:14
ACCOUNTS receivable
 Manufacturing industries 04:12
 Service industries 04:14
ACTIONS 07:03
AEROSPACE industries guide 04:13
AGRICULTURAL industries guide 04:12
AIRCRAFT industry guide 04:13
AIRLINES
 Reservations 07:03
AMUSEMENT and recreational services 04:14
ANALYTIC techniques 07:03
APPAREL industry guide 04:12
ARRIVAL times 07:03
ASSEMBLY industries guide 04:12
AUTOMOBILE
 Industry guide 04:12
 Repair and service companies 04:14
AUXILIARY storage 07:03
AVAILABILITY 07:03

B

BANKING
 Real-time system application 07:03
BATCH-type systems 07:03
BILLING
 Manufacturing industries 04:12
 Process industries 04:12
 Service industries 04:14
BOAT building industry guide 04:12
BUFFER features
 1490 07:01
 Use in real-time systems 07:03
BUSINESS associations 04:14
BUSINESS consulting firms 04:14

C

CAPABILITY 07:03
 Checklist for models 07:05
 Parameters 07:04

D

CAPITAL investment
 Manufacturing industries 04:12
 Process industries 04:12
CARD systems
 Transmission terminals
 1013 07:01
CENTRALIZED accounting
 Laundry supply industry 04:14
CHARITABLE organizations 04:14
CHECKLIST
 Real-time systems requirements
 for capability models 07:05
CHEMICAL industries guide 04:12
CITY ledger
 Hotel accounts receivable 04:14
CIVIC organizations 04:14
CLAY industry guide 04:12
CLEANING industries guide 04:12
CLUB accounting 04:14
COMMUNICATIONS
 Control
 Systems
 7740 07:01
 Units
 7710 07:01
 Facilities 07:03
 Glossary 07:02
 Products 07:01
 Systems
 1050 07:01
 1060 07:01
CONSTRUCTION scheduling
 Manufacturing industries 04:12
CONSULTING firms 04:14
CONTROLS
 Communication
 7740 07:01
 Transmission
 1448 07:01
 Programmed
 7750 07:01
COSMETIC industry guide 04:12
CUSTOMER inquiries
 Banking industry 07:03
 "CUT-and-try" procedure 07:03

DATA

Collection systems
 375 07:01
 1030 07:01
 Real-time 07:03

D (Cont'd)

Communication systems	
1050	07:01
1060	07:01
Communication units	
7710	07:01
Control systems	
7740	07:01
Projection	07:03
Reduction	07:03
Transceivers	07:01
DEFINITIONS	
Communications	07:02
DESIGN	
Criteria	07:04
Techniques	07:03
DICTIONARY	
Communications	07:02
DIRECT mail	
Laundry supply industry	04:14
DISK drive channel	
1311	07:01
DRUG industry guide	04:12

E

EDUCATIONAL agencies	04:14
ELECTRICAL industry guide	04:12
ENGINEERING	
Aerospace industries	04:13
Consulting firms	04:14
Manufacturing industries	04:12
Process industries	04:12
EQUIPMENT	07:01
Selection	07:03
EVALUATION techniques	07:03
EXCHANGES	07:03
EXPANDABILITY	07:04
EXPENSE accounting	
Service industries	04:14

F

FABRICATION industries guide	04:12
FILE maintenance	04:14
FOOD	
Control	04:14
Industries guide	04:12
FORECASTING	
Manufacturing industries	04:12
Process industries	04:12
1408 transmission unit	07:01
1414 I/O synchronizer	07:01
1448 transmission control unit	07:01

F (Cont'd)

1490 buffer feature	07:01
FRATERNAL organizations	04:14
FUNCTIONAL environment	07:03
FURNITURE industry guide	04:12

G

GASOLINE blending	
Industry guide	04:12
GENERAL ledger	
Service industries	04:14
GENERAL systems solution	07:03
GLASS industry guide	04:12
GLOSSARY	
Communications	07:02
GUEST accounting	
Hotel industry	04:14

H

HOTEL industry guide	04:14
----------------------	-------

I

IMPLEMENTATION techniques	07:03
INDUSTRIES	
Aerospace	04:13
Assembly	04:12
Fabricating	04:12
Manufacturing	04:12
Service	04:14
INFORMATION systems	07:03
INPUT-OUTPUT	
Synchronizers	
1414	07:01
Unregulated order	07:03
Volume, timing and sequence variations	07:03
INQUIRY	
Systems	07:04
Units	
Remote	
1014	07:01
Stations	
7900	07:01
INSTRUMENTS industry guide	04:12
INVENTORY	
Control	
Fabricating industries	04:12
Laundry industry	04:14
Process industries	04:12
Real-time system	07:03
Trading stamp companies	04:14
ITERATION	07:03

L		O	
LABORATORIES		ON-TRACK betting	04:14
Research and development	04:14		
LABOR unions	04:14	P	
LAUNDRY and linen			
Suppliers industry	04:14	PAINT industry guide	04:12
LEASING companies	04:14	PAPER industry guide	04:12
LEATHER industries guide	04:12	PAPER tape	
LIFE cycle analysis		Readers	
Aerospace industries	04:13	1011	07:01
LINEAR programming	07:03	PARAMETERS	07:03
LINE of balance		Capability	07:04
Aerospace industries	04:13	Reliability	07:04
M		PAYROLL	
MACHINE group queue file		Manufacturing industries	04:12
Aerospace industries	04:13	Process industries	04:12
MACHINERY industries guide	04:12	Service industries	04:14
MAGNETIC tape		PENSION lists	04:14
Transmission terminals		PETROLEUM industry guide	04:12
7701	07:01	PHARMACEUTICAL industry guide	04:12
7702	07:01	PILOT operation	07:03
MAINTENANCE scheduling		PIPELINES	
Manufacturing industries	04:12	Scheduling	04:12
MANAGEMENT advisory firms	04:14	PLASTICS industry guide	04:12
MANAGEMENT Operating Systems		POINT-to-point transmission	07:01
Fabricating industries	04:12	PRECISION assembly industry guide	04:12
MANUFACTURING		PROBABILITY theory	07:03
Industries guide	04:12	PROBLEM definition	07:03
Real-time system application	07:03	PROCESS industries guide	04:12
MATERIALS planning		PROCESSORS	07:03
Manufacturing industries	04:12	PROCESS overlap	07:03
Process industries	04:12	PRODUCTION	
MATHEMATICAL techniques	07:03	Control	04:14
MATRIX algebra	07:03	Scheduling	
MEAT industry guide	04:12	Aerospace industries	04:13
MEMBERSHIP organizations	04:14	Manufacturing industries	04:12
METAL industries guide	04:12	Process industries	04:12
MINING industry guide	04:12	PROFESSIONAL associations	04:14
MISSILE industries guide	04:13	PROGRAM	
MODELING	07:03	Control	
MONTE Carlo technique	07:03	Systems	
N		7750	07:01
NETWORK scheduling		PROGRAMMING	07:03
Aerospace industries	04:13	PROPERTY accounting	
NIGHT audit		Manufacturing industries	04:12
Hotel industry	04:14	Process industries	04:12
NONPROFIT educational and		PURCHASE records	
research agencies	04:14	Manufacturing industries	04:12
		Process industries	04:12

Q

QUALITY assurance
 Manufacturing industries 04:12
 Process industries 04:12
 QUEUING theory 07:03

R

RAILROAD equipment
 Industry guide 04:12
 READERS
 Tape
 Paper
 1011 07:01
 REAL-time systems 07:01
 Design
 Characteristics checklist 07:05
 Criteria 07:04
 Significant factors 07:05
 REFINERIES
 Scheduling 04:12
 RELIABILITY 07:03
 Parameters 07:04
 RELIGIOUS organizations 04:14
 REMOTE inquiry units
 1014 07:01
 RESEARCH organizations 04:14
 RESPONSE speed 07:03
 RESPONSE time 07:04
 RETAILING industry 07:03
 REVENUE accounting
 Service industries 04:14
 ROUTE accounting
 Service industries 04:14
 ROYALTIES
 Petroleum industry guide 04:12
 RUBBER industry guide 04:12

S

SALES
 Analysis 04:14
 Records
 Manufacturing industries 04:12
 Process industries 04:12
 SCHEDULES
 Automobile maintenance 04:14
 SERVICE
 Industry guide 04:14
 SERVICE bureaus 04:14
 SERVICE rate 07:03
 7701 transmission terminal 07:01
 7702 transmission terminal 07:01
 7710 transmission control 07:01
 7740 communication control 07:01

S (Cont'd)

7750 transmission control 07:01
 7900 inquiry station 07:01
 SHIPS
 see BOATS
 SIMULATION 07:03
 65 data transceiver 07:01
 66 data transceiver 07:01
 67 telegraph signal unit 07:01
 68 telephone signal unit 07:01
 SPACE industries guide 04:13
 STABILITY 07:04
 STOCK brokerage
 Inquiries 07:03
 STORE and forward 07:04
 STOVE industry guide 04:12
 STUDY Organization Plan 07:03
 SYNCHRONIZERS
 1414 07:01
 SYNTHESIS 07:03
 SYSTEMS
 Information 07:03
 Inquiry 07:04
 Management operating 04:12
 Real-time 07:03

T

TAPE
 Magnetic
 Transmission terminal
 7701 07:01
 7702 07:01
 Paper
 Readers
 1011 07:01
 TECHNIQUES
 Analytic 07:03
 Design 07:03
 Evaluation 07:03
 Implementation 07:03
 Mathematical 07:03
 TEMPORARY employees lists 04:14
 1001 data transmission system 07:01
 1009 data transmission unit 07:01
 1011 paper tape reader 07:01
 1013 card transmission terminal 07:01
 1014 remote inquiry unit 07:01
 1030 data collection system 07:01
 1050 data communication system 07:01
 1051 terminal unit 07:01
 1060 data communication system 07:01
 1061 control unit 07:01

T (Cont'd)

TERMINAL units	
1051	07:01
Use in real-time systems	07:03
TEXTILES industries guide	04:12
1351 disk drive channel	07:01
357 data collection system	07:01
TIME dependency	07:03
TRADING stamp companies	04:14
TRANSACTIONS	
Processing	07:03
TRANSIT time	07:04
TRANSMISSION control units	
1448	07:01
7750	07:01
TRANSMISSION terminals	
Card	
1013	07:01
Tape	
7701	07:01
7702	07:01

T (Cont'd)

TRANSMISSION units	
1408	07:01
7710	07:01
TRANSPORTATION	
Equipment	
industry guide	04:12
Real-time system application	07:03
	V
VEHICLES	
Industry guide	04:12
VENDOR analysis	
Manufacturing industries	04:12
	W
WAREHOUSE	
Real-time control	07:03
WOOD products industry guide	04:12
WORK cards	04:14



INTRODUCTION TO THE TECHNICAL HANDBOOK

In the design and implementation of information processing systems, solutions range from the introduction of unit record equipment to the use of large-scale data processing machines. In the systems sciences, just as in the physical sciences, a broad range of skills and knowledge is required to perform this work. The body of knowledge relating to a systems engineer's work is drawn from both established and new disciplines, and is growing rapidly to reflect an expanding technology.

The "Technical Handbook of Systems Engineering" synthesizes and distills this large body of systems engineering knowledge into one extensive reference volume, including important facts, accepted concepts and principles, and tested operating practices, methods and techniques.

As is typical of handbooks, information and facts have been compiled from known, tested sources without pushing out to the forefront of existing knowledge. Subject matter is neither comprehensive nor detailed in coverage. Rather it represents a selection and condensation of appropriate material from many different sources and the synthesis of many people's experiences.

The handbook contains more material and information than any one individual is expected to know or utilize at any given time. This is simply a recognition of the variability of a systems engineer's job. He moves from one specialized assignment to another, and in so doing, concentrates on different business functions (accounting, marketing, engineering, etc.), and on different industries (banking, insurance, retailing, etc.). The handbook, then, encompasses the broad scope of subject matter in which a systems engineer may need to demonstrate technical competence.

Each systems engineer is assigned a copy of the handbook for his personal use. The handbook has been prepared and published for IBM personnel only, and is not available outside the Company. It joins the "Systems Engineering Manual" as another principal resource for the practicing systems engineer; the former emphasizes methods and techniques, while the latter stresses administration, policies and organization.

For the experienced systems engineer, the handbook serves as a technical reference from which he can quickly retrieve frequently used information. Many of the necessary facts required in systems work are at his fingertips in this single reference volume.

For the less experienced systems engineer, the handbook acts as an introduction to new topics,

and a guide to additional sources that will provide more detailed information.

Each individual area is placed in an appropriate section based on most common, current usage. Within an area the pages are numbered sequentially. While these area classifications have been carefully made, most effective reference can be made through use of a relatively detailed topic index. The Table of Contents and Index (Section 00, Area 00) will be updated to reflect supplementary distributions.

Systems engineering knowledge is usually categorized by techniques, industry, application, programming, and equipment; handbook sections closely follow this pattern. The contents of the handbook are summarized briefly, section by section, in the following paragraphs:

Section 00, General, structures and introduces the handbook. An index and table of contents are provided, and the role of the systems engineer in IBM is described; the area you are reading covers general information on the effective use and improvement of the handbook.

Section 01, Functions, identifies the tasks a systems engineer performs. The work is described in terms of the application of various systems engineering tools and techniques. The different goals and objectives that are important at several points in time are also emphasized.

Section 02, Study and Design Techniques, contains a more detailed discussion of specific techniques used in studying and designing a business or scientific information system. A number of techniques are described in condensed form as they are appropriate for application in certain pre-sale tasks.

Section 03, Implementation and Operation Techniques, supplies extensive technical information employed in the installation and maintenance of information processing systems.

Section 04, Industry, provides a brief summary of the major characteristics of various industries, along with a discussion of key applications, and a guide to IBM and other literature on these industries.

Section 05, Application Techniques, deals with common problems occurring in many specific applications. The formulation and evaluation of alternative solutions to these common data processing problems is reviewed; important scientific and commercial techniques are covered.

Section 06, Program Techniques, relates to special methods and approaches of programming customer or development problems. The techniques are general in nature, crossing over machines and applications, and emphasize principles and

comparisons among alternatives.

Section 07, Real-Time Systems, describes special techniques, applications and equipment considerations for systems that are real-time in nature, including TELE-PROCESSING[®], control and other online, fast-response operational systems.

Section 08, Tables and References, includes various standard reference material, list of periodicals, books and societies, and descriptions of general mathematical tools and techniques of use to the systems engineer in designing or implementing information processing systems. General mathematical and other tables have not been inserted unless of special interest in performing systems work.

Section 09, Equipment, provides selective descriptions of major classes of equipment, together with appropriate reference material summarizing equipment characteristics.

Portions of the Technical Handbook of Systems Engineering have been contributed by many different individuals engaged in systems engineering and related work in a large number of IBM locations and divisions. All possess a high level of skill and experience in their specialties. Systems Engineering is indebted to all of them for their support in establishing this fundamental exchange of systems engineering knowledge.

A handbook is a dynamic project, continuously under review to refine and expand its content and

coverage. Future improvement depends largely on the extent to which systems engineers throughout IBM participate in its maintenance and modification. This contribution may be made in several ways. The reader should submit recommended modifications and corrections to the present material, or he may suggest inclusion of new subject areas. If, in addition to recommending a new area, the systems engineer wishes to follow through in its preparation, he should request permission from his manager and approval from the editor of the handbook. All correspondence concerning the handbook should be directed to:

Editor, Technical Handbook of
Systems Engineering

Data Processing Division Hdqtrs.

112 East Post Road

White Plains, N. Y.

Since all technical sources need to be updated periodically, formal arrangements have been set up to supplement the handbook as new subject areas mature and as established areas are appropriately defined. To keep his copy current and accurate, the systems engineer should insert these new areas or changes into the handbook as they arrive.

ROLE OF THE SYSTEMS ENGINEER IN IBM

A systems engineer is a highly skilled technologist responsible for the design and implementation of information processing systems. Although the need for individuals expert in business systems can be traced to the founding of IBM, the current urgency is directly related to the development and rapid growth of the electronic computer during the last decade. In view of the recent appearance of thousands of new systems engineers, it is appropriate to establish their identity in terms of their work, outline the required body of knowledge, and discuss their relationships to IBM and to the customer.

BACKGROUND

Until the early 1950s, company effort was concentrated on the marketing and installation of unit record equipment. Although widely accepted, the equipment was limited to fairly standardized applications. Salesmen were able to survey, sell and install the equipment with the assistance of systems service representatives. Systems service representatives had the dual function of (1) educating customer and IBM personnel in the wiring of machine control panels and in the application of unit record equipment to business problems, and (2) providing assistance to the salesman in installing the unit record equipment in a customer office.

The introduction in 1953 of the first production line models of large IBM electronic data processing machines opened up many new and many expanded application areas. The rapid acceptance of the equipment and the scope of its applications ushered in a host of new challenges. A specialized systems service group, called field technical representatives, was formed to assist in the implementation of equipment applications. This group was composed mainly of former customer engineers who had received extensive training on the newer equipment. As the product line expanded and data processing techniques became more widely accepted and applied to increasingly diversified areas, it became increasingly difficult for a salesman to be completely knowledgeable in all the needed disciplines. Sales assistants and systems representatives received specialized training either in unit record equipment or in commercial or scientific use of electronic computers. As the use of electronic computers began to assume a more vital role in industry, a new breadth of understanding and knowledge was required far beyond the needs of the past. A new profession, in fact, was required to

provide technical guidance in the use of IBM equipment, in systems and application planning, and in programming and testing. Therefore, in 1960 the systems engineering function was organized within the Data Processing Division.

CHARACTERISTICS OF THE SYSTEMS ENGINEER

In the design and implementation of information processing systems,* the systems engineer is concerned basically with the application of equipment to the solution of business problems. To develop business systems capable of meeting complex demands of the future, a systems engineer should combine technical competence in many subjects with a fresh, creative approach to problem solution. In this respect, he is not unlike the engineer designing physical products, or the architect designing a new home. While many systems engineers are involved in work similar to that done with unit record equipment in the past, more and more, the experienced and skilled engineer is expected to be capable of solving larger-scale problems in a wider range of work. "The Role of the Systems Engineer in IBM" is written to emphasize his future role as a professional problem-solver of such calibre.

A systems engineer should be professional in attitude, standards and competence. In all of his dealings with a customer, he maintains IBM's traditional high standards of approach, behavior and ethics. He sets high personal performance standards, and accepts full responsibility for the quality and effectiveness of his work. He requires knowledge from many disciplines (mathematics, information theory, accounting, etc.), plus the general tools for formulating and solving systems problems. Specific know-how in equipment, programming, industries and applications rounds out the professional characteristics of the systems engineer.

A systems engineer spends much of his time in planning work. As a planner, he views the business he is working with over extended time periods from a broad, generalized frame of reference. Goals and objectives of the business, as well as the way a business reacts to its environment, are paramount considerations to him. He must become thoroughly familiar with all facets of a business, and the logic behind the information system must be

*An information processing system is a combination of men and machines operating under defined procedures to achieve some stated goal or purpose.

Section	Area	Page
00	02	02

as clear to him as it is to the managers and supervisors of the system. His recommendations may have far-reaching impact; in making them he must balance the expected results and savings against the cost of obtaining them.

Thus the systems engineer cannot isolate himself from operations in the manner of the basic research scientist. He must exercise leadership and coordination during the implementation period to insure maximum realization of his plans.

Upon completion of his training program, a systems engineer generally performs such duties as making surveys and developing solutions to customers' problems. He may assist in the preparation and presentation of proposals and development and installation plans. These assignments will normally cover a number of different customers and prospective customers, and a variety of industries and applications. As a systems engineer gains experience and knowledge, he may continue in this general vein, or he may become a specialist in particular areas to make the best use of his knowledge, training, experience and ability.

Whether he continues as a generalist or becomes a specialist, and whether he is assigned to an application development project, a continuing association with one or more customer accounts, or the design of a large-scale business system, the systems engineer finds great diversification in his work. He may be called on to:

- Prepare a project plan for conducting a systems study.
- Report progress on a current study to customer management.
- Coordinate the efforts of a multiple-member study team.
- Diagnose problems in a specific area of the business.
- Provide guidance in programming and testing applications in a newly designed system.
- Advise on the organization of a new data processing installation.
- Train customer personnel in the use of equipment or the handling of a new routine.
- Discuss technical problems with management consultants.
- Demonstrate the applicability of a utility program or general-purpose simulator.
- Assist in the preparation of a systems proposal.
- Conduct a demonstration of a newly designed system.
- Coordinate program testing at a datacenter.

To perform effectively in this wide range of tasks, a systems engineer must be able to:

- Apply knowledge from many specialized fields and from the current literature.
- Analyze the underlying causes of problems quickly and accurately.
- Resolve varied problems with adaptable basic techniques.
- Communicate his ideas and maintain harmony among the personnel involved.
- Determine when to consult other specialists, such as special industry or installation planning representatives, communication equipment specialists, etc.

Some systems engineers discover that certain fields of endeavor or types of work hold considerably more appeal than others. For example, a systems engineer with a strong mathematical background may wish to devote much of his time and effort to the application of statistical and numerical techniques for solving scientific data processing problems, or to designing mathematical models. On the other hand, a systems engineer with an accounting background may want to direct his talent toward the design of special control and evaluation techniques for data processing systems.

There are many other dimensions and opportunities in specialization, depending on the interest and motivation of the individual -- for example:

- Development of general-purpose, diagnostic and control programs.
- Design of programming languages.
- Emphasis by type of application: commercial, scientific, real-time or process control.
- Education and training of personnel.

The systems engineer carries a large share of the responsibility for training customer personnel as part of his daily routine. This training varies from the formal to the informal: orientation programs at the beginning of a study, classroom instruction at an education center, brief sessions for new operators on the operation of a program, or depth training on special topics such as FORTRAN, use of decision tables, etc. The need for training customer and company personnel is a continuing one; it presents opportunities for both the generalist and the specialist.

IBM provides many paths of growth for a systems engineer, both as a generalist and as a specialist, in systems engineering itself as well as in related areas such as sales, marketing and product

Section	Area	Page
00	02	03

development. All of these paths can lead to technical and managerial positions at the branch, district, region, headquarters, product development laboratory, and manufacturing facilities of IBM.

SPAN OF RESPONSIBILITY

Throughout the discussion of the characteristics of a systems engineer, reference has been made to the types of work he performs. Under each of the following responsibility headings, the job of the systems engineer is developed in broad outline and associated with the several environments in which he will find himself as he is assigned to various customers or projects. The job description, as developed here, characterizes more the higher-level performance expected of advanced systems engineers.

Major Task Responsibilities

Over the course of his career a systems engineer will be involved in a number of major tasks. They are listed here in the order he faces them: from the presentation of a problem to its resolution through a fully operating system. Each task is discussed at some length in Section 01 of the handbook.

1. Examination of a presently operating business (in whole or in part) to gain insight and understanding of the principal relationships, structure, objectives and methods, and to identify and define problem areas.
2. Determination of what the system is required to do in terms of inputs, operations, outputs and resources, and how well the system (or application) must perform.
3. Formulation of design alternatives as a solution to the problem, independent of equipment configuration.
4. Selection of unit record or computer equipment configurations for each of the design approaches, and refinement into complete systems solutions.
5. Evaluation of the system in terms of savings and costs over future time periods.
6. Description of the systems design at a general level and communication of the solution to the customer in the form of a proposal.
7. Implementation of the system, including detailed system design, programming, training of personnel, physical installation preparation, conversion and systems test.
8. Improvement of systems performance during operation, and follow-through to resolve special problems.

Business Environment Responsibilities

The systems engineer works in a complex environment. Not only must he quickly accommodate himself to the special surroundings he encounters today, but he must also be flexible enough to move on to new situations tomorrow and make an equivalent adjustment to different problems and circumstances.

He may be associated with one or more relatively small, single-location concerns, or, at the opposite extreme, he may be placed in a large corporation with branch plants at many remote locations. His main responsibility might be the design of special applications (e. g., accounts payable and receivable, invoicing, payroll, inventory records) for unit record equipment, or he could well be the leader (or member) of a large-scale systems study (e. g., design of a nationwide finished stock control system, order entry system, or real-time reservation system) which may result in an order for several IBM 1401s, 1410s, or a 7094.

Variability of environment applies also to industry. Systems engineers are used in every industry: banking, manufacturing, insurance, wholesale distribution, retailing, construction, public utilities, government, and so on. The unique features, methods, practices and terminology of each industry have to be studied and understood before the systems engineer can perform effectively. Beyond this, he must become conversant with the special conventions in functional areas of the business. For example, a manufacturing concern normally has marketing, engineering, manufacturing, accounting and personnel departments. The systems engineer may be called on to automate the engineering design activity, and therefore need to acquire knowledge of design practices, or he could be part of a management information control system study and be concerned, to a degree, with all functions of the business.

Technical Responsibilities

To obtain maximum performance in such varied environments, a systems engineer needs to be technically competent in a number of disciplines. He should have a basic understanding of engineering and mathematical techniques. In addition, he should have some educational background in both science (physics, chemistry, biology) and business administration (economics, accounting, marketing). A systems engineer requires an unusually high level of analytical ability to work with scientific

methods and the creative ability to develop new business systems. He should be able to conduct numerical and statistical analyses, and apply management science tools such as the construction of mathematical models, linear and dynamic programming, and the development of planning and scheduling networks. The suggested educational background will provide him with the basis for developing these abilities.

At the same time, the professional systems engineer must have an extensive knowledge of unit record and computer equipment, as well as related programming systems (including standard application programs). In addition, he must have knowledge of industry practices, specialized applications (such as interest calculations in a bank), and standard methods of the field (for example, flow-charting).

Even though today's systems engineer may not have all the qualifications mentioned here, he operates effectively by concentrating on industry, product, or application areas. However, these qualifications indicate the technical goal for which a professional systems engineer should strive.

At present, systems engineers are being drawn from many different sources: college graduates with degrees in engineering, physics, economics, business administration, mathematics, etc.; systems service personnel with years of experience in solving systems problems; data processing supervisors, methods analysts, and functional specialists, each with valuable training in particular areas. They are furthering their education in less familiar systems engineering subject matter both through self-development and formal training programs. Knowledge and experience are prime assets in the systems engineer of today, regardless of where he has acquired them. Later, as the profession matures and more universities offer courses specifically designed to cover a comprehensive body of systems engineering knowledge, a specialized college degree will become an important consideration in selection and placement.

Leadership and Administrative Responsibilities

Whether the systems engineer is working alone, or as leader of a study team, he needs a certain amount of skill in dealing with people. A team leader will, for example, be required to:

- Prepare project plans for extended periods to maximize productivity and creativity.
- Organize personnel with various specialized

skills into a cohesive working team.

- Schedule and assign the work in accordance with the availability of skilled manpower.
- Check and control progress to schedule, and replan as necessary.
- Report progress periodically to IBM and customer management.
- Communicate ideas, plans and technical interpretations effectively, both orally and in writing.
- Maintain study team morale at a high level, and motivate individual performance at an acceptable pace.
- Schedule and moderate regular study team meetings to report on current progress and plan future work.

When a systems engineer is cast in the role of project leader, capability in such human relations as those indicated above is of vital importance to study team efficiency and to IBM-customer relationships.

The systems engineer's leadership role is influenced in large measure by the customer's size and prior experience with equipment. It may vary from active participation to advising and counseling customer personnel as they conduct the study.

Sales and Promotional Responsibilities

Customers and potential customers judge a company in many ways: product quality and price, service, and the calibre of its employees. The image of IBM presented to customers and the business world is conveyed mainly by salesmen, customer engineers and systems engineers. While a salesman bears the major responsibility for customer relationships, the systems engineer can positively affect this association to the degree he provides a thoroughly satisfactory service through the application of his technical competence. As an IBM representative, the systems engineer accepts this marketing responsibility if he is to carry his fair share of the IBM team selling effort.

A systems engineer, as another arm of the marketing organization, fulfills this responsibility in a number of ways:

- Awareness and communication of short- and long-term company goals and objectives.
- Knowledge of and ability to discuss the efforts of IBM in research and development, space projects, personnel relations, etc.
- Active participation in community and professional associations.

Section	Area	Page
00	02	05

- Promotion of standard products from other divisions (such as typewriters and dictation equipment) and newly announced data processing equipment.
- Distribution of reference manuals on systems, applications, programs, etc.

In the main, however, the real value of a systems engineer stems from his position as a technical consultant to the customer, and the degree to which he:

- Applies knowledge from the disciplines related to his work.
- Defines, analyzes and resolves problems in a creative manner.
- Plans for the integration of equipment, personnel and procedures into a serviceable operating system.

These efforts are of special consequence in the increasingly commonplace situation where several data processing manufacturers are competing for the sale. Here, the systems engineer who works very closely with the salesman may contribute to a favorable resolution of the problem by:

- Defining the true system or application requirements, and specifying an equipment configuration that properly satisfies them.
- Studying strengths and weaknesses of competitive equipment and counseling the salesman accordingly.
- Assessing and interpreting sales approaches which other manufacturers will most likely pursue (for example, emphasis on tape sorting, or paper tape as an input/output device, etc.)
- Assisting in the formulation of IBM sales strategy, as well as in the writing and presentation of the sales proposal.
- Insuring that supplementary values such as equipment modularity, progressive implementation, and special IBM services before, during and after installation are adequately stressed.
- Determining and explaining programming systems that will be of value to the customer.
- Supplying technical follow-up wherever it is needed (for example, by making calls on the customer with the sales representative during the period the systems decision is being made).

Other Responsibilities

A systems engineer should constantly be on the lookout for opportunities to be of service during the course of routine systems operation. Such opportunities may occur when a machine has been inoperative for an excessive period of time and needs service, when a technical problem arises on a new application requiring a systems engineer's attention, etc. A reputation for service is built on the successful handling of many small matters.

Classroom teaching affords the systems engineer an unusual opportunity to discuss routine problems and maintain continuing communication with customer personnel during sessions on programming and use of the equipment. The primary purpose of these classes, though, is to teach the customer everything he should know about the equipment, and how to secure full value from it (for example, the importance of chaining instructions on a 1401, the kind of programming necessary if the print editing feature is not present, the significance of the dual-speed carriage in relation to the work performed, etc.).

As he applies data processing equipment to the solution of customer problems, the systems engineer is in a position to observe limitations and the possible need for new equipment options and features. These opinions and findings should be conveyed back to management for action. While this is an indirect or implied responsibility, it is the systems engineer's way of upgrading product value and maintaining a dynamic product line.

PROFESSIONAL DEVELOPMENT

Systems engineering is a new field of professional endeavor, maturing as it grows. A systematic body of knowledge is being developed gradually, as are professional standards of performance. These are mainly problems for universities and professional societies, but an individual systems engineer can make his own personal contribution through several avenues:

- Keeping abreast of current developments in the data processing field by reading technical journals and texts, attending seminars and symposiums, and discussing new ideas with other systems engineers.

- Adding to the body of knowledge by designing new applications and devising new methods to formulate and resolve systems problems.
- Documenting and reporting new approaches and results of current projects in technical bulletins, symposium publications, and through other specialized forms of communication.
- Participating in outside professional organizations to exchange ideas and information.

- Preparing self-development plans to indicate personal career objectives and alternate methods of attainment.

By these actions, a systems engineer not only promotes his own growth, but also contributes to the further development of the profession and the continuing leadership of IBM in data processing technology.

STUDY AND DESIGN

The life cycle of a business system is divided into three stages: study and design, implementation, and operation. In the first stage, a new system is designed after the present business has been studied for understanding; in the second stage, a detailed design is prepared and the system is installed; and in the final stage, the system is placed in routine operation, then modified and improved. While a systems engineer has key responsibilities in each stage, more and more of his work is being directed at the first stage, study and design. In many instances, present systems engineering work lacks the scope and range described here; however, the reader can see where the particular tasks fit into an overall structure or pattern and can gain an insight into the direction of the future.

Study and design is separated into three phases:

1. Study of the business as it is presently constituted.
2. Specification of the present and future requirements of the system.
3. Design of a new system.

STUDY AND UNDERSTAND THE PRESENT BUSINESS

The existing business is studied to gain insight into principal relationships -- that is, what inputs are accepted, what operations are performed to transform inputs into outputs, and what resources are used. The magnitude of the study depends on several factors:

- Type and size of business
- Size and scope of study
- Depth of penetration and refinement

Variability of Application

Systems engineers working with IBM salesmen and customer employees may conduct studies in a wide variety of businesses: banking, manufacturing, retailing, insurance, government operations, etc. Each different industry follows unique policies, practices, and regulations in servicing its markets, all of which will have a decided impact on the final system design. Within an industry, a separate business may comprise a few hundred employees working at a single location, or it may be composed of thousands of personnel situated in multi-plant locations in several states.

In addition to business type and size, a systems engineer's task is defined by the scope of the study.

He may be called upon to study an isolated, specialized area of the business, or he may become involved with a complete, across-the-board appraisal of all operations. Furthermore, the assignment may be to simply mechanize all or part of a manual procedure or to create a completely new system where few restraints are placed on design results. Study objectives will largely determine how much data is to be gathered and analyzed. Relatively little data gathering would be required for a straight mechanization study, but considerably more would be necessary in a highly creative redesign. Consequently, the systems engineer will at one time be working as an individual with a well-prescribed problem, and another time be part of a specialized team devoted to the development of a large, complex system for a multi-plant business. Subsequent discussion cannot always be focused directly on each of these broad spans of situations, and the systems engineer will recognize that he will have to make his own interpretations on the applicability of material to his conditions.

Study Organization and Planning

Prior to the start of a study, a careful plan of action should be prepared, stating the scope and purpose of the work to be performed and how the time and resources of the team are to be employed. The plan reflects management judgments and desires, as well as technical considerations of the study team. For a small, uncomplicated study, the plan might be a verbal agreement between management and IBM sales and systems engineering representatives; for larger studies, a definition of study scope and purpose, outline of tasks, and detailed time and cost schedules would be included in the plan.

Purpose, scope, time, and depth of study influence the size and composition of the study team. Unless the study is quite small and straightforward, a basic study team may comprise two systems engineers, since two people working together can provide each other with a stimulus of ideas to make their combined efforts far more than twice as productive as one person working alone. Larger-scale studies often require offsetting talents: functional specialists, mathematicians, management science specialists, etc. The team leader here should be a person with demonstrated abilities for planning, organizing and administering work.

As the study moves along, the team should keep management periodically advised of progress. During review sessions progress is summarized and measured against the schedule, current problems discussed and resolved, and plans made for the periods ahead.

Data Gathering Objectives

To secure proper understanding of the present business, initial data gathering is concentrated on general features and topics. This provides a broad view of the business as it operates in its special environment. Attention is focused on structural elements -- that is, inputs of materials as they are received from suppliers, outputs in products and services as they are produced for customers, and resources of inventories, finances, personnel, and facilities as they are applied to the production of goods and services. Finally, the operating dynamics of a business are examined and documented to show how inputs are converted to outputs. Cost, flow, sequence, timing and accuracy of events are important factors to identify in this part of the study, and may require special analyses.

Only the data with direct relevance to the study is gathered, and collected data should be continually questioned for its value in defining the present system and for its potential use later in specifying system requirements and developing a new system design.

General Business Data

Particularly in large-scale studies it is necessary to understand a business in wide perspective. To an extent this is also true in smaller studies but the subject is treated much more briefly.

Present goals and practices of a business are frequently shaped by important historical events. Identification of these major milestones of progress from the past can help to explain why a business operates as it does, and in what direction it is moving.

Industrial background and rank within the industry is also of significance in the interpretation of current policy and practice. How profitable is the business in relation to its competition? Are certain practices followed by all members of an industry? Does this business set industry policy or is it a follower?

Since systems are designed to support and fulfill business goals and objectives (that is, those special contributions a business makes to its environment), goals are defined in this phase of the study.

Where they have not been expressed explicitly, or require redefinition in concrete rather than general terms, interviews with management are helpful in arriving at an acceptable definition of basic goals.

Business goals are implemented by policies and practices. Policies are ideas, attitudes and philosophies as distinguished from methods and procedures -- for example, a code of ethics, a plan for expanding into new territory, an attitude toward employees and promotion, etc. As such, these courses of corporate action are important as explanations of the way a business is conducted.

The impact of federal, state and local regulations has to be evaluated for its effect on business operations. Tariffs, subsidies, banking law, utility regulations, agencies, consent decrees, etc., all either restrict or support business operations, and in turn influence recordkeeping procedures.

Structural Data

Structural data depicts the interaction between the external business environment -- for example, customers and vendors -- and the resources of the business. As it is gathered, the systems engineer makes a general review of major inputs, outputs and resources of the present business.

The market for products and services is broadly described, with emphasis on sales and distribution characteristics of existing and planned products, as well as information on past and future trends of product demand. Methods of marketing are analyzed -- that is, whether the business sells directly or through franchises, wholesalers, and retailers. Specific sales promotion strategies are described, and products are compared with those offered by the competition. Information on trends in product mix, shifts in the composition of income, and breakdowns of income by product lines are noted, where this contributes to the total description of the market.

Materials and suppliers (inputs to a business) are examined and described in the same manner. Input materials and services are classified by source, type, total and unit cost, availability, cyclical need, or other features. Major vendors or sources are ranked according to dollar volume. Procurement practices and competitive conditions in the market are assessed for their effect on various segments of the business. Throughout this analysis, only the inputs essential to fulfilling business goals are documented; thus, office supplies in a manufacturing concern would be excluded,

Section	Area	Page
01	01	03

while raw materials used to produce goods are included, whether supplied by an outside company, or another division of the same company. In some types of business -- for example, public utilities -- availability of supplies has to be projected over long periods of time. Inputs for a lending institution would include types of investors providing funds, nature of fundings, and information on the amount of funds secured by various categories.

A third class of structural data includes the resources of a business: finances, inventories (both physical and informational), personnel and facilities. Where companies are publicly held and government regulations require public disclosure, a wealth of data is usually available, although breakdowns by division and product line may be confidential. The balance sheet of financial operations, statement of income and retained earnings, and statements of expense and overhead are basic documents to be surveyed in financial analysis. All three may be found in the annual report along with notes explaining major entries. The consolidated expense report is useful during the operational review for making cost allocations by selected categories.

Personnel of the business are structured through one or more organization charts to show lines of authority, numbers of persons reporting, and responsibility. Occasionally these charts may be outdated or nonexistent, and the systems engineer will have to compile them on his own. Reports classifying personnel by component, skill, location, pay class or job level and other groupings can be used to indicate organization patterns by other than conventional reporting lines. Other types of personnel data are also studied where appropriate: union membership, local labor markets, turnover statistics, fringe benefits, stock options and profit-sharing plans, management attitudes to employees, etc.

Inventories fall into two general classes: physical stocks of raw, in-process, and finished goods and materials, and files of information. Physical inventories and money (financial enterprises) act as buffers between peaks of demand from the environment and ability to produce within a business. They are classified by type (such as raw material), location, value, and cost of maintenance. Informational inventories include the operational experience, decision rules and logic, and historical data on operations as they are contained in files (for example, a sales catalog, a library of computer programs, a file of account records). In addition to files, information inventories include standard procedures and instructions, and the experience and

knowledge of people accumulated over time. A description of physical and information inventories encompasses many different characteristics: level, flow, turnover, age, cycle, demand, access, and capability for reaction to variations in demand.

Facilities include land, buildings, machine tools, and data processing and communications equipment. Manufacturing plants, sales offices, research facilities, warehouses, distribution centers are itemized and supported with detail on layout, area, capacity, cost and application. Plans for new construction and expansion are noted. Machinery and data processing equipment are listed by class, application, cost, where this data is useful. The communication network (for example, teletype network in a multi-location investment house) is a most essential facility in certain businesses; its characteristics should be described in terms of switching centers, costs, current and projected volumes and capacities, transmission speeds, and interoffice connections. Locations can be displayed on a network map.

Customer's Data

The systems engineer must be careful to preserve the confidential nature of the information which he may gather in the course of his study. He should be discrete in discussing the customer's affairs both inside and outside IBM, and he should not unnecessarily release or circulate documents the customer shows him. Typical examples of information the customer may regard as confidential are customer lists, unpublished financial data, and future plans, but there may be others. In making studies for process control systems, the systems engineer may also learn secret processes and other proprietary data which must not be revealed or disclosed except in connection with the study of the customer's problem.

Data Gathering Techniques

Generally, data about the present business is gathered by employing three different techniques in varying combinations:

- Interviewing
- Sampling and estimating
- Searching records

Interviewing is perhaps the most productive method of securing information, particularly when time is at a premium and record files are poorly maintained. This technique can be used to collect

basic statistics about a business by questioning employees; at a more formal structured level it is useful for gaining understanding about complex interactions within the business (depth interviews).

Sampling and estimating are especially valuable in dealing with large volumes of data in complex situations. Information on elapsed and response times usually is not readily available, for example, and can be established economically by examining a valid sample of data. The determination of a valid sample can be simplified by using any of a number of published nomograms on the subject.

Searches of files, records, and publications are frequently necessary, but may be too time-consuming. At all times, the systems engineer must assess the end results of data gathering against the cost and time involved in collecting the information. Considerable general and structural data, however, can be found by searching sales catalogs, trade and government publications, annual reports, company forecasts, summary reports on current operations, etc.

Analysis of Business Operations

These early investigations are aimed at a broad, generalized interpretation of the entire business as it functions in its environment. To complete the review of the present business, the systems engineer turns next to a more detailed analysis of operations, and concentrates on an analysis of the various transformations and conversions that take place in producing outputs from inputs. In so doing, the task may become so voluminous that he may need to break the analysis into segments for more convenient manipulation. It is important to remember not to confuse detail with understanding. Gather only enough information to gain effective understanding.

Businesses are organized and structured into departments or functions because an individual can manage only so many people effectively, and there is the further consideration that various types of work can be grouped by the need for special skills and knowledge. In a manufacturing concern, for example, the conventional vertical structure includes marketing, engineering, manufacturing and accounting functions, while in a bank, sections may be organized by time deposits, demand deposits, loans, etc. Each of these major functions is then divided into more specialized fields of work -- for example, in a manufacturing department, there will be any number of specialized sections: production, engineering, quality control and inspection,

production control, purchasing, traffic, shop operations, etc.). For a small study isolated to assigned areas of the business, this poses no problems for the systems engineer, but in multi-function or overall business studies, a decision must be reached on whether to conduct the analysis to the limitations of vertical organization structure, or find another way of classifying the business so that it facilitates analysis. An alternative to conventional organization is the restructuring of a business into study-oriented classifications called activities.

The Activity Concept

An activity, in its special meaning for systems studies, is a logically related group of business operations directed toward the fulfillment of one or more goals of the business. An activity is, when properly formulated, generally self-contained and has few informational ties to other activities. Since activities are goal-directed, they often cut across functional business lines. Such a classification has no necessary impact on actual organization patterns, but it does provide the systems engineer with a means of more effectively accomplishing his responsibility -- system study and design. When the activity concept is used, this places a greater importance on the way business goals and objectives are defined, since activities are developed from goals. Where applications are prespecified the activity concept can have only limited usefulness.

Technique of Activity Formulation

Activities are derived from extensive exploration of the nature and practical requirements of a business. Preliminary definitions arrived at during this phase of the study may well stand up without revision, but in large-scale studies, the first approximation probably will be modified in the light of subsequent developments and further thinking into the problem.

No single best method exists for defining business goals and activities, then interrelating the two. In general, the systems engineer either starts at the top of the business and works progressively from the general to the detailed level, or he reverses the process by starting out with details and working back to generalizations by carefully regrouping details into meaningful wholes. There is a third possibility based on combining the best features of these two contrasting methods.

Under the top-down, or deductive method, statements of specific business goals are prepared in conjunction with management. From these statements, activities are formulated to encompass the

work necessary to carry out individual goals or groups of goals. The process is iterative, and as activities are laid out in detail, certain modifications in goal definitions may occur. Once activity content has been decided on (at least temporarily), operating data is analyzed and documented to provide a description of its internal composition.

With the inductive method a systems engineer initiates operations documentation without any prior structuring as he encounters information in moving from one department to another. When all the key operations of a business have been recorded, they are combined into logically related groups. A rough flowchart is drawn up for each group and studied for content, balance and completeness. To establish an acceptable activity, operations may have to be regrouped several times, and more extensive interviews conducted with personnel to compile additional data and verify results. When the activities have been arranged to the study team's satisfaction, they are documented in required detail.

The third or composite method contains features of the first two. Using goals stated by the study team and management as a guide, activities are formulated and one or two are selected for examination. If the results show acceptable goal and activity alignment, as well as activity content, the remaining ones are documented in the same fashion.

Choice of a method depends largely on how well the true goals of a business are understood and can be clearly stated early in the study, how much freedom a study team has, and how far activity scope and definition departs from existing organization patterns. The deductive method is more appropriate when a study team is fairly well constrained by the desires of management in regard to the analysis, or when the goals and activities closely follow current practices and organization structuring. In the case of small-scale studies, or studies involving only minor segments of the business (for example, the production scheduling, dispatching and shop control elements of a manufacturing concern), there may be no need for an elaborate goal and activity definition. When the activity has been predefined for the systems engineer, documentation of operational data proceeds directly and without further consideration of definition and boundary statements.

The inductive method is normally used when relatively few restraints are placed on the study, and a completely new system design is the objective. The composite method finds its best expression when one or two activities of the business predominate, and the others, although considered, are of secondary importance.

Activity Boundaries

The principal reason for formulating activities, as mentioned previously, is to separate a business into manageable and relatively independent elements which can be easily understood, analyzed and evaluated. Activities position a business for study, but do not realign an organization into new and permanent arrangements. Activity structure must be logical and acceptable to management, who will eventually make systems judgments in terms of these definitions. Therefore, activities should not depart so far from conventional practice that management cannot easily relate activity costs back to established accounting classifications. On the other hand, functional organization tends to conceal true business goals, and is not always a suitable framework for conducting a study of an information system. The ultimate decision on how far to implement the activity concept rests with management. Where the business is small or the present functional organization is adequate for study purposes, the decision may be to proceed with documentation along these lines. But if the business is complex and many areas are included in the study scope, then it is well worth a systems engineer's time at this point to assist in devising an activity alignment that will make his job easier to execute.

Compiling Operational Data

Operational data gathering procedure is affected by how well individual activities have been defined.

If activity boundaries are well established as a result of the formulation process or nature of the study, then the systems engineer gathers personnel, materials, equipment, and other costs relating to the activity and draws up one or more system flowcharts to describe the sequence of events and time characteristics between operations (an operation is a related set of processes applied to the conversion of inputs into outputs, using resources to effect the transformation -- for example, "Compile master production schedule from customer orders and requests for bids").

Where activity boundaries are not clearly evident in advance, then operations are documented as they are encountered, integrated into activities, displayed on flowcharts, and costed after the activity content has been revised and settled.

With either approach, the objective is to arrive at a total time and cost for each activity. The resulting flowcharts of operational sequence and time should be reconciled to the total costs assigned to the activity.

Documentation Format

The IBM Study Organization Plan, a comprehensive program outlining a total approach and methodology for systems studies, includes a documentation technique useful in recording operational (and other) data. Under this plan, an organization chart for the business is displayed at the top of a Resource Usage Sheet. Below each departmental or functional block of the chart, personnel, material, equipment and other costs are inserted as they reflect the current expenses for operating each segment of the business. Since the total of the departmental costs closely approximates the complete operating expense of the business, this provides a systems engineer with a figure to reconcile against as he reassigns these costs to the several activities, or builds them up from an analysis of separate operations. The flow and sequence of events within an activity, and the elapsed time between operations are displayed on another form, the Activity Sheet. Consequently, when system flowcharts have been completed for each activity, the cost of inputs, outputs, operations and resources should show up on a band of the Resource Usage Sheet assigned to that activity. Other forms (Operations, Message, and File Sheets) are provided for more detailed operational analysis.

Depth of Penetration

During operational analysis, management must indicate how deeply the systems engineer will probe the business, and how extensively he will document operations for this and subsequent phases of study.

For many studies, it may be quite sufficient to document activities at a very general level. Thus, in a small business, a single application, or in a mechanization study, the Resource Usage Sheet and one or more Activity Sheets would comprise the entire operational documentation. But in complex studies, or ones leading to a completely new approach, it may be necessary to have expanded descriptions of files, messages (a recorded or unrecorded communication from one point to another -- for example, a report, a telephone call, or a sales order slip), and the various processes or actions that make up or take place within individual operations (for example, "compute monthly withholding tax" is a process within the operation "calculate net pay"). When the need for detailed documentation is established, IBM study documentation forms such as the Operation Sheet, the Activity Sheet and the File Sheet may be prepared for all activities, for selected activities, or for special areas within a single activity.

Cost Analysis

Accounting classification of costs in a business seldom conforms to expense distributions needed in a system study -- whether activities closely follow conventional organization lines, or depart sharply from them. Therefore, after the systems engineer has compiled departmental or functional costs from operating expense reports and other sources, he will need to employ special methods to accomplish cost allocations by operation and activity.

Interviews with supervisors, sampling, and estimating should be employed wherever possible in cost analysis to save time. If, for example, three activities run through one department, costs can be distributed in accordance with answers to such questions as:

- How much budget is allocated to each activity?
- How much would operating expense be reduced if this activity were removed from the department?
- How much would it cost to establish this activity as an independent department?

The interview approach is not always effective, so the systems engineer may have to rely on estimates. Estimates should always be checked with cost accountants and departmental supervisors for reliability, and eventually reconciled to the total cost being distributed (that is, the consolidated overhead statement or similar control figure).

Cost analysis can be conducted by successively breaking down gross departmental costs (or budgets if this is easier), or by working up from operations. In a recent study, the systems engineer used this latter approach, estimating costs for the inputs, outputs, processes and resources (files) involved in each operation of the business. Upon completion of the analysis, the sum of these costs was compared with total operating expense. The two figures were then reconciled through a series of interviews with the supervisors. Many studies will not require this level of detail, and a total cost of personnel, equipment and material by activity will be adequate.

Time Analysis

Various types of time data are gathered to show how long it takes to perform an activity, how much time elapses between operations, and how quickly the system can respond to normal and emergency situations. Peak, as well as average, times are included.

Sampling techniques are more appropriate here than interviews, because people are less accurate in appraising time than costs. Time data may be

documented operation by operation on the grid area (left) of an Activity Sheet and on Operations Sheets, if this level of detail is desired.

Elapsed time within operations (that is, the time between the occurrence of successive inputs, or the time between the arrival of the first input and the availability of an output) can be secured by following a live sample of documents through a series of operations, and time-stamping them after they have been processed (this avoids the implication of a time and motion study). Time relationships can also be reconstructed from file data, correspondence, memos, and other documents which fix time in one manner or another. Peaks can be determined by running samples during periods when the peak is most likely to occur. Frequently, time data can be readily converted to costs by assigning an average rate or estimated costs to the time associated with an input, process, output or file.

Volume and Frequency Analysis

Peak and average volume and frequency information for inputs, outputs and files is gathered over representative periods and displayed on the grid area of the Activity Sheet alongside time data (or on Operations Sheets). Record searches of files and reports will produce much of this data, but it should be supported by special tests or samples to pinpoint cycles and fluctuations in these average figures. Volume data is not meaningful unless associated with frequency of occurrence. Clues on peaks, cycles, and frequency are obtained through familiarity with the business and its environment.

Interrelationship of Time, Cost and Accuracy Data

Activity Sheets demonstrate a flow and sequence of events, in addition to volume, frequency, and time information by operation for a defined activity. The Resource Usage Sheet shows costs from the larger background of the business as they have been allocated to functional organization components, and further to designated activities.

Regardless of the technique employed, a final step in operational analysis involves a reconciliation of the two. In so doing, the costs shown on a single activity band of the Resource Usage Sheet become representative of the events displayed on a related activity flow diagram.

One additional item is the development of quality measures to determine the output accuracy, the correctness of the present system.

These major documentation forms (and the more detailed Operations, File, and Message Sheets) may be supplemented with special time, cost, accuracy and volume studies considered necessary.

Report Preparation and Communication

Final results of the study of the present system can be summarized (if necessary) in a report to management. Various exhibits can be displayed in three sections -- general, structural and operational -- to accommodate a gradual transition from general to detailed information about the business. Among others, the report should have these objectives:

- Demonstrate understanding of the existing business.
- Develop a new view of the business in terms of activities.
- Establish benchmarks in time, cost and accuracy.
- Provide an adequate base for conducting later phases of system study.

Report length will vary with the size and scope of the study. For a small business, an application study, or straight mechanization, the present business report might contain only a few pages of general and structural information followed by a Resource Usage Sheet and one or more Activity Sheets. In a large-scale, multi-plant study, general and structural data could run up to 20 or 30 pages, supported by a Resource Usage Sheet, several Activity Sheets, and dozens of Operations, Files and Message Sheets. An appendix might even be necessary to display results of special studies, and detail to supplement the several analyses.

The finished report is transmitted to management for review, and perhaps an informal meeting is scheduled to discuss the findings, interpret special analyses, and prepare for advance phases of study.

The total length of time for doing the work described in phase I can vary from a few days to two or more months.

DETERMINE SYSTEM REQUIREMENTS

Once the systems engineer, together with the salesman and customer personnel, has gained an understanding of the business as it presently exists, he turns his attention to the second major task of study and design: a determination of what the system is logically required to do to meet present and future

business goals and objectives. To accomplish this, two basic questions have to be answered:

- What is the system required to do in terms of inputs, outputs, operations and resources?
- How is the performance of the system to be evaluated?

Variability of Emphasis

Emphasis on requirements varies from one study to another. Where the nature of the business is undergoing change and growth, and new business requirements differ from the present, a thorough analysis and specification is needed to adequately prepare for new system design.

For many small business applications, or in mechanization or improvement studies, the specification might contain only a brief statement of objectives, recognition of potential growth, and an outline of a few factors to appraise the value of the system.

This does not imply, however, that the systems engineer may jump directly from a study of the present business to new system design even in a study of restricted scope and purpose. The specification can be used to point out areas of opportunity in an improvement study, thereby upgrading what may have started out as a simple system modification. In a straightforward mechanization, the requirements specification is valuable for bridging the gap between two basically different types of systems and hence establishes a base for more effective design.

Application of Current Business Data

To a certain extent, data is evaluated as it is gathered and organized into manageable activity groupings throughout the present business study. Yet, if this assignment were carried out with all due speed (as it should be), the evaluation would not be thorough or complete. A first step in the requirements phase involves a review of this information for a number of reasons:

- Problem areas should be clearly defined and diagnosed as to cause.
- Preliminary goal and activity definitions should be reviewed to insure that their scope and content provide for future conditions.
- Present operational data has to be projected to show potential future levels of volume, rate, frequency of occurrence, etc.

Management may also want to have some idea of potential savings at this point in time, but beyond

a generalized, conservative approximation of affectable dollars (that is, the total dollars in a business that the new system can affect, but not necessarily displace), such statements should be delayed until a formal new system proposal is prepared. The IBM study team in a total business study may wish to make this judgment privately to indicate the activity on which to concentrate initial emphasis, if cost savings are an important consideration.

Refine Business Goals and Objectives

Current business goals were documented earlier; the task now centers on confirming or revising this definition, and insuring that future as well as present goals are reflected. Management determines goals for a business, but system planners must understand and interpret them if systems are to be designed for their fulfillment.

In goal analysis, broad general-purpose statements are converted to direct, specific definitions (for example, profit is a general goal of most businesses) of what a business is attempting to do. In this respect, they are ends rather than means. Business goals and objectives are stated in terms of:

- Products and services to be supplied.
- Maintenance of physical and informational resources.
- Improving relationships with consumers, the public, shareowners, etc.

Information on goals and objectives is obtained from management interviews, a survey of long-range business plans, a review of industry projections and similar sources.

Precise goal statements are needed to supply proper direction for a study team in a large complex business; they are of less importance in mechanization and improvement studies.

Establish Activity Scope and Boundaries

Activities are the means by which goals are achieved. Already formulated in the present business study, they are accepted or modified to conform to the new goal statement. Generally, activities should correspond to goals in a fairly close one-for-one relationship, but there will be instances when a number of activities contribute to a single goal, or when a single activity will contribute to two or more goals.

The process of aligning activities with goals yields broad activity definitions. These generalized statements are then amplified to show activity

content, as well as scope and boundaries. Exclusions are just as important to designate as inclusions.

Individual scope and boundary descriptions are checked for internal completeness and consistency against present business activity sheets, and for practical working size. They are also compared with one another to resolve conflicts and overlapping, and to insure the relative independence of each one. Throughout requirements specification, analysis is oriented around the size and content of a single activity, a condition that facilitates simultaneous execution of much of the work.

Requirements Determination

Thus far in this phase, the systems engineer has been preparing for subsequent analysis; problem areas are now defined more accurately, and present business data has been defined and clarified. The central task of this phase, however, is the determination of what a system is required to do, activity by activity, in terms of:

- Inputs it must accept
- Outputs it must produce
- Operations it must perform
- Resources it must use

To identify requirements, the systems engineer moves away from present operations and views a business from the position of what is logically necessary (or imposed) to fulfill the goals of an activity. Establishing logical requirements first, then adjusting them to accommodate practical or necessary considerations produces a more valid specification than one devised solely on the basis of present operations. Most activities contain many variable and interacting factors and conditions that make it difficult to accomplish this in a single attempt; frequently, one or more iterations are necessary to arrive at a satisfactory specification.

Descriptive requirements are developed first for inputs, outputs, operations and resources, then amplified with quantitative data such as volumes, frequency, timing and rate. To increase the precision of requirements, techniques of correlation analysis, trend projection, and even simulation are applied where necessary.

Required Inputs and Outputs

Outputs are closely associated with activity goals and for this reason are often subject to analysis first. Working from collected data and activity goal and scope statements, a systems engineer

specifies those outputs that are logically needed to satisfy a goal, and those that are imposed by the environment or by management (for example, a W-2 form).

Customer acceptance controls the form and content of many outputs, particularly in a business competing in services (such as banking). Legal and audit stipulations either affect outputs or are outputs in themselves (for example, an income tax report); industry practice, as recognized by individual companies, affects output formats. An output may become an input to another activity and therefore must be acceptable to it; it may also be required as feedback for repetitive processing or recycling of an activity.

Required inputs have many of the characteristics of required outputs, and similarly may be imposed by the environment or other activities. In addition, some inputs are logically required to furnish information for outputs, and provide access to, or maintain, resources.

Each input and output is analyzed for peak and average volumes and rates, cyclic or periodic properties, trends and patterns -- to a degree of detail that is appropriate for the study. Present business data on peak and average volumes can be used as a base, as long as it is expanded to encompass future requirements over the expected life of the system. Sales forecasts, operating budgets, and management plans are sources for this information. In the absence of this data, estimates of potential growth can be made from industry forecasts, or from present averages extended into the future. Sampling techniques are employed whenever possible in trend projection to cut down the volume of data.

Certain other input and output information is needed for the specification beyond descriptive and volume notation: media (for example, 8-1/2 x 11 form, letters); source and destination (for example, customer, shop); number of fields and characteristics; and some indication whether form and content are fixed or may be altered.

Required Operations

Required operations are those that are necessary to produce a required output, to accept an input, to maintain a resource, to fulfill an activity goal, or to satisfy an audit or legal requirement. Operations are determined through observation, by hypothesizing operations logic from required inputs and outputs, or by experimentation (such as simulation).

Operational analysis is conducted after inputs, or outputs, or both have been specified. In this way, the transformation reasoning or logic between inputs and outputs can be more clearly demonstrated. With this approach, only the processing steps that are really essential are identified. Another approach is to decide how present system operations can be modified, improved or deleted to accommodate the newly stated input and output requirements. This latter method is useful when there will be no significant changes in operation content or sequence in the new system.

Three types of data describe the quantitative characteristics of an operation:

- Input and output factors (that is, number of fields of data that must be accepted and produced).
- Frequency of execution (that is, how often the operation is performed over a selected time period).
- Process summary.

In this latter category, the nature and estimated number of processes involved in the execution of an operation is noted by showing:

- Number of arithmetic and logical processes to be performed.
- Number of relations (comparisons of one factor with another) to be examined.
- Number of times a resource is to be consulted for data (lookup).
- Number of edit or audit functions necessary.

Operations Logic

Relationships among operations are often complex; they may show sequential, causal or time dependence from one to another. At times, special techniques and documentation formats will have to be applied to properly construct the cause-and-effect relationships among a series of events. Actions do not just happen in a business; they are caused by some prior event, or combination of circumstances, or have some sound basis for being performed. This underlying logic has to be understood before connections between events can be seen and understood. Where this logic becomes too complex for narrative description, flowcharts and decision tables can be used to analyze and display significant operational relationships.

Required Resources

Resources are likely to be subject to constraints, and therefore are specified last. In general, imposed personnel, inventories (physical and informational), facilities, and finances are listed along with quantity and cost information. Logically required resources, on the other hand, are left open on the specification to permit greater freedom in system design.

Personnel requirements are confined to skill and job descriptions, since the types of positions and related work content may change considerably in the new system.

Information resources (files) are described by name and cost, if imposed. Should characteristics change during requirements analysis, more detail would be needed: size of file in characters, age of data, retention rules, peak and average message volume, access requirements, etc.

Simulation offers certain benefits, particularly for large-scale, real-time system studies. For example, the level of raw, in-process, and finished goods inventories in a manufacturing concern can be calculated for several potential sales volumes with the various IBM Inventory Management Simulation programs. In a like manner, the IBM Job Shop Simulator or The General-Purpose Systems Simulator can be employed to ascertain the effect of changes in location, grouping, and utilization of machine tool and personnel resources.

Iteration of Requirements

At this point, input, output, operation and resource requirements have been developed first for one activity, then for the remaining ones within study scope. For improvement and mechanization studies, this may adequately state what the system is required to do. In larger studies, or ones where constraints are infrequent, there may be a need to define requirements more precisely. Completeness and consistency in the specification is facilitated by reversing the approach during this refinement process; if inputs are established at the outset, then the system engineer begins with outputs the second time around and works in logical steps back to inputs.

Performance Measurement

The second major task of requirements specification involves the construction of a measurement plan for appraising the value of the new system design. A measurement plan consists of two parts:

- A series of time, cost, accuracy and volume measurement factors.
- An indication of the relative importance of each selected measurement factor.

Identification of measurement factors and rating scales is performed simultaneously with the specification of requirements.

Measurement Factors

Some measures are direct and quantitative and can be displayed easily on numerical scales -- for example, dollar cost of a system, number of days from receipt of a customer order to shipment of a finished product. Others are indirect or derived from combinations of factors -- for example, budget variance, return on investment, productive efficiency. Still others are qualitative or subjective -- for example, product appearance, customer goodwill.

A representative number of measurement factors are needed to project a balanced appraisal of system performance. To achieve this breadth of coverage, several major measurement categories are designated first (for example, time, cost, efficiency, reliability, accuracy, flexibility, etc.), then a number of detailed measures are identified within each general category. For example, "Time" would be a major category in airline and brokerage studies; it is refined to show such measurable factors as customer inquiry response time, record update time, order processing cycle time, etc.

In selecting measurement factors, it is better to have too many than too few. In particular, each side of an interacting condition should be measured (for example, length of product cycle vs. level of inventory required to support the cycle).

Relative Value

A relative value is determined for each measurement factor; it should consider:

- Present operating level
- Acceptable performance range
- Desirable performance range

Operating level information should appear among data already gathered. Acceptable and desirable operating ranges are the several points at which new system performance is considered to be satisfactory and more than satisfactory, respectively.

During measurement reviews with management, a systems engineer is not merely seeking information; he is attempting to discern how a manager feels about his business, what is important to him, and what he expects the business to gain from the new design. It is sometimes more important to develop insight into feelings, opinions, biases, and the pressures on an individual than it is to assign concrete values. To diagnose the basic reasons behind management decisions and opinions, a systems engineer not only has to be prepared with factual data, but must be adept at leading discussion on sensitive subjects, and interpreting the real meaning behind management statements. At times it will be difficult to discover the causes behind a management position. Unless they are brought out into the open, though, the systems engineer will not be able to decide just what is acceptable and desirable in a business. These situations have to be faced in most studies, and require the exercising of good judgment on the part of the systems engineer.

Measurement of performance levels is never an absolute, objective process. The systems engineer should be aware of the need to measure early in the requirements phase, and be constantly seeking an appropriate series of factors which adequately evaluate management objectives. After they have been selected and worked into a comprehensive framework and plan, they form the basis for judging the relative worth of various system designs.

Requirements Report

On completion of requirements analysis, the several tasks in system design can be initiated. In those studies where the system requirements specification is a marked departure from present practices (that is, where a creative design is

desired, and few constraints have been imposed), this is a convenient point at which to summarize progress and report results to management.

A requirements report is usually organized with a summary section, and individual sections for each activity. A summary section integrates the activity sections, assesses the impact of the analysis on the business, states business goals and objectives, and defines activities and their interrelationships. An individual activity packet includes such information as activity goals and objectives, scope and boundaries, general considerations on cost and policy, requirements for inputs, outputs, operations and resources, and finally measurement factors and their relative value.

BASIC SYSTEM DESIGN

In basic system design, the systems engineer lays out a master plan for the new system, concentrating first on the single, most important activity. Design possibilities are considered as they relate to system elements: inputs, outputs, operations and files. The best possibilities among these elements are fused into a few design alternatives, using various design concepts as a base. The process is repeated for the remaining activities. Finally, relationships among the design alternatives for various activities are analyzed for possible consolidation and compromise, to eliminate the overlapping and unsuitable ones. The steps in basic system design are summarized as follows:

1. Select the single, most important activity for initial attention.
2. Examine various design possibilities for each element of the activity: inputs, outputs, operations and files.
3. Synthesize these possibilities into specific design alternatives, each built on a design concept.
4. Repeat the process for other activities within study scope.
5. Consolidate the multiple activity designs into the best two or three to be carried into system selection.

Design Environment

System design, while requiring imagination and creativity, will be more fruitful if it is at the same time a disciplined effort. Thinking in terms of design concepts independently of specific equipment configurations is one key to the development

of effective system designs. This applies to all types of studies. The quality of the design can be substantially upgraded to the degree a climate of originality is cultivated to achieve the major breakthroughs a systems engineer seeks. Beyond the application of creative effort, other talents and resources should be brought to bear on the subject. Design quality also depends on the application of sound judgment based on experience, including knowledge of equipment characteristics and capabilities, and awareness of results accomplished in previous studies.

Activity Selection

On the basis of one or more of the following features, the initial activity is designated for basic design:

- Large size — either input/output volume, or computing complexity.
- High affectable dollars — potentially large savings are involved.
- Dominant characteristics — a single performance measure (for example, response time to customer inquiries) may be so important that it overrides other considerations.
- Management preference.

Analysis of System Elements

Using various sections of the requirement specification as a reference, input/output, processing, and file possibilities are examined individually for potential inclusion in the design solution.

Direct, specific questions are addressed to each subject area to probe the appropriateness of individual possibilities. If punched card input is being explored, for example, considerations like peak and average volume, elapsed processing time, verification procedures, economy, uniqueness, etc., are assessed before accepting cards as a reasonable input solution.

Throughout the analysis, emphasis is placed on dominant characteristics, since they have a decided influence on design. Frequently, input/output volumes and characteristics demonstrate this type of dominance.

Input/Output Characteristics

Inputs generally have to be accepted in the form they are received; the design problem is one of converting them to machine-sensible form. With outputs, the problem is one of producing them in a form acceptable to the environment (or another activity).

Before stating the form information takes to be acceptable for computer processing, input possibilities such as the following are considered:

- Are inputs to be handled separately, or processed in batches?
- Are inputs received in processable form, or do they have to be converted and edited?
- Can inputs be processed randomly as received, or do they have to be sequenced by certain control fields?
- What is the significance of input volume variation on system performance criteria, particularly during peak periods?
- Is interruption to be allowed? How does this affect reliability requirements?

Outputs are examined in similar fashion:

- Are reports to be printed or punched, or are they to be in some form of audio or direct display?
- How much of report content is to be summarized, detailed, or listed by exception?
- Which reports must be generated on schedule, demand, or exception?
- What are the requirements on readability, format, number of copies, etc.?
- Are standard forms required, or are non-standard acceptable?
- Will output go to another activity as a signal, tape record, or other machine-sensible form?
- Will output data be reused?

After these and other possibilities have been studied, and inappropriate, unlikely, or unacceptable ones set aside, documentation formats can be stated as punched cards, magnetic tape, etc. The selection of possibilities among activity elements takes into account equipment capabilities without specifying the exact input/output equipment.

Processing Characteristics

In the evaluation of processing possibilities, primary concern is on the magnitude and complexity of computing, and its implication on design. Among the design possibilities:

- Is processing random or sequential?
- How frequently are operations executed?
- What are the predominant characteristics in operations: arithmetic, logical, edit, relational, or lookup?
- How complex is operations logic in terms of computing time?
- Does processing include main-line operations only, or all exceptions?
- Will operations require significant restart and monitor routines?

Certain manual or human considerations have to be decided upon in addition to the above:

- How automatic or man-independent is the system to be?
- Is manual override necessary?
- What kind of audit trail is to be provided?
- Are errors to be corrected immediately or set aside for later processing?

Out of this analysis, a statement of processing characteristics is compiled (for example, number of operations per second, memory size, special instructions, etc.).

File Characteristics

Finally, possibilities are evaluated for files:

- Do separate files contain common data, and can they be consolidated?
- How frequently are they referenced for inquiry?
- What is the frequency of change, or updating?
- What is the growth rate?
- How many ways are files referenced -- that is, if they are in one order, can they be resorted for other sequences?

Conclusions on file possibilities are documented to show such characteristics as file size, average access time, etc.

All through the analysis, acceptable possibilities are maintained at a basic minimum.

Formulating Design Alternatives

Design alternatives are formulated by combining appropriate input, output, processing and file possibilities. Each alternative is built up around the unifying force of a design concept, which relates to the way a planned system will carry out information processing requirements.

Of the number formulated, many will be discarded because of inherent imbalances, rough timing information, obvious cost constraints, and other factors outlined in the requirements specification. With an extensive background in equipment capabilities, it is reasonable to conduct this type of evaluation without going through detailed system design or extensive throughput timing.

Most frequently, alternatives are constructed on a design concept that has already been proved and accepted. If not, more time will be needed to insure that the design is feasible and economic. Examples of design concepts include centralized data processing, regeneration (as opposed to file

reference), online processing, exception reporting, etc. The concept behind a standard application program, when applicable, should be specified as a principal option for the affected activity.

At the beginning of basic design, a systems engineer deals with parts and pieces; during alternative formulation, the role is reversed and events are visualized in organized patterns or as a whole. Under this approach, a design progressively becomes an integrated entity, rather than an aggregation of parts.

A large number of design alternatives may result from the basic design process. Alternatives are gradually eliminated until the best two or three remain for the principal activity.

Documentation

Enough data has been accumulated to prepare a generic system description, and again, the Activity Sheet can be used as the description format. It should contain a statement of major inputs, outputs, operations and files required to implement a design alternative, along with logical flow of information, associated volumes, time relationships, etc.

Multiple Activity Integration

In a multiple activity study, design alternatives are formulated for all remaining activities. The resulting best alternatives for secondary activities are then compared with the major activity for resolution of conflicts and incompatibilities. Activities are also reviewed as a group for potential consolidation among inputs, operations, etc.

Sometimes the dominant characteristics in a major activity are so overriding (for example, order and inquiry processing in an airline study) that other considerations are subordinated to them. When this occurs, other activities are accommodated either as equipment capacity is available, or as additional components are added to the system.

In some instances, there will be a greater justification for a design when a second activity is added to achieve the higher price/performance ratio which larger equipment offers.

Other techniques can be employed to resolve volume and time overloads as additional activities are studied for impact on the basic design: breaking down monthly billing cycles to weekly or semi-weekly cycles; matching input/output-bound activities with compute-bound activities, etc.

Consolidation potential exists for files, inputs, outputs and, occasionally, operations. Files of the several activities should be reviewed for

common characteristics, content, and application from information in the requirements specification, and combined where practical. One activity may use the same or similar inputs, or produce outputs similar to another activity. Inputs and outputs of the several activities are therefore studied in order to resolve incompatibilities in form, content, or timing of data between activities.

On completion of multiple activity integration, two or three design alternatives can be composed to encompass the several activities. Each of these composite design alternatives is then documented to provide a basis for system selection.

Summary

Basic system design leads to a definition of a new system at the generic level, and establishes a conceptual base from which system selection can be initiated. The depth of description should be quite limited, providing only enough detail to effectively support the work of system selection.

Systems engineers are called upon to develop system approaches under conditions other than those described in this section, and may prefer to introduce certain modifications to the steps suggested here as circumstances change. For example, a systems engineer may have good reason to proceed through system selection for a single design alternative before other alternatives are formulated; another may find the imposed constraints so considerable that few alternatives or concepts are open for evaluation.

There are other situations, too, that can influence the procedure to be followed: a new application is being planned for an installed system, a customer has a preference for a particular equipment model, or a special study has indicated the desirability for a certain equipment configuration. Now it is necessary to design approaches capable of solution by specific equipment. Design alternatives are still formulated, but on an equipment-oriented basis. When freedom of design is compromised by the equipment selected, it may, in fact, call for greater ingenuity in the design of a valid system approach than otherwise.

Whether evaluation of elemental possibilities, formulation of design alternatives, and subsequent activity integration are performed on an enlarged or diminished scale, they are necessary ingredients in the development of productive and economic systems.

SYSTEM SELECTION

System selection entails the development of a number of equipment configurations for each surviving design alternative, and successive refinement and reduction of these solutions to the one producing the best operating system. The qualification of the solution is established by the way it:

- Accommodates time, cost and accuracy factors and constraints (for example, a limitation on purchase or rental price) described in the requirements specification.
- Permits an economic investment in implementation.
- Demonstrates a positive effect on business profitability and efficiency.

Optimum system solutions are derived by following a careful plan, such as follows:

1. Evaluate the relative importance of various factors that have a bearing on the problem: customer needs, performance requirements, etc.
2. Postulate specific equipment solutions for a single design alternative.
3. Evaluate the applicability of the several possible equipment combinations to the design alternative.
4. Select the best equipment configuration for this alternative through a run timing and cost analysis.
5. Repeat the above steps for other design alternatives still under consideration.
6. Compare the resulting solutions with each other.
7. Designate a complete equipment configuration from this comparison process, taking into consideration its broad effect on the business and on the implementation process.
8. Identify and cost out the human and procedural aspects of the system as they relate to the equipment configuration to provide a total statement of the new system and its associated operating costs.

Again, these steps will vary with individual circumstances and special conditions -- for example, presence of installed equipment, desire by management to make the final equipment decision from proposed solutions, etc.

System selection is more than a process of selecting from among available equipment. As the systems engineer progresses, acquired knowledge is applied to succeeding steps, modifications and tradeoffs are introduced, and the procedure iterated to achieve better and more acceptable solutions. Consequently, the system evolving from the selection process is often a composite of several equipment configurations and design alternatives.

Equipment Selection

Several families or classes of equipment are specified at the outset for a design alternative. This can be done by bracketing the high and low extremes of the problem, and perhaps selecting a third to cover average conditions. Following this technique, equipment is selected on the basis of maximum and minimum number of seeks, etc. Thus, an initial specification might call for a 1401, 1410, and 7070 or 7074, but not a 1401 and 7094. To make these decisions, a number of influencing factors are considered: programming languages, standard applications, modularity, etc. They are discussed in the following paragraphs.

Programming Systems

Preference for a particular program language may affect the equipment decision, as well as implementation timing. Many programming systems require a minimum number of magnetic tape units, certain types of input/output units, a specific amount of core storage, and different kinds of optional machine features. For example, on a 7040, the use of full FORTRAN requires 16,000 positions of core storage. Where only four tape drives are needed to meet system requirements, but the selected programming system needs five to compile, five tape drives have to be specified.

Major language systems include FORTRAN, Autocoder, COBOL, Report Generator, Symbolic Programming System, but a variety of other one-to-one and specialized scientific languages may have to be evaluated. In some cases, this decision influences a choice of basic equipment -- for example, the 7010 provides a full-scale FORTRAN compiler, while the 1401 employs a "load and go" FORTRAN.

Other factors have to be assessed in reaching a language decision:

- Cost of compilation vs. the cost of programming, using experience for a related class of programs.
- Ease of programming and testing, and efficiency of both the compiler and object program. (Higher-level languages tend to lower run efficiencies, and are less effective when storage has to be closely conserved.)
- Efficiency and applicability of business and scientific-oriented languages. If the system comprises mathematical calculations in the main, FORTRAN may be appropriate; if it is input/output-dominated, then Report Generator and Input/Output Control System (IOCS) may be preferable.

- Availability of utility programs, and speed of sort programs.
- Presence of experienced programmers in the business. If experience level is high, Auto-coder and SPS can be used; otherwise, FORTRAN or COBOL might be easier.
- Open vs. closed shop programming. Where programming is to be performed by a group of professional specialists (closed shop), the more machine-oriented languages can be applied with less difficulty.

Standard Application Programs

The amount of programming can be reduced to the extent that standard application programs are appropriate. These programs are generally written for specific equipment (for example, the demand deposit program for the 1440-1412); therefore, their applicability is assessed in conjunction with the equipment selection process.

Modularity

Most large-scale systems are planned for gradual implementation over a number of years. This fact emphasizes the need for recognizing equipment modularity to minimize the transition to larger systems as the business grows and more activities are added. A family of equipment (for example, 1440, 1401, 1410, 7010) that provides load deck compatibility from small to large system offers clear advantages to an expanding business.

Input and Output Devices

Input and output possibilities examined in basic design can now be stated as specific input readers and punches, and output printers or other special devices with the consideration of these points:

- Time sharing
- Tape and card read speeds
- Printer speeds
- Number of print positions
- Number of characters per inch
- Tape rewind time; start and stop speeds

With serial systems, some of these decisions are better made after computer runs have been defined and described.

Processors

The central processor is selected on the basis of throughput time per unit; that is, the number of operations to be processed for a given period of

time is multiplied by average processing time and summarized. Peak and average volumes are factored into this calculation, and consideration is given to use of possible supplemental storage (for example, multiple 1301s to support a 1410 system).

Storage

Although there will undoubtedly be some reorganization within the system during run analysis to take advantage of storage capacities and arrangements, an initial designation on the number and types of storage units is made from data on the total number of characters to be stored, required access time, type of storage, and storage format.

Other Factors

Beyond the imposed constraints shown in the requirements specification (that is, limitations on rental or purchase costs, presence of installed equipment, management preferences, etc.), other factors must be kept in mind as they generally influence equipment selection decisions. Some of them will be covered by the requirements specification, others involve good sense and judgment. Selection decisions, for example, should recognize or account for:

- Ease and economy of implementation
- Need for system flexibility
- Growth potential of individual applications and the business as it relates to equipment
- Selection and training of personnel
- Customer attitudes as they have been influenced by advertising, sales appeal of equipment features, services offered by the manufacturer, etc.

Other Selection Techniques

This latter point deserves amplification. In the selection process, systems engineers are defining equipment configurations that will perform all necessary applications and activities under near-optimum conditions. However, this evaluation is not always conducted in isolation, and the systems engineer may be faced with comparisons made by management consultants, other manufacturers, or even the customer's own personnel. Two techniques are commonly used to make these evaluations:

- Feature-by-feature comparison
- Power factor comparison

Under the features method, a list of system characteristics and features is compiled and performance noted for each area of the system. The system with the most "pluses" is regarded as the best one. For example:

Feature	Equipment		Choice
	A	B	
1. Basic cycle time (microseconds)	11.5/char.	6/word	B
2. Card reader speed	800 cps	2,000 cpm	B
3. Printer speed - numerical	1,285 lpm	900 lpm	A
4. Edit instruction available	Yes	Yes	Even

This process is continued until all characteristics are covered. The method suffers from several shortcomings:

- The relative weight or importance of individual characteristics is not accounted for in the decision.
- Interrelationships among characteristics are not recognized.
- No provision is made for measurement against performance criteria, or other subjects influencing the decision (for example, programming languages).

Such a comparison presents an incomplete and inaccurate evaluation, and its use is due mainly to the ease of application.

The power factor technique is built on the assumption that the selection process can be simplified by ranking all computers on the market according to their relative power. The fallacy of this approach can be demonstrated through two statements: "the 7074 is 15 times more powerful than the 7070" and "the 7074 is equal to the 7070 in power". Both statements are valid under specific conditions; for example, the first one is valid in an engineering design application, and the second in a completely tape-limited application, such as a tape merge. Power factor considerations are more appropriate in an evaluation of the market for future data processing systems, and like problems.

Equipment Selection Decisions

As the preceding discussion suggests, equipment selection is a technical problem, and at the same time a sociological one. Sound, logical reasoning is necessary to structure the solution, but these results may have to be modified somewhat to meet the special conditions encountered in a business. A

systems engineer should never overcompensate for the personnel factors, though, since his first responsibility is to the specification of an equipment configuration that properly meets system requirements through full utilization of individual components and features, without overdesign. In addition, the specified equipment should permit economical growth, where this is important.

The final selection of such a system is not possible unless preliminary equipment decisions are refined technically through run timing and costing.

Run Definition

In a concluding step to equipment selection, two or three configurations are prepared for each design alternative. Relative differences among them are decided from detailed analyses of run times and associated costs.

First, the system is organized into computer runs which will afford optimum performance at minimum cost. Most activities have natural points of segmentation at which a reasonable separation can be made for the initial run timing. In a serial system, the natural breakpoints are:

- Input conversion and edit
- Sorting
- Master file updating
- Output editing and conversion

These points are slightly different for a real-time system:

- Input
- Transactions (batch, single)
- Inquiries
- Outputs
- Reports

Simultaneously with run definition, other considerations are explored, such as:

- Maximum and minimum equipment configuration. On small-scale systems, the minimum and the maximum configuration act as restrictions. On large-scale systems, a minimum configuration is usually established by the selected programming package (particularly the compiler), while core storage acts as a maximum restriction.
- Conversion of source data to magnetic tapes or cards, and on the output side, conversion of magnetic tapes to punched cards or printed output.
- Operational time limits (one shift, five days per week, etc.) and deadlines (daily, weekly, etc.).
- Application program minimum configuration requirements.

After run segmentation is completed, a preliminary run organization chart is drawn up to show the flow of information through the system, and the relationship of runs to files. The chart illustrates the magnitude as well as scope of computer runs.

Expanded Equipment Configuration

Identification of some equipment features and options may have been delayed until after runs are defined. Any features omitted for various reasons are now selected, including such options as:

- Processing Overlap
 - Advanced Programming
 - Compressed Tape
 - Tape Intermix
 - Tape Switching
 - Multiply-Divide
 - Print Storage
 - Additional Access Arms (1405)
 - Cylinder mode (1301)
 - High-Low-Equal Compare
- and, if they were not finalized earlier:
- Number of disk drives and packs
 - Number and speed of tapes
 - Printer line speed
 - Card readers, punches, etc.

Preparation for Run Timing

Each complete equipment configuration for a design alternative is subjected to a rough or detailed run timing and cost analysis, as required, to demonstrate which one offers maximum performance at minimum cost.

The preliminary file definition is expanded to establish:

- Restrictions on file format (for example, block size limitations)
- Record formats from core storage and magnetic tape unit characteristics
- Average record length of each file

An approximate passing time and designation of required tape reels can be calculated for each file on the basis of volume, anticipated tape density and a tentative blocking factor.

In relation to inputs and outputs, runs are generally defined in a descending order of total running time. Card inputs are translated into card read time by dividing card read speeds into volume. Tape assignments are made to achieve a balanced condition on channels, and consideration is given to error and exception routines. Checkpoint and restart procedures are needed in the longer runs.

Run Improvement

Once these preparations have been concluded, run design is reviewed to improve the overall effectiveness (for example, reducing the number of tapes, etc.) of each run. Opportunities may exist to combine short or split long runs, for example, to achieve greater run efficiency.

Run Timing and Costing

Utility runs, such as sorts and merges, are timed from published data. For other types of runs, tape passing and internal processing time (including core storage and tape interference time) are combined on the basis of system buffer characteristics. In an unbuffered system, run time is the sum of tape passing and internal processing time. In a buffered system, run time is the greater of internal processing and tape passing time on the channel with the heaviest load.

From run time data on the several equipment configurations, a decision is reached on the best equipment for a single design alternative. This process is repeated for other alternatives, and a final equipment solution is established by synthesizing these various results into the configuration most appropriate for the business.

Simulation

Other methods are currently being developed to aid in the task of system selection: run timing generators, simulation, etc. The use of simulation is particularly important in systems requiring several data channels, or involving a communications network which must handle queues of varying length.

Prior to conducting a system simulation, a careful statistical analysis of data flow, peak and average loads, and time limitations is made, and representative equipment selected to bracket the requirements. The system is then manipulated on a simulator designed especially for this purpose, or on the IBM General Purpose Systems Simulator. With successive passes through the simulator, system features are varied: number of channels, speed of channels, core storage size, queue storage size, number and speed of input/output terminals, etc. On completion of the simulation runs, an optimum machine configuration is proposed.

Although simulation can be of significant value in the equipment selection process for complex, large-scale studies, it cannot be used without recognizing:

- The cost of computer time to carry out enough simulations to obtain sufficient data.
- The amount of programming involved in writing a special-purpose simulator, or preparing logical block diagrams for a general-purpose simulator.
- There is no way to completely validate simulation results short of actual installation and operation of the system.

Personnel and Procedures in System Selection

Emphasis thus far in system selection has been concentrated on the equipment configuration. By definition, however, a business system comprises people, procedures and equipment. Consequently, to complete the task, there must also be a costing and timing of the human "configuration" and its associated procedures as required to operate the system once it is fully installed.

Salaries and clerical support costs are calculated from actual payroll data, or industrial averages, and projected over the useful life of the system. Combined with timing and cost data for the equipment configuration, this produces a total system operating cost, the end objective of system selection.

Documentation

As in basic design, the Activity Sheet can be employed as the principal documentation format. Each run is flowcharted in support of a general systems diagram. Volumes, costs, time relationships and other significant data are noted in the grid area of the sheet and cross-referenced to the diagrams. Description at this general level of detail is adequate for most studies, but when selective detailing is desirable, Operation, Message, and File Sheets can be used for detail data display. Titles, numbers, and salaries of job positions, and equipment features and prices are included in separate summaries.

APPRAISAL OF SYSTEM VALUE

The basic worth or value of a new system is determined in the final analysis by managers who are responsible for protecting and increasing a company's profitability, and providing a return on shareowners'

investment over an extended period of time. Managers appraise a system by how well it measures up to selected decision-making yardsticks:

- How will the system impact on profits and markets for the next few years?
- Are sufficient funds available to support initial installation, or does money have to be borrowed? If so, for how long?
- What is the economic worth of such general advantages as higher operating efficiency, greater flexibility, etc.?

An important task of the systems engineer is to find explicit answers to these questions, and establish an ultimate system payoff in terms of cost and the economic effect of time, efficiency, flexibility, etc.

To prepare a comprehensive economic analysis of systems value, four types of data are needed:

- New system operating costs projected over the useful life of the system.
- Implementation investment (cost) required to bring the system into full operation.
- Present system operating costs projected over the estimated useful life of the new system.
- Basic values which the new system offers to the business, expressed in economic terms.

Costs are then related to values to produce a total system impact on the business. Projected present system costs are used as a base for making comparisons. Therefore, if two potential designs have been carried through to this point, they are evaluated separately with the present system base and not with each other.

New System Operating Costs

During system selection, costs were compiled by computer run for the equipment and personnel configuration and associated procedures. These figures are now projected to encompass time periods up to the useful life of the system, recognizing anticipated growth and planned expansion in activity coverage. Major cost categories include equipment rental or depreciation, salaries and fringe benefits, materials and supplies, and related indirect costs.

Sales forecasts, implementation time schedules, and long-range business plans are typical sources for making this estimate. Since sales projections are subject to forecast error and variability, it may be desirable to state future sales at three levels: average, minimum and maximum expected volumes. When this approach is followed, related costs are

adjusted to comparable bases, taking into account cost improvement rates, salary differentials, and fixed and variable cost aspects. The step function characteristic of costs also has to be recognized in this compilation -- that is, the fact that computer systems can handle increases in volume at relatively minor increases in costs up to certain levels. As volumes build up progressively, costs often remain on a plateau, then suddenly jump with another slight volume increase. The timetable of implementation normally reveals when new activities are introduced into the system, and cost estimates are adjusted to reflect these successive additions to the load. In case no decision has yet been reached on whether to buy or rent the equipment, parallel investigations may be necessary to demonstrate the differing cost impacts.

Occasionally no adequate cost precedent exists, and special analyses are required to derive valid data. Statistical methods are valuable for collecting samples of performance, while the techniques of experimentation and observation can be applied in varying degrees of detail to special problem areas.

Implementation Investment

Although implementation is the second stage, a nominal amount of preplanning is performed at this point to indicate the magnitude of the implementation investment and its time impact on new system operating costs. Preparation of this estimate involves thinking through the subjects on which implementation decisions have to be made, and evaluating time and cost implications for each of them, without being concerned with the techniques or substance of implementation work.

For each of the five major implementation tasks -- detail system design, programming, physical installation, conversion and system test, and personnel selection and training -- an estimate is prepared showing how long each will take, how many trained personnel are needed, how much training each person will require, and how much it will cost. This general data is summarized in bar and line graphs, PERT network charts, or standard forms such as the System Pre-Installation Schedule.

The preparation of implementation time and cost estimates is discussed briefly below in the framework of each of the five tasks:

Detail system design.-- Estimates on time and cost are compiled for the separate design assignments: forms design and layout, detail file design, run book preparation, etc. Several techniques are employed to obtain data: sampling

representative routines and extrapolating results, using data from prior studies, applying empirical standards, or conducting small-scale desk tests.

Programming. -- Time and cost implications of coding, desk checking, debugging, and testing are estimated in a similar fashion. The detail design may have to be reorganized somewhat to facilitate efficient sequencing, as well as lap and gap phasing of the work. Total programming time can be reduced to the extent that generalized and utility programs and program testing aids are applicable.

Physical installation. -- Detailed guides are available on the various aspects of site selection and construction, air conditioning, equipment and office layout, and electrical and cable requirements. The main concern here is the provision for an adequate time schedule and sufficient funds, since this period can rarely be appreciably shortened.

Conversion and system testing. -- In the preparation of this estimate, a variety of assignments are included:

- Preparing and editing files
- Establishing file maintenance procedures
- Providing training in system operation
- Planning for pilot or parallel operation, etc.

A realistic schedule is constructed on the basis of the amount and type of work to be performed and the availability of personnel to handle it.

Personnel selection and training. -- The expense of selecting and training personnel is maintained separately from implementation personnel salaries. Additional time should be allowed for in the implementation schedule to interview, test, rate and train new personnel. Estimates for these tasks are then recompiled into an implementation cost and time summary.

Present System Operating Costs Projected

The basis for appraising the new system is an estimate of how much the presently in-place system would cost to operate if it were to be maintained for the same expected life as the new system and had to support the same levels of sales forecast. Included costs are the present personnel, equipment, supplies, and other indirect expenses that will be taken over or superseded by each new activity in the order it is introduced. Of necessity, the compilation cuts across conventional organization and functional boundaries to pick up expenses associated with an activity's scope and content, and also accommodate the progressive buildup of

activities in a system (that is, present system costs are not picked up until the comparable new system activity is planned for operation). Accounting classifications probably will not categorize expenses by these relationships, and costs may have to be built up through interview and estimates.

Balance sheets of financial operations, income and expense statements, and costs gathered in the foregoing phases are sources for this data. Expenses are extrapolated from current operations to the anticipated volumes of future years, taking into consideration the same factors (productivity wage increases, cost reduction trends, changes in material costs, etc.) used in the new system cost projection.

Economic Value of a New System

Aside from showing reductions in the cost of processing data, a systems engineer must demonstrate other positive benefits and translate them into economic value, however intangible they may seem. What is it worth, for example, to reduce finished goods stockout conditions from 1 in 4 to 1 in 20? This is a decided advantage to a business, and it can be converted to an economic value by showing how much sales are increased when a customer is able to secure an immediate shipment rather than turn to a competitor.

This same type of reasoning can be applied to each of the measurement factors outlined in the system requirements specification, and others added since then.

The impact of the new system on each factor is assessed and described, and an economic value is assigned to this performance. Examples of other possible factors include:

- Decrease in the length of a product or service processing cycle.
- Improvement in product or service quality.
- Shortened response time to inquiries from potential customers.
- Increased employment stability.
- Better promises kept on customer shipments.
- Greater stock availability to service a variable demand.
- Effect of cost reductions, elimination of spoilage, waste, and obsolete materials.
- Influence on other resources (for example, accounts receivable, inventories, utilized floor space).

To secure realistic values in some of these areas, such techniques as logical analysis may have to be applied.

The emphasis on and need for an extensive economic value appraisal differs from study to study. There are situations where the appraisal is not required at all, or a simple narrative description of values would suffice -- for example:

- Nonprofit agency
- The new system is imposed or is the only alternative (as when a management insists on an online system to meet competition)
- Cost savings alone justify the new system.

It is also advisable to seek out management's views on the degree of economic evaluation required; a ten-year projection is wasted, if three will do.

On the other hand, where savings are marginal, or represent only a minor part of true system value, economic analysis is the only alternative to fully demonstrate a system's capability for maintaining and increasing business profitability. When this is the case, the several cost and value analyses are drawn together and integrated into a total systems value appraisal.

Relating Costs to Values

A complete system value appraisal is accomplished by preparing three reports that will assist management in making decisions: profit and loss, return on investment, and a cash flow statement.

Cost-value relationships on the present and new system can be displayed in financial report style by preparing a "pro forma" summary of operations sheet for part or all of the business. The exceptions principle and variance ratios can be used to cut down the number of calculations. The projection can be further simplified by separating non-affectable costs out of the computation. This approach requires extensive data extrapolation, yet must be a reasonable and believable estimate of future conditions. Unless the impact of reduced cycle time and other factors can be accurately interpreted for effect on sales volume, sound judgment would dictate use of the same sales forecast for both the present and proposed system projections.

A statement of return on investment graphically shows the incremental value of a proposed investment in terms of earnings by measuring new and current system cost and investment differences. The value of inventories is an added dimension to this calculation beyond operating costs.

The third form of appraisal, cash flow, reveals the impact of the proposed system on the cash position of a business, and whether sufficient funds will be available to meet commitments as they arise.

Much of the information in an economic appraisal of system value depends on a proper interpretation of future events. It is advisable therefore to amplify how key comparisons were developed and what their significance is to the business. Major emphasis is placed on the areas of management interest -- that is, the time at which the new system begins to pay back on initial investment, and the magnitude of the savings once the system is in full operation.

SYSTEM DESCRIPTION AND PROPOSAL

In a final step of the design process, the new system is fully documented to:

- Provide management with an understanding of the system.
- Supply technical data for various specialists who will implement the system.

Since the report is the basis for making a system proposal to management, its form and content are a matter of prime concern.

Report Content

The sequencing of topics, amount of detail, degree of subject emphasis in the new system description are problems to be resolved by the systems engineer in accordance with his own special needs and attitudes. But whether the report is ten pages or 120 pages long, it should have direction and balance, and should communicate its message concisely, with a continual awareness of the reader's interest and viewpoint. The narrative should be supported by charts, graphs and pictures to explain points that are difficult to visualize.

Realizing that these reports will differ substantially in approach from study to study, the following outline of form and content is suggested as a guide to the systems engineer in preparing quality descriptions of the proposed new system.

Preface. -- The introductory section of the report contains a letter of transmittal to management with credits to study participants and contributors. Other includable items: index, summary of content, general considerations affecting the whole study, aims and objectives of the study, etc.

Management abstract. -- Since this report must cover a wide range of subjects, it is usually advisable to provide a summary in the beginning for the executive who wants to gain insight and understanding about the proposed system without poring over

technical details. Within this section, a manager must find the facts he needs to make long-range system decisions. Its content should reflect a manager's viewpoint of the activities concerned and include an appraisal of basic system values and advantages to the business. The abstract should be thorough in coverage, yet selective, present significant facts only, and be well written. However brief, it should address itself to these subjects:

- Recommended courses of action in regard to the new system.
- Objective appraisal of system advantages, benefits and savings.
- Recapitulation of results from earlier study phases.
- Highlights and features of the new system in operation. (A system flowchart in which technical symbols are replaced by artwork or photographs is one way of accomplishing this objective.)
- Review of the investment required during the implementation stage.

New system in operation. -- This and succeeding sections of the report supply more detailed descriptions of the proposed new system for other operating managers and technical specialists of the business. Here, the general system description from the abstract is exploded to show detailed information flow, performance data on the equipment, and organization of the personnel who will operate the system.

When preparing timing estimates for this section or the appendix, care must be taken to make them as accurate as possible. They should always be called estimates, and the report or proposal should state that they are based on such assumptions as normal operator efficiency, good supervision, normal work flow and sound programming and internal policies. If the timing estimate is not an estimate of the overall time to do the entire job, it should be so stated. For example, if the estimate is based on tape time only, this should be so stated in the report or proposal. If setup time is not included, it should be shown separately.

Similarly, statements concerning dollar or labor savings should be made only where there is a sound basis for them, and they should be made as estimates based on statistics furnished by the prospect, not as promises or "guarantees".

Implementation plans. -- Various implementation time and cost schedules are discussed in this section to provide operating personnel with information

Section	Area	Page
01	01	23

on the tasks to be performed, types of personnel who will perform them, estimated length of time each task will consume, and the cost of each task.

Appraisal of system value. -- The abstract outlined economic values of the system as they impact on the total business in terms of profits, cash flow, and return on investment. In this section, system advantages are examined as they will interest accounting, facilities, and property personnel. This includes:

- Cost comparisons between present and proposed systems over a projected useful life.
- Reasons for rejecting alternative approaches.
- Detailed descriptions of other system values, including such intangible benefits as flexibility, reliability, improved service, etc.

Appendix. -- This is fundamentally a technical validation section, containing a selected body of information useful to methods and programming personnel. Some of the includable items:

- Computer run descriptions and books
- Simulation details
- Record, file and message descriptions
- Physical planning detail
- Detailed equipment characteristics
- Disk and tape requirement calculations

Documentation emphasis is directed in particular to critical and advanced elements of the system.

Presentation to Management

The new system report is a concrete, firm system solution for a business. It may be used as a formal proposal to management, or the proposal may be drawn from the report.

The systems engineer usually works under the direction of a salesman in the preparation and presentation of a system proposal to management. Both may contribute to the writing of the proposal, each selecting sections for which he is best suited.

The completed proposal is transmitted to management for review and evaluation. In certain cases, particularly in small application or mechanization studies, this is the concluding task of study and design, except for periodic follow-up to determine the future course of action.

For the large majority of studies, however, the proposal should be supplemented with formal management review sessions to amplify and interpret various parts of the proposal. Consequently, when the report is turned over for management review, a request is made for an oral presentation of results to the several managers of the business who will influence the final systems decision.

Subject matter for oral presentations is concentrated on the important system features and values, and on solution areas that differ appreciably from the present system. Details should be omitted for coverage in specialized sessions or reading in private. Where possible, the presentation should stress a slightly different structure than the report, to avoid the tendency to cover the same ground and in the same depth. Extemporaneous presentations are more effective than memorized ones. The narrative should be supported with visual aids such as flip charts, slides, flannel boards, Vu-graphs, etc. Artwork should have a professional touch, but not be overelaborate.

Once the formal presentation is made, time should be allowed for discussion of the proposal. This requires thorough knowledge of study results and adequate preparation prior to the meeting.

Proposal review is a key event in system studies, and the need for careful preparation cannot be overemphasized. It becomes even more important in competitive situations. However, whether competition is present or not, the systems engineer must always ensure that his design meets requirements, and yet is not loaded with specials and extras. He should also maintain a continuous communication with management through contacts, progress reports, and perhaps the teaching of courses. These and other considerations are all part of the job of a systems engineer.

When appropriate, the systems engineer should draw attention in the report or proposal to the many services IBM provides to assist all its customers: training and education of customer personnel, existence of system support annuals and publications, experience in comparable studies, availability of expert consultation, system maintenance, etc. In pointing out these services, however, stressing "bigness", such as total number of customer engineers and systems engineers in the U. S., total dollars spent on schools, and numbers of similar installations in the U. S., should be avoided. It is fair selling, however, to state numbers relevant to the particular customer. For example, it is reasonable to expect a 1401 prospect to be interested in the number of customer engineers in the applicable branch office who are trained on the 1401; the number of 1401s installed in the particular city in which he is located; and the training schools available to him locally and in the region.

System Demonstrations on Equipment

Where the study is complex, or the design quite unique, it may be desirable to demonstrate selected parts of the proposed system running on equipment.

If the technique of demonstration is selected as the method of showing management how the proposed system will operate, the required programming and coding, arrangements for using a computer, etc., should be planned far enough in advance to prevent any lengthy delays once the study is terminated. The decision to prepare a live demonstration depends on the availability of the proper equipment at the proper time. This should be determined before a commitment to demonstrate is made.

Proposal Follow-up

After the proposal has been presented to management, a period of time elapses before a decision is forthcoming on the new system. In the interim, the systems engineer and salesman should make appropriate follow-up calls on the customer and, if the situation permits, conduct training sessions for customer personnel and begin planning for the several tasks of system implementation.

IMPLEMENTATION GUIDANCE

The ultimate success of a business information system is directly related to the amount and quality of the detailed planning that precedes its installation. The systems engineer is assigned a prominent role in performing the several "get-ready" task of system implementation, along with the computer programmer and the customer engineer.

Estimates on the timing and cost of implementation work were compiled earlier, principally to indicate the magnitude of the task to the customer. Now, the work elements of implementation are identified and described in greater detail. For convenience, they are grouped into five major categories in accordance with their general sequence of execution:

- Planning and progress control
- Personnel selection, training, and organization
- Detail system design and programming
- Physical installation planning
- Conversion and system test

PLANNING AND PROGRESS CONTROL

In view of the large number of tasks in implementation and their interrelationship, adequate plans and schedules must be prepared in advance to insure on-time installation. Adherence of progress to the implementation schedule provides earlier revenue for IBM and facilitates factory production scheduling; for the customer, it provides ease of installation and earlier realization of savings from the new system.

The systems engineer, when acting as the team leader in this effort, lays out the basic plan, administers it, measures progress to schedule, and modifies the plan as necessary to accommodate changes and slippage. In making up the job schedule, consideration is given to whether assignments can be performed sequentially, in parallel or overlapped with others. Usually a balanced effort is achieved in a minimum of time by employing all three methods in combination.

As implementation proceeds, the systems engineer will be called upon to demonstrate leadership in many other ways:

- Reporting progress to management.
- Motivating the team to above-average efforts.
- Maintaining good personnel relations on the team and with customer employees.
- Conducting team progress sessions to resolve technical and administrative problems.
- Showing confidence and control in troublesome situations.

Documentation Tools and Techniques

Several types of schedules and techniques can be employed to document and display implementation plans.

A series of forms referred to as General Pre-installation Schedules can be used to record overall progress.

Another standard form, the Applications Development Bar Chart, is used to portray planned and actual accomplishment by individual program name or number. Other forms are available for reporting programming progress in greater detail: Weekly Record of Progress, Record of Progress Chart, Program Development Schedule.

Personnel assignments and progress can be displayed on a Manpower Loading Chart. Assignments to projects are generally made in one or more of three ways:

1. Assign an application or program to one individual for complete follow-through. This is often the most practical approach for small applications.

2. Assign two or more individuals to develop a program as a team. This precludes dependence on one person's skill, permits completion in a shorter time, and makes several people knowledgeable on a given program.

3. Assign one person to defining and block diagramming a program, another to coding and testing. This method is useful when programmers have varying degrees of experience.

Any or all of these charts can be used to document the original schedule and designate progress. They should be prominently displayed in the work area of the implementation team, and posted daily. Charts are revised and redrawn periodically as the work is rescheduled, and to accommodate future progress evaluations.

Recent developments in the field of precedence diagrams -- for example, program evaluation review technique (PERT), critical path method, etc. -- can also be adapted to the broad planning of implementation jobs.

When using PERT, for example, three estimates (the most likely, optimistic, and pessimistic) are prepared for each event in a project, and recorded graphically in a network of interconnecting events -- some in parallel, some in sequence. Circles are employed to depict events; connecting lines show the relationships among events. Events are points in time, and time is considered after the events have been identified and linked in a total flow

Section	Area	Page
01	02	02

plan. The difference between this and other techniques is the ability to demonstrate a probability of completion for a series of events, and to define the longest (and therefore the most limiting) individual path to project completion. A complete description of PERT is outlined in the IBM general information manual "PERT -- A Dynamic Project Control Method" (E20-8067).

The degree and complexity of implementation documentation is determined by the specific requirements of the project at hand. For example, PERT could be applied exclusively, or it may be supplemented in certain areas with application or manpower bar charts. Other situations might require only general forms and charts. The important point to recognize is that implementation work must be thoroughly preplanned, organized, integrated and measured to schedule. Documentation format is a secondary consideration.

PERSONNEL SELECTION, TRAINING, AND ORGANIZATION

In the usual circumstance, implementation work is carried out by customer personnel under the guidance and direction of a systems engineer. He will, of course, perform selected assignments on his own. Experienced IBM programmers and other personnel may be present for a time, or even throughout the implementation stage. If a computer is being installed for the first time, personnel selection and training will consume a significant amount of time.

A systems engineer's responsibility may range from providing advice and counsel, to becoming directly involved in the training of personnel and assisting in the structuring of a data processing organization.

Recruiting and Selection

Personnel are recruited within the company, from data processing employment agencies, from universities and institutes of specialized training, or in response to direct advertisement. Job descriptions should be prepared in advance for each position, and the systems engineer should be instrumental in writing them. He can be of further service by helping to formulate selection criteria, regardless of who does the interviewing.

Selection is based on a person's interest, intelligence, education, background of experience, and aptitude for the work. Special tests designed for these positions can be applied to evaluate aptitudes. While these tests have shown fairly high correlation

between attained grades and subsequent success in the field, they should not be relied on exclusively.

Training

New and in-place employees need to be selectively trained once they are hired for the data processing installation. IBM offers a variety of courses on unit record equipment and computers for the executive, the DP supervisor, programmers, machine operators, etc. Other possible training sources should also be considered: nearby training institutes, programmed texts for use in self-development plans, and courses prepared and offered by the systems engineer on location. Arrangements should be made to complete these courses far enough ahead of the need in order to permit individuals to upgrade their skills through practice and application.

After the formal training has been concluded at IBM education centers and other places, on-the-job training should be encouraged to improve each person's proficiency. A tape librarian, for example, should be learning about library maintenance procedures and certain aspects of the system to be able to communicate properly with system operators and to understand the control forms that will be used. Similarly, the console operator should attend programming school and write at least portions of programs. He should also be operating the system during test sessions.

The education of offline personnel should not be overlooked. Individuals who will operate remotely located punched card and tape equipment require the same training as central operators. Other customer people not directly involved in the data processing operation, but affecting its function at either the input or output side, should be given orientation on the new system and equipment.

Organization

Data processing organizations differ considerably among users according to size, nature of the business, and management philosophy. A basic factor influencing organization structure, for example, is the degree of centralization or decentralization in a company. Centralization is often associated with high-speed, large computers run on a closed-shop basis (that is, only professional programmers from the central data processing component are permitted to program the machines, and all jobs must be channeled through these individuals). On the

Section	Area	Page
01	02	03

other hand, there are many successful installations where equipment is centralized and system planning and programming responsibility are assigned to user components. For instance, if there were three payrolls from three different divisions to be run on the same processor, they could either be designed and made compatible by the central group, or designed and programmed independently by the three user divisions with the central group providing commonly needed techniques or programs -- for example, sort and merge routines, label handling, restart procedures, etc. Nevertheless, certain rules apply to all organizations:

- Authority and responsibility should be defined and delegated.
- Relationships between the data processing and functional organizations should be specifically outlined to prevent overlap and possible conflict.

This latter point cannot be overemphasized. Information moves horizontally through a business across departmental lines, but most companies are structured vertically into functional groups (sales, engineering, accounting, etc.). Because of this organizational paradox, the data processing department should report to some individual at the first level of management -- a comptroller, vice-president, manager of communications, or the equivalent. Normally, the data processing manager will have direct authority and comparable responsibility for planning applications, recommending equipment, and installing, operating and maintaining the system. He will also have the usual responsibility for budgets, personnel selection, and communicating status to management.

The data processing manager has less direct control over operations and programming, and devotes more time to administrative duties. System efforts may be conducted by teams consisting of systems engineers, programmers, and functional specialists assigned to a particular project. In this way, application and computer knowledge is effectively combined.

Many variations on these approaches are possible. Over the short term, the systems engineer is interested in developing the best combination of talent to accomplish system implementation. When this work is completed, however, the people involved in it will probably remain with the system during full operation. The systems engineer must therefore make his initial decisions on organization in such a way that they will work out best for the business over the long term.

DETAILED SYSTEM DESIGN AND PROGRAMMING

A critical task in implementation is the preparation of detailed procedures for the new system for use by the programmer. The more effective the communication between system design personnel and programmers, the better the results, since good design provides a sound basis for good programming. While the responsibility of programming, wiring control panels and operating the equipment belongs to the customer, IBM is responsible for providing technical guidance and education.

Data Sources

Initially, available data from the study and design stage is reviewed. The system requirements specification will show what inputs, operations, outputs and resources are required, along with data on peak and average volumes, frequency, rate, number of characters to process, etc. Requirements could have changed in the intervening period, and are checked once more for validity and completeness.

The new system description or proposal contains a system flowchart of the entire job, and if a run timing and cost analysis was performed, descriptions of each of the runs are available, at least in preliminary form. Where messages, files and operations have been described in considerable detail, this documentation is extracted as source data for detailed design.

Run Development and Refinement

It is entirely possible that some of the necessary data is not available, or that runs have been defined in a preliminary manner and require further attention. Even if run descriptions are relatively complete, they should be re-examined from several points of view to see how they can be improved. Detailed design is an iterative process; runs are analyzed internally and as a group, and progressively refined through successive evaluations.

The first step, then, is a complete and detailed review of outputs, inputs, operations and files for each activity to ascertain the completeness and accuracy of the requirements. The system flowchart and requirements specification serve as background information for this review.

Next, run definitions are examined for content and balance. Opportunities will exist to split or combine runs, or reorganize them into more effective designs.

Following this, runs are documented in considerable detail at a procedural level of description. System flowcharts and program flowcharts are subjected to intensive study to determine how they can be improved. Run refinement can be illustrated through the example of an accounts receivable application.

Case Study in Run Improvement

The following example briefly shows how alternatives are introduced to modify the original design and elaborate on its quality.

Basic case data:

1. Computer: 1401 Model C4, 8K, Advanced Programming, High-Low-Equal Compare, with a 1402, 1403 (Model 2), and four 7330 tape units.
2. Application requirements:
 - Accounts receivable update program
 - 60,000 customer records on magnetic tape
 - 1,000 cash receipts, 1,500 new invoices, and 200 miscellaneous transactions daily, entered by punched cards
3. Record size:
 - 150 characters customer data
 - 50 characters customer balance
 - 20 characters per open item (maximum of 5 per record)

The first run was designed to read and write a master file containing 60,000 unblocked, fixed-length records of 300 characters, read 2,700 card transactions, and process the inactive records and transactions. Run time in total: 94.4 minutes. The bulk of this proved to be tape time.

Based on these results, several decisions were made:

- Block the customer records to a maximum of 1,200 characters.
- Use a blocking factor of 4.

With these changes, run time was recomputed for: read and write a master file of 15,000 fixed-length records of 1,200 characters, read 2,700 card transactions, process transactions and the inactive records. New run time: 63.7 minutes. While tape blocking reduced tape read and write time by one-half, process time increased substantially, because of the record movement to and from work areas.

It was established that about 3% of the customers would be active on any given day, and this suggested a "change tape" approach where only the active records would be written out instead of the entire master file. Another run was designed to read the same 15,000 blocked records in the master file, read and write a "change tape" of 450 blocked fixed-

length records of 1,200 characters, read 2,700 cards, and process transactions and records. Run time: 35 minutes.

At this point, the run was studied from a different viewpoint: a split file approach -- separating the master information from the open items. The approach was analyzed in detail, and the run redesigned as follows: read and write the master and open item files, read 2,700 transaction cards, process transactions and records. Run time: 23.6 minutes.

With split files, the addition and deletion processing becomes more involved, the ability to handle record inquiry more difficult, but the total run time is quite a bit shorter. Again, this is not intended to be a solution to an accounts receivable routine, but rather an illustration of iteration in the run design process.

From this example, some conclusions can be drawn in regard to the work:

- Run design is based on the presence and analysis of a substantial amount of valid data. As each run in the case example was developed, it was necessary to dig deeper into the make-up of the customer file to find information supporting each of the approaches.
- Run design requires applied imagination. Knowledge of requirements and equipment must be blended with alternative courses of action formulated by the systems engineer.
- Run design requires judgment to determine when the point of diminishing returns is reached. The systems engineer has to decide when he has expended enough effort on design and move on to other matters. Not all designs need to be timed; alternate approaches can often be dismissed by mentally estimating costs and comparing them with savings.

Run designs are made relatively final at this point in time, in anticipation of the programming effort. However, there is still room for improvement as the systems engineer gains and applies new knowledge of the system.

Treatment of Exceptions

A systems engineer may be called upon to decide whether to handle unusual and special transactions as part of the main-line design, or whether they are to be treated independently -- that is, whether exceptions and infrequent occurrences should be subject to manual or automatic processing. Complete handling of exceptions might impair economical processing of volume transactions, yet they cannot be ignored. Some middle course is usually followed.

In planning for exceptions, impact on peak load conditions is appraised, and routines established for their processing, automatically or manually. If they are to be handled offline, the routine states why they are to be processed manually, how they are to be processed, and when they will re-enter the system in subsequent runs.

Checkpoint-Restart Procedures

Checkpoint-restart procedures are useful in lengthy programs to return the system to some prior point where processing is known to be correct, and restart from that place rather than from the beginning.

Checkpoint records are needed to restart a program, and are established through standard sub-routines. This routine stores the content of registers, accumulators and indicators, and keeps track of the record being worked on for each tape reel. When a checkpoint is taken, the entire storage is dumped on tape, and this record contains all data necessary to reinitiate the run. Checkpoints may be placed when every end-of-reel condition occurs, where totals are to be crossfooted, etc.

A program restart routine reads in a checkpoint record and reinitiates the run which has been interrupted for some reason (power failure, damaged tape, operator error, introduction of another program, etc.). The program alteration switch can also be used after this latter condition. Other techniques (such as address stop) can be applied to stop the program at a predetermined instruction.

Emergency Bypass Procedures

It is advisable to have standby procedures to anticipate the possibility of a system being totally or partially inoperative, irrespective of the reason. Since they are used infrequently, they should not be elaborate, but should reflect the customer's specific needs. Bypass procedures may consist of:

- Working arrangements with another customer using compatible IBM equipment.
- Manual routines.
- Partial processing during inoperative periods.
- Switching of components (for example, card and tape) in the event of partial inoperation.

An acceptable time-delay factor should be calculated to establish how long the system can be down before the bypass procedures are instituted.

Accounting Controls and Audit Trails

Accounting controls are incorporated in a system to provide checks and balances on record accuracy, and to maintain proper divisions of responsibility and accountability. The degree of control is a matter of insurance: how much one can afford to spend for protection.

Actually, data should be controlled from the time it enters the customer's premises until it is disposed of in final form. Accounting controls for the data processing system begin with the edit of input transactions, and consist of hash and batch totals, card or transaction counts, etc. During processing, control is evidenced by record and transaction counts and subtotals; after the run, accounting control totals are reconciled to input control totals. Controls for operator errors, programs and tapes are discussed later under system operation.

An audit trail is a retraceable path of data flow, or as one firm describes it, "the means by which details underlying summary accounting data may be obtained, and the method of locating documentary evidence."* An audit trail, then, consists of a path of data and the method for locating details.

Audit trails are needed for a number of reasons:

- Customer service (transaction detail)
- Checking discrepancies
- Legal
- Internal audits
- Means of updating master file on a file reconstruction procedure

They are appropriate for applications having a financial impact on the business, or where the volume of data is so large that rehandling is not practical. Audit trails are less frequent in many engineering problem applications, or where data volume is low and readily available for checking.

The following are factors to consider in constructing a good audit trail:

- Details supporting a summary are readily available.
- Details are retained only as required for legal, audit and operational purposes.
- Stored data is miniaturized (Microfilm) or summarized consistent with requirements, in order to reduce maintenance and space costs.

*"The Auditor Encounters Electronic Data Processing" (F20-8057) and "In-Line Electronic Accounting, Internal Control and Audit Trail" (F20-2019), both prepared by Price Waterhouse and Company.

- Data processing department audit trails are made to coincide with those of the auditors, where possible, to avoid overlap and near-duplicate trails.
- Data sequence is maintained by the control field most frequently in demand (or some compromise of several demands).

Preparation for Programming

After an activity or application has been completely defined and described in terms of computer runs, the runs are translated into operating programs. Since runs are developed through a continuous process of refinement, good working relationships and constant communication between systems engineer and programmer are a vital necessity. Only in a climate of harmony and understanding can these two individuals achieve an effective total effort with a minimum expenditure of time and money.

The selection of a programming language probably was specified earlier as a factor in equipment selection, but if the decision was not made then, it must be made at an early date in implementation, from such alternatives as COBOL, SPS, FORTRAN, Autocoder, etc. In a like manner, the capabilities of generalized programs are appraised. For example:

- Assemblers and compilers.
- Input/Output Control Systems.
- Report Generators.
- Sort and merge programs.
- Utility programs.

Assignments for the various programming work elements -- flowcharting, coding, debugging, and testing -- are reviewed and handed out in accordance with the established plan (that is, the work elements are split up among the programmers, or they are given responsibility for completing an entire program through all work elements). Sometimes it will be advisable to push one or two programs through to completion well in advance of the others to demonstrate how key parts of the system will operate. This is a good morale builder when programming consumes large amounts of time, and progress is not too evident over extended time periods. Also, less experienced programmers should be assigned to the easier problems until they build up experience and background.

Flowcharting

The procedure by which data is processed is graphically represented in a program flowchart. It permits the programmer to visualize the sequence of arithmetical and logical operations, and clarifies the relationship between one part of a program and another. Program flowcharts are drawn up to such a degree of completeness that other individuals can follow the description without trouble and code the application, since the original designer may not always be available to interpret it. Neatness is not a prerequisite in the initial diagrams and subsequent modifications, but after the programs have been written and the program flowcharts are firm, they are redrawn neatly for inclusion in the run book. Finished flowcharts are sectioned in the loose-leaf run book so that program steps are related to the referenced part of the flowchart on any two facing pages. After program test, flowchart pages are labeled with the location of the beginning instruction for that part of the program. Flowcharts should be updated on a continuous basis as changes occur.

Flowcharting documentation can be supplemented by preparing decision tables, particularly for developing and representing the logic of a system. A decision table* reveals graphically what courses of action are taken, based on any given condition or combination of conditions. As in flowcharting, choice of language and symbolization are standardized wherever possible.

Programming

Programming is started after runs have been rigorously defined and record and input/output formats are finished. The sequence of steps to take in programming the job and length of time to allow for them are determined by the systems engineer. For example:

- When the programming job is so large that it must be split among many people, the systems engineer should develop plans ahead of time to reduce the amount of reprogramming when all elements of the program are merged (by segmenting the program and assigning blocks of numbers to each portion, controlling the use of program statements, branching routines, etc.)

*IBM general information manual "Decision Tables -- A Systems Analysis and Documentation Technique" (F20-8102).

Section	Area	Page
01	02	07

- When the decision logic for a system is quite complex in relation to input/output design problems, programming might be carried out as follows:

1. Concentrate first on the programming and debugging of the logic, keeping output instructions in the simplest of formats.
2. After testing the logic programming, re-write the input/output instructions.
3. Retest the new input/output instructions.

- Unless programmers are experienced, the principle of the learning curve will apply to the speed at which this work is accomplished; that is, later programs will be written faster and contain fewer errors than earlier ones.
- Even with proficient personnel, programs will be reviewed and rewritten to incorporate modifications that will reduce processing time.

Whenever possible, programming practices and procedures should be standardized. For example:

- Set aside storage for utility and library routines, and for subroutines useful in several programs (such as end-of-job and error routines).
- Program halts and initialization routines are standardized. (Initialization refers to the first group of program steps executed in any processing run to set up internal start conditions. Two types are required: one to clear storage and start the system before loading a program, another when program restart is necessary. This latter routine performs such functions as reset program switches, modify instructions and temporary storage areas to proper values, rewind tapes, turn off tape signals and input/output indicators.)

Many subroutines are available from IBM; others will have to be written to promote further programming standardization. Frequency of usage determines their relative priority and location in storage and on tape.

Documentation

The mass of information developed in programming can be communicated to operating personnel only if it is thoroughly and systematically documented. To avoid misunderstanding between line and data processing personnel, to train new employees, and to aid in program maintenance, a detailed run manual is prepared for each program in an installation. This manual, usually contained in a large three-ring binder, serves as the official instruction and

complete description for a program, and is developed concurrently with problem definition and programming. A complete checklist of contents is outlined in a later section of this handbook.

Sufficient detail to operate a program at the computer console should be maintained in a console run book. This reference manual is composed of information such as operating instructions, record layouts, and control cards excerpted from the master run manual. A console run book (binder or small folder) can be prepared for a single run, a series of runs, or even a complete activity.

In addition to the run data, there should be a documentation of the procedures, coding, forms, and instructions that are external to the data processing system and peripheral gear: for example, the handling, routing, and control of mail; procedure for packing and shipping stock; etc.

All documentation (record layouts, program listings, operating procedures) is maintained on a current basis, and reflects changes as soon as they are incorporated.

Standardization of Practices

The establishment of standards in practices, policies, records and instructions is a major factor contributing to the efficiency and ultimate success of a data processing organization.

A standards manual should be maintained by the data processing manager or someone delegated with the responsibility. It contains policies, copies of procedural forms, and instructions which govern the operation of the system.

Record control is one example of an area that requires standardization of practices, especially where one file is processed in more than one computer run, or where more than one programmer uses a single file. Responsibility for controlling data records in these situations should be assigned to one person.

Other specific examples of possible standardization include:

- Format of run manuals for each program
- Symbols used in flowcharts and block diagrams
- Coding method
- Macro and/or subroutines
- Utility programs
- Magnetic tape file and punched card retention policies
- Program desk-checking procedures
- Tape-labeling procedures
- Program restart policies

- Standard programming techniques
- Allocation of index registers, alteration switches, storage locations, etc., for specific purposes (for example, it is often desirable to locate error subroutines in the same storage, or memory locations for every program)
- Tape error-correction routines
- Typewriter messages and codes
- Stop addresses
- Tape drive addressing
- Symbolic programming labels or tags
- Standard system controls
- Standard options on sorts, merges, assembly routines, etc.
- Tape-handling rules
- Machine-scheduling policies
- Personnel practices in the machine room
- Personnel duties
- Program and tape library rules

Preparations for Program Testing

However experienced the programmer, or efficient the coding technique, newly written programs will contain errors. The customer is allowed a specific amount of test time at data centers, selected IBM plants, or education centers to debug programs before converting to full operation. Of course, if comparable equipment is already located in the business, or is accessible in a nearby plant, it might well be used for the same purpose, upon agreement between the two parties. To secure the best utilization of this time, a certain amount of pretesting is carried out, and complete preparations are made for the tests prior to going to a data processing center.

Program Pretesting and Desk Checking

Program logic is desk-checked initially by following sample data through the program flowchart or coded program. This sample data is the forerunner of actual test data. Desk checking may be either manual or machine-assisted. Unit record equipment, for example, can be applied to uncover clerical errors when set up to indicate the presence of blank columns in program cards, missing zone punches, erroneous double punches, etc.

Test data preparation requires the same care as writing the program itself. For the first test, only a small amount of data is needed, enough to check out various routines over a range of conditions. Data records are reviewed to insure that they contain no unexpected errors. Later, when the main

parts of a program have been checked, deliberate errors can be introduced to determine how the program handles them. Ultimately, much larger data samples may be withdrawn from files and processed through the system. Results are precalculated to verify those derived by the program.

The program is audited to isolate possible clerical errors such as:

- Branches to unprogrammed routines
- Incorrect or omitted indexing
- Uninitialized temporary storage areas
- Missing constants

The program is then keypunched, verified, and prelisted on a computer or accounting machine. Multiple listings are prepared in the following sequence: page and line, label, operation code, and operand. Lists are checked for:

- Punctuation (commas, decimal points, plus and minus signs, etc.)
- Spelling of mnemonic operation codes
- Duplicate labels
- Spelling of labels when used as operands
- Keypunch errors

If a symbolic language is used, many errors are detected when assembling the program. Corrections are incorporated into the source-language card deck, as well as the object program deck. Reasons for the correction are noted on change cards for the object program deck. Where the number of corrections is excessive, the program should be reassembled before testing.

Additional listings may be prepared as follows:

- Halt listing, specifying page and line, label, assembled storage location, and action required for all programmed halts.
- Switch listing, specifying page and line, assembled core storage location of switch setting instructions and program switches (including explanation of on and off conditions for the latter).
- Index work listing, specifying label, assembled storage location and explanation of contents for index words or registers; also page and line, label, and assembled storage locations of index word modifying instructions.

Programming personnel should be familiar with the utility programs used in testing; these include tracing a program's operation, printing of storage, printing of tapes, writing storage on tape, clearing storage, and program loading. Many automatic testing programs (for example, Auto-Test, Programmed Automatic Testing, etc.) incorporate these features. These programs are valuable for

Section	Area	Page
01	02	09

reducing test time, and for obtaining the documentation necessary to debug a program. When decision tables are included in the documentation format, tables can be desk-checked either separately to isolate errors quickly, or in groups to facilitate merging. If machine time is available but at a premium, writing may be overlapped with testing; that is, a few tables are written, and while they are being tested or waiting for test, others can be written or desk-checked. Diagnostic programs can also be employed to speed up the identification of errors.

Administrative Preparations

Responsibility for coordinating the test effort should be assigned to customer personnel. This involves deciding on the:

- Sequence of programs to be assembled or tested.
- Number of tests per program.
- Proportion of the time to be devoted to training personnel.
- Determination of the test load for a given time period.
- Designation of assignments during test: console operation, tape and card handling, and peripheral operation.
- Point at which console debugging or multiple program testing should be terminated in favor of the next test.

A set of console and operating instructions is prepared for each program to show:

- Setting and meaning of each sense or alteration switch on the console.
- Number and density of each magnetic tape unit.
- Tape unit number on which each tape reel is to be mounted, whether the reel is to be file-protected or not, and disposition of each reel upon completion of test.
- Storage location, console indication, program status, and action required for each program halt.
- Sequence of cards in the reader (clear storage, bootstrap, program variable dates, control and detail cards, for instance).
- Type and disposition of cards in the card punch stacker or stackers.
- Form type and carriage tape to be used on the printer, and disposition of the output.

A program package being readied for testing includes:

- Detail flowcharting
- Source language coding sheets

- Source language card deck
- Object program card deck
- Test data
- Listings as specified
- Console and operating instructions

Any necessary control panels are wired in advance, if possible, and brought to the test center already checked out. Wiring diagrams and narrative descriptions of the wiring accompany each panel.

Testing at Data Processing Centers

Formal testing takes place after preliminary program checking is completed, and suitable arrangements are made with the data processing center for blocks of test time. In some instances, this work can be conducted remotely by locations offering this service. However, the customer should have a solid background of previous test experience and should have developed good test habits and techniques before sending out programs to be tested at a remote location.

Whenever practical, the first test session at the data processing center is devoted primarily to an orientation on the equipment, particularly the console. Simple programs are assembled and tested to familiarize personnel with the machines and testing techniques.

Less complex programs are processed first to provide experience for handling more complicated ones. A larger number of programs are scheduled than time permits, since a priority program may need further revision and be pulled out, or others may be debugged faster than planned. Additional programs reduce the tendency to make hurried corrections in any one program in an attempt to maximize test time.

As the work is scheduled over a series of sessions, succeeding programs normally should not be made dependent on results of earlier programs, in case trouble is encountered. This is particularly significant where other systems are involved, as in the example of using an IBM 1401 for input/output to an IBM 7070. Where dependency must be recognized, input test data for the dependent program should be prepared separately, as if actual results were being received from the first program.

During the test, console and operating instructions are followed explicitly. At the termination of a program test, console status is noted (if other than end-of-job), and output marked with program identification, date, and time of test.

As errors are identified, the documentation is revised as necessary: object program deck, assembled program listing, console and operating instructions, source language card deck, etc. Obtained results are compared against

desired results. Output is then utilized to establish where errors exist; subsequent effort is concentrated on these areas only. When an error condition is located and corrected, detail checking continues from that point on to insure that the correction does not cause other errors.

Although relatively simple programs are processed at the outset to promote training, emphasis should be shifted rapidly to the most important programs in the system. Only in this way will the key programs receive sufficient attention through retest to ensure that they are properly debugged. When testing is well along, individual programs can be linked together and tested under full system conditions.

A major objective of the testing process is to produce trouble-free programs, but appropriate attention is also directed at other objectives, such as:

- To increase programmer and operator proficiency.
- To develop good test habits and techniques for carryover to system operation.

PHYSICAL INSTALLATION

The task of physical installation includes:

- Selection of a site for the new data processing system.
- Construction of the room or building, taking into consideration floor loading, type of floor, ceiling height, number and dimensions of entrances and exits, acoustics, viewing area, building and fire codes, etc.
- Provision for air conditioning, temperature and humidity recording equipment, etc.
- Layout of the area to accommodate equipment, data files, working space, potential expansion, etc.
- Specification of electric power, lighting, and cable requirements.
- Selection and ordering of office equipment (files and storage cabinets for printed forms; cards, magnetic tapes and control panels; desks, chairs, work tables, carts, communications equipment) and any other secondary equipment.

Since the physical installation cycle cannot be appreciably compressed, an adequate time schedule and sufficient construction funds must be provided early in the implementation stage. Beyond basic construction or modification investment, site cost is influenced by whether a customer wants an elaborate installation or a basic setup.

While the main responsibility for this work falls on architects, contractors, and IBM customer engineers, the systems engineer makes specific contributions on his own, such as in the preparation of plans and schedules (reflecting necessary lead times) for the performance of the various jobs.

Physical installation manuals are published by IBM covering system installation requirements, both for the general and the specific equipment case.* They should be studied and distributed to personnel involved in this work in advance of schedule preparation.

CONVERSION AND SYSTEM TEST

When the equipment is installed, data processing personnel must convert the present business to new system procedures and routines rapidly and smoothly. Some preliminary conversion planning was carried out back in system design, but detailed schedules and assignments are drawn up about the time programming begins. The scope of the conversion task is established when a decision is made on how large a part of the total system is to be included in the initial changeover, and how much is to be phased out for progressive implementation in future time periods. Thus, conversion may encompass any of a number of situations, depending on the ability of the staff to handle the work; it may involve a dominant activity of the business, a series of slightly improved applications, or a few terminals in a complex TELE-PROCESSING network. Progressive implementation is a practical alternative to accepting too big a job at the outset. This approach is particularly useful when the size of the conversion team is limited and the same activity is to be installed in several different departments, branch plants, district warehouses or offices -- for example, a multi-plant payroll application, or a stock control procedure for all product distribution centers. In this way, the conversion load is leveled, and new applications are picked up as quickly as the personnel can handle them, and as the equipment is built up to accommodate them.

Conversion Planning

Regardless of the strategy, a realistic time schedule for conversion is prepared, based on the amount of work to be accomplished and the availability of equipment and personnel to perform it. Extra personnel or equipment may be needed on a temporary basis to complete conversion preparations. The responsibility for converting portions of the system

*IBM general information manual "Physical Planning" (F24-1052-0) and other IBM publications.

Section	Area	Page
01	02	11

is assigned to specific individuals and closely coordinated to the total schedule. Some required steps in conversion are:

- Gathering data for master files.
- Editing files for completeness, accuracy and correct formats.
- Consolidating files, creating new files, and determining file maintenance procedures.
- Identifying the equipment (that is, punched card or computer) to be used for editing.
- Providing training and instruction for machine operators and personnel in departments supplying source data and receiving processed data from the system. These people should be advised on all changes impacting their operations.
- Determining what additional programs are required for conversion.
- Establishing schedules for cutover to the new system.
- Coordinating the actual conversion process.
- Measuring the accuracy of the new system.
- Comparing results of the new and old system to each other.

According to preference, one of two approaches will be followed at the time of conversion: parallel or pilot operation.

Parallel Operation

Under the parallel approach, current data is processed simultaneously on the old and new systems. Operations are usually continued through one complete cycle of processing (for example, one billing period). When one part of the system is operating successfully, another is put into parallel processing. Some duplication of personnel, space and equipment may be necessary for a period of time. Full system operation in parallel is not always required, if segments of the application or activity (for example, the first few of a series of branch offices to be included in the system) are representative of the whole.

Pilot Operation

With the pilot approach, the new system (or selected parts) is put into test operation on a full-volume basis. Results are checked and required adjustments are made before the new system is declared operational.

Pilot processing verifies the entire operating performance of an activity or application. This

was not possible in program testing when programs were checked out separately. Pilot processing also provides extensive training for user personnel under conditions they will encounter in actual operation, including the preparation of input data.

The exact method of pilot operation is influenced by existing conditions:

- If the application has not been performed previously, the pilot run is conducted under controlled conditions with sufficient checking to assure correct results (manual checking, editing, and balancing of data volumes are required in this and subsequent situations).
- If the application was manually processed, a selected past time period should provide volumes and results that are representative of current policies and procedures. Considerable effort will have to be expended in the preparation of master files and input/output data, and in the conversion process itself.
- If the application was previously processed on punched cards, conversion will be somewhat easier, but data has to be prepared by editing, checking, and combining files, then developing conversion procedures.
- If the application has been processed on another computer system, files should be in fairly good condition for conversion. Changes in logic or method must be considered. Where the two systems are compatible, a simulator program might be used in the conversion -- for example, simulation of an IBM 650 on an IBM 7070. Simulators are used only as a temporary measure, however.

Occasionally, neither parallel nor pilot operation is applicable (as in an entirely new systems approach); in these few cases, a method is devised to meet the specific need.

System Test

The above operations test the new system by verifying that programs are performing as intended, both individually and in groups. Even though the equipment is thoroughly checked before being turned over to a customer, these operations also serve to verify the reliability of the equipment under near-operating conditions. Two other techniques are useful in testing equipment:

- Diagnostic routines designed for testing purposes.

- Complex programs prepared by the customer from actual data, requiring use of all components.

Specially designed programs for checking out each feature over a range of variable situations that simulate unfavorable conditions (marginal checking) may be appropriate.

Planning Guidelines

As in physical installation, the presence of and adherence to a comprehensive conversion plan will pay dividends. With such a plan, conversion will proceed with relative ease; without one, conversion could turn out to be a difficult, troublesome task. Certain guidelines are helpful in preparing the plan:

- Keep management informed. Establish a regularly scheduled meeting at which implementation progress is discussed with the customer executive responsible for the installation. IBM and the customer should be

informed of conversion progress, since both have much at stake in its success.

- Allow sufficient time. The amount of work involved in this task is often underestimated.
- Convert during periods of low business activity. In most businesses, the volume of data varies, because of seasonal or other demands. Where this characteristic is evident, it may be desirable to convert when data volume is low.
- Phase the conversion. Too large a load in the beginning only causes more conversion problems. Whenever possible, conversion should be carried out progressively over extended periods.
- Prove the files. For programs to run effectively, files must be free from error.

When conversion to the new system is completed, implementation ends, and a period of full-scale, routine operation begins.

OPERATION SUPPORT

From the standpoint of time, the operations stage is the longest period in the life cycle of a business system. Once the system is in full or partial operation, there is a continuing need to modify, improve, and measure performance to maintain the system at peak efficiency. The operations stage, then, should be a dynamic rather than a static period, during which many changes take place; new ideas and techniques are applied, programs are rewritten as languages are improved, new equipment models replace older, obsolete units, personnel are retrained, and the organization expands to handle new applications, etc.

The systems engineer is less directly involved in this process than he was during study and design, but he can make a significant contribution by the way he anticipates the problems of routine operation and establishes or recommends plans for upgrading the system. He must, however, be careful not to overstep his authority and take direct action in areas that are more properly the province of the data processing manager or IBM salesman and customer engineer.

This part of the handbook describes the plans that should be formulated and executed to keep a system operating at a high level of effectiveness: schedules, console operation, magnetic tape control, program control and maintenance, performance measurement, growth and expansion, and personnel considerations.

OPERATING SCHEDULES

In view of the multiplicity of applications, programs, main-line and peripheral operations going on at any one time in a data processing section, operating schedules should be prepared well in advance of the time the work is to be performed.

Scheduling Practices

While schedules should be sufficiently tight to guard against idle machine time, they should allow for setup, manual operation in case of errors, and unavoidable delays. Setup time, in particular, is closely scheduled to reduce the number of tape reel and card deck changes, but should provide for some unexpected changes during processing. When many small runs are being processed, setup time between jobs will account for a high percentage of the total time consumed.

Monitor, schedule, supervisory and executive routines can be used to reduce setup time. One approach to the problem is to maintain a program master tape on a daily basis in the sequence of the jobs to be run; call cards are used to pull in the appropriate program. Test, utility and sort programs can then be maintained in card form or on separate tapes.

Specific times should be set aside for program testing of new applications and the maintenance and improvement of existing programs.

Peripheral Operations

An important element influencing required buffer time is the scheduling of peripheral or auxiliary operations -- that is, machine operations on data preceding and following the processing of data by the main system.

Peripheral operations are scheduled in close conjunction with the planning for the main system, since a change in one will very likely necessitate a change in the other. Three areas of peripheral operation are included in the scheduling:

- Preparation of input data: punching, sorting, collating, listing and editing cards; converting cards to magnetic tape.
- Temporary operation of file conversion where files are prepared and entered into the system.
- Disposition of output data: same operations as for input, as well as magnetic tape to card punch and tape-to-printer operations.

Planning of peripheral operations includes writing of procedures, computing workloads, timing of operator and machine functions, wiring and testing control panels, and test deck preparation.

Documentation

After the sequence and timing of peripheral and main-line operations have been determined, a schedule is prepared and documented on an operating schedule form. Time is recorded across the top; peripheral and main system components are shown on the left side. Solid horizontal lines across the chart indicate the time machine units are in operation, while numerals above the line show the appropriate run number.

CONSOLE OPERATION

Responsibility for console operation is assigned to a specific individual who is familiar with the machine and the operation of the various runs as described in the run and console manuals. He should also understand the utility programs used in the system.

In addition to the principal responsibility of operating the console, this individual may have other duties:

- Obtaining necessary materials for a run: input tapes and cards, programs, operating instructions, etc.
- Disposing of the output data.
- Checking tape labels and running program tests.
- Maintaining status records.

Program maintenance personnel are often assigned to assist the console operator when necessary.

Documentation

A daily log of continuous operations is maintained at the console: program number, start and finish time, tape drives and other peripheral equipment used, etc. Records of equipment usage provide a basis for calculating rental charges and assessing line departments for a share of the operating costs. Standard error forms are available to record the status of console indicators when an unanticipated stop occurs.

Manual entries and error detections are automatically printed on the console typewriter of some data processing systems. By printing indicative and control information, a complete log of the operation is available for audit purposes.

A time clock, preferably a consecutive spacing recorder or interval timer, should be placed in the console room to allow exact timing in and out of all production and test runs, and to measure actual against planned time.

MAGNETIC TAPE CONTROL

When a large number of magnetic tapes are used in a data processing system, special control techniques and procedures are followed.

Tape Library

A major control is exercised through the establishment of a tape library under the supervision of an individual responsible for:

- Documented control over issuance and return of magnetic tapes, program decks and tapes.
- Control and maintenance of program documentation master copies.
- Removal and replacement of file protect rings.

The number and size of reels to be stored determine the required size of the facility. Information is drawn from:

- System and program flowcharts.
- Volume of data in each tape input and output file.
- Number of history files.

As an example, three complete sets of reels are normally required for each master file: one for read-in, one for writing out the updated master file, and a third for backup. After an updating run, the input set is forwarded to the library as the new history or backup file; the output set becomes input for the next updating run; and the old backup set can be used as an output or work tape for the next master file updating.

The number of reels in a file is determined from:

- Number of characters in a record
- Number of records
- Number of records to a block
- Method of recording alphabetic and numerical data (in certain systems)

The number of records per reel can be obtained from charts reflecting the above factors as variables; this result is divided into the total number of records to obtain the number of reels.

Reels should be cycled efficiently in the library to minimize the dollar investment in tapes. This is accomplished by scheduling reels into system runs, and into and out of history files.

As tapes are ordered, adequate lead time is calculated and provision made for a few extra reels to use on additional application tests, special one-time reports or statistical runs, utility routines and programs, etc. Upon delivery, they are assigned a serial number and applied to different applications by this number. Reels are scheduled through the system and into and out of history files; control sheets for issuance, mounting, processing and return of tapes to the library are set up from these schedules.

Section	Area	Page
01	03	03

Magnetic tape must be stored under special temperature and humidity conditions; otherwise, the tape has to be conditioned prior to processing. Containers are stored on edge in cabinets or racks available from several manufacturers. Tape reels are always stored in a tightly closed container when not in use.

Whether the tape library contains a few or many reels, rigorous file and record control must be practiced to maintain vital company information for immediate access. The following points are considered in setting up a control plan:

- The location of any given file (one or more reels) has to be known at all times.
- Files are always signed for when released from the library.
- Reels are always stored in the same location in the library, by serial number, file or application.
- Information is recorded on tapes reaching scratch date to permit their subsequent release for other applications.
- Documentation should be concise, understandable, and require a minimum of entries.

In general, control documentation for a tape library is built around three types of records: file history, reel history, and tapes available. A file history card is prepared for each file, describing jobs on which it is used, required input tapes, retention cycle, date written, scratch date, etc. Tape issue and return signatures are noted on this form, as well as serial numbers for the purpose of tracing changes. Each reel of tape in the library has a reel history card identifying serial number, corresponding file (past and present), number of times written on, tape age and length, reported skips or errors, etc. As entries are made to the file cards, they are also made to each reel of file. A tape-available record is prepared for each working day in specified future periods. When scratch date entries and serial numbers are added to the file history, they are entered simultaneously into the tape-available record. At the time this date becomes current, the tapes listed here may be reissued as work tapes.

Some users maintain these records on unpunched IBM cards to facilitate handling and rearrangement; others punch the information into the cards to obtain periodic listings for audit.

Handling and Care of Magnetic Tapes

Several factors have to be considered in establishing a procedure for handling magnetic tape:

- Tapes should be in their containers when not mounted. They are not to be placed on tables

or handled without cover protection.

- In the event of a power failure, mounted tapes are removed from the read-write heads of each unit in ready status before restoring power. This prevents transmitting extraneous noise to the tapes as power is turned on.
- The top of a tape unit is not to be used as a working surface. Material placed on top of the units is exposed to heat and dust from the blowers and obstructs cooling.
- Tapes should be protected from dust and dirt, since foreign particles reduce the intensity of reading and recording pulses.
- Tape edges have to be kept free from nicks and kinks, because recorded information comes within .020" of the edge.
- During shipment, tape and reel are packed securely in a dustproof container, containers are hermetically sealed in a plastic bag (ordinary plastic bags can be sealed with a hot iron), and additional support is obtained by enclosure in separate reinforced cardboard shipping boxes.

When a reel is bent or broken as a result of dropping, the tape may be crimped at the edge, or soiled. It is inspected visually, or by mounting on the hub of a tape unit. If the tape is still serviceable, it is rewound on another reel, and the broken one is discarded. Crimped-edge tapes containing important information are reconstructed through tape-to-printer or other machine operations. Where reconstruction fails, partially or in total, tapes are recreated from the original input and control data, or from status at the last updating along with subsequent transactions. In any event, dropped tapes and reels are thoroughly cleaned, and if no damage is revealed, they are tested for fitness before reprocessing.

External and Internal Tape Labels

Tape reels are identified in two ways: external (visible) and internal (invisible) labels.

External labels are affixed to the reel and come in two types: (1) permanent reel labels, which contain information on serial number, date received from manufacturers, and current tape length, and (2) paper labels with adhesive backing, containing such information as file number or name, reel number, run date, date of work processed, output tape unit address, numbers and tape addresses of runs used on, and scratch date. Colored labels can be secured to designate different classes of application -- for example, blue for cost accounting, red for inventory, etc.

Application of the external label is illustrated in the following sequence of events:

- The librarian searches the tapes-available list to satisfy a request for a scratch reel.
- Before releasing the reel, the existing label is checked to see whether scratch date has been reached.
- If scratch date has been reached, the temporary label is removed from the side of the reel, and a file protect ring inserted.
- Entry is made to the tapes-available sheet.

External labels and rings should not be on a tape reel simultaneously. Upon reissuance from scratch status, the old label is removed, and a file protect ring is put in to permit writing on the tape. When a new file is created and turned over to the librarian, the label may be only partially complete. Entries such as scratch date and "used on" are transferred to the label from the file history.

A number of standard internal tape labeling routines* have been devised; most of them follow a consistent pattern for writing and checking header and trailer records on a tape reel.

A header, the first record on a magnetic tape, usually contains the following information:

- Header serial
- File number
- Tape serial number
- Reel sequence number
- Creation date
- Retention cycle
- File identification

The header may be checked visually by the console operator, or automatically by the program to verify that:

- File number is correct for the job in process.
- File is on proper tape unit.
- Sequence of subsequent reels is correct.
- Scratch tape has been reached on reissued reels before writing on them.

Several checking techniques can be employed:

- As a program writes a new header on an output reel, it is also printed to allow comparison with the external label (especially for scratch data).
- Label data can be punched for checking purposes.
- Label data can be printed out on an external label for attachment to the reel when it is returned to the library, or when the job is reprocessed.

A trailer, the last record on a magnetic tape, follows the tape mark, and usually contains:

- EOR or EOF
- Block count
- Record count
- Hash total
- Contents of control fields of first and last records on the reel (optional).

EOR is placed in the trailer of every reel except the last, where EOF is used instead. Thus, under input file conditions, the program will know when the last reel has been processed by testing for EOF. Record counts and hash totals facilitate systems checking; the latter should be compatible with utility sorts and merges.

When a master file is composed of many reels, knowledge of the contents of the first and last record control fields is useful. These reel boundaries may be written on the external file label. In a 100-reel master file with transactions affecting only a few reels, this practice allows updating by running only the reels with transactions against them.

With sorting and testing applications there may be no need for labels, and certain tapes are set aside for this purpose. Checkpoint memory dump tapes should either contain a header or be separated from other tapes to prevent usage as an output scratch reel. Under the latter condition, a program might attempt to read in the header from this tape, bring in a checkpoint record instead, and get itself wiped out in process. It is even conceivable that the foreign program might take over and process the input tapes if tight automatic controls are not present, or if the console operator fails to notice the condition.

PROGRAM CONTROL AND MAINTENANCE

A number of considerations enter into the documentation control and maintenance of programs for an operating data processing system.

Program Control

Just as with magnetic tape files, program decks, printer carriage control tapes and control panels are clearly marked for identification, stored in a specific location, and issued from and returned to a library.

Original programming cards and machine-language decks are frequently maintained through a card file in the tape library. The librarian is then responsible for issuing and filing program decks, and keeping track of the persons to whom they were released, and when.

*"IBM Standard Tape Label" (C28-8142)

Section	Area	Page
01	03	05

Decks should remain intact and are not revised without approval and documentation. A card count control procedure can be established to provide additional control of program decks.

Program Tape Master File

Aside from program deck control, there are advantages to be gained in the maintenance of all programs on a reel of magnetic tape:

- Program loading from tape is faster than reading cards.
- Risk of losing cards is eliminated.
- There is no need for an online card reader.

On the other hand, a plan of this kind requires:

- A utility routine to locate the desired run for loading (the operator keys in the run number, and a search program locates and loads the program into the machine).
- A maintenance or run-corrector routine for adding, deleting or changing programs on the tape.
- A tape drive during program loading (perhaps an alternate drive for one of the program output files).

Several factors are involved in the design of a program tape master file:*

- The tape can be made up of actual program cards, or it might consist of large blocks of machine instructions. The latter requires less tape space, and speeds program loading.
- The sequence of the programs on tape affects search time. Under ideal conditions, the programs are in proper sequence for a day's operation, but this may require separate tapes for individual applications or a particular processing cycle. Sometimes the program can be loaded at the beginning of the first master file reel.
- Sort and utility programs could be carried on a separate tape.
- Duplicate tapes should be retained in the library in the event of tape damage. Duplicate card decks should also be maintained for backup.

Sort and Utility Programs

Sort and utility programs are controlled in the same manner as production programs. Special factors to consider are:

- Control cards for a frequently used standard sort run must be clearly identified and filed in a specified location.

*"Program Testing" (F28-6038), "IBM 7070 Utility Programs" (J28-6058), "Programming Systems Concepts" (R25-1551-1), "IBSYS and IJOB Systems for the 7040/44 and 7090/94".

- Since some utility programs are used frequently and even unexpectedly, it may be advisable to maintain a set of these programs in a card file near the console.

Subroutine Library

Selected standard routines are incorporated in most programs. The IBM Autocoder system provides for maintenance of these subroutines on tape as macro statements in the Autocoder library.

Control responsibility is delegated to the programming staff, and in particular to those working on computer standards and methods. However, subroutines are used at the time of program assembly, and it may be desirable to keep the card decks or tapes in the machine room program library.

A complete and accurate subroutine manual is compiled by computer methods personnel to advise programmers and program maintenance specialists of the standard routines available to them and of their application.

Program Maintenance

After a program has been in satisfactory operation for a time, it may require modification for any number of reasons:

- The need for additional output information.
- The desire for a change in input or output format.
- Normal changes in the business: new products, revised discount structure or commission rate, etc.
- Introduction of improved programming techniques, or new data processing equipment.
- Increase in the scope of an application or activity.
- Poor or incomplete definition of requirements at the time the system was designed.

Routine and minor program maintenance may well be performed by operating personnel, but substantive redesigns should be referred back to the systems engineer and staff programmers.

Any changes brought about by program modification should receive the same thorough documentation as the original systems design and implementation. Maintenance programmers, librarians and console operators are responsible for seeing that these revisions are properly introduced into program manuals, console run books, and related records.

In establishing maintenance procedures and responsibility, two other factors have to be taken into account:

- The original programmers are to be relieved of any accountability for the program.
- There should be sound justification and need for the change.

PERFORMANCE MEASUREMENT

A general manager must keep his finger on the pulse of the business and must be constantly aware of how well each functional component is performing. Performance measurement is just as applicable to the data processing organization as it is to any other area of the business. In fact, it may be even more important here, especially after a large investment has been made to bring the system into operation from the design stage, with everyone looking for favorable results.

Development of a Plan

The data processing unit is perhaps in a better position to establish a comprehensive measurement plan than any other functional group because the documentation is so thorough and jobs have already been timed to permit laying out realistic operating schedules.

The steps in developing a measurement plan are quite similar to those outlined for the main business system during requirements specification:

- Select a series of time, cost, accuracy and volume factors that cover all phases of the operation.
- Establish time and cost standards historically or through special analyses.
- Measure actual to planned performance on a continuing and periodic basis.
- Take corrective action in areas that are found to be operating at less than satisfactory levels.

Measurement Factors

The size and scope of the measurement plan are determined by the data processing manager. Some of the measurement factors that may be included in the plan are:

- Ratio of total operating costs realized (salaries and indirect expense) to budget
- Overtime dollars
- Relationship of total data processing expense to some appropriate base such as total overhead expense of the business, net sales billed, etc.

- Number of errors generated per time period (also number of errors detected in input data)
- Number of complaints registered on service
- Amount of machine idle time by unit
- Amount of machine downtime, for reasons other than lack of load
- Cost of operation by application, activity or user department
- Length of time jobs are in queue beyond planned start time
- Length of time job processing exceeds planned times
- Appearance and quality of output reports
- Ratio of floor space utilized to total business floor space
- Ratio of promises kept to scheduled due dates
- Losses (time and cost) due to operator errors (for example, mistakes at console, misplaced or damaged reels, etc.)
- Personnel turnover
- Quality of housekeeping
- Currency of documentation updating (do principal instructions, programs and other documentation reflect all required modifications to date?)
- Estimate of cost and time savings realized by activity and application, as compared with the estimated cost of operation under the previous system

The list is representative rather than all-inclusive, and suggests the type of factors to be incorporated in the plan. Some will be measured periodically, others on a weekly or monthly cycle.

Cost information is built up over a period of time before it becomes useful as a recognized standard. Allowed time cycles are drawn from run timing data and planned work schedules, as well as special studies on job steps, operators and equipment.

Reporting Performance

Operating performance is reported periodically (monthly, as requested, etc.) to management in a formal document. Comparison of actual to planned performance of selected factors can be plotted on charts and graphs to serve as a constant reminder of work status and progress for data processing personnel. Where practical, performance reports are mechanized.

PERSONNEL DEVELOPMENT

The overall quality of a data processing organization is determined largely by the competence and motivation of its personnel. Again, this is mainly

a responsibility of the data processing manager, but personnel considerations are described briefly here for those situations where a systems engineer may be called upon for advice and counsel in a new installation.

In general, a manager wants productive people who are interested in their work and in improving their technical competence. To cultivate a good working climate, group cohesiveness, and a high level of productivity, a manager devises and follows a broad plan of personnel development.

Elements of such a plan include:

- Preparation of selection criteria for recruiting and hiring
- Clear delegation of responsibility with comparable authority
- Provision for periodic retraining and job rotation of personnel (job enlargement is encouraged over intensive specialization)
- Annual performance rating, appraisal and review
- Good working conditions and environment
- Recognition and reward for superior performance
- Development of self-improvement plans for promotable personnel
- Development of effective communications to and from other functional departments, management, and within the data processing unit
- Preparation of position descriptions and job instructions by personnel performing the work
- Providing opportunities for promotion in the business
- Encouraging participation in the decision-making process

Technical competence of the personnel can be further developed in several ways:

- Attending selected data processing symposiums, seminars and workshops presented by professional groups
- Visiting other well designed data processing installations to review applications, techniques and practices
- Planned reading of selected texts and periodicals in the computer field to secure new ideas
- Enrolling in special courses at nearby universities and training centers

SYSTEMS DEVELOPMENT AND MATURATION

In the early days of operation, many problems must be faced and resolved to smooth out the working of the system and improve its efficiency. Some of these refinements will be small:

- Work flow is rescheduled in the light of actual practice and experience.
- Preventive maintenance schedules are established.
- New operating instructions are prepared and issued (for example, documentation control through use of aprons and batch numbers).

Others will be more significant in their impact:

- Where a policy of progressive implementation is followed, new activities and applications are added in accordance with a planned installation schedule.
- New techniques and practices are evaluated for potential in the system.
- New and improved services are rendered to user departments.
- Changing requirements will necessitate substantial modifications to certain parts of the system.

Then there are more fundamental forces at work which influence every data processing organization, large and small:

- The business grows and expands as broader outlets are found for current and new products and services.
- Computer equipment and technology are developing so rapidly that no one in the field can afford to be complacent.

Data processing managers, systems engineers and programmers are directly involved in this process of continuous revision, modification and improvement to upgrade the quality of the system. Beyond routine system maintenance spot audits, for example, should be conducted periodically in various areas of the system, and a complete system review and appraisal performed once a year, or every 18 months. Finally, when the useful life of the system is over, some five, eight or ten years in the future, the complete cycle of study and design, implementation and operation has to be initiated once again.



DISPLAY TECHNIQUES FOR REPORTS

INTRODUCTION

Effective visual communication is an essential aspect of the systems engineer's job. The ability to summarize, interpret and emphasize information in a visual display not only gives him a means for effectively conveying his work to others, but also serves to assist him in the analysis and solution of his engineering problems.

Graphic presentations make use of a variety of display techniques. However, all of these techniques have the common purpose of communicating quickly, clearly and memorably. In order to do this, the graphic display must have unity, simplicity, organization and visibility. That is, it must be unified in the sense that it delineates one central theme, simple enough so that it eliminates the trivial and unnecessary, organized toward the goal of effective communication, and visible in that it can be easily seen and understood.

Although the fundamentals of graphic displays are understood by most systems engineers, there are many ways in which such displays can be decidedly improved through a better understanding of display techniques and the proper use of such techniques.

The purpose of this section is to present information that will allow the systems engineer to make more effective use of two basic display techniques — graphs and tables — in the preparation of visual communication material.

GRAPHS

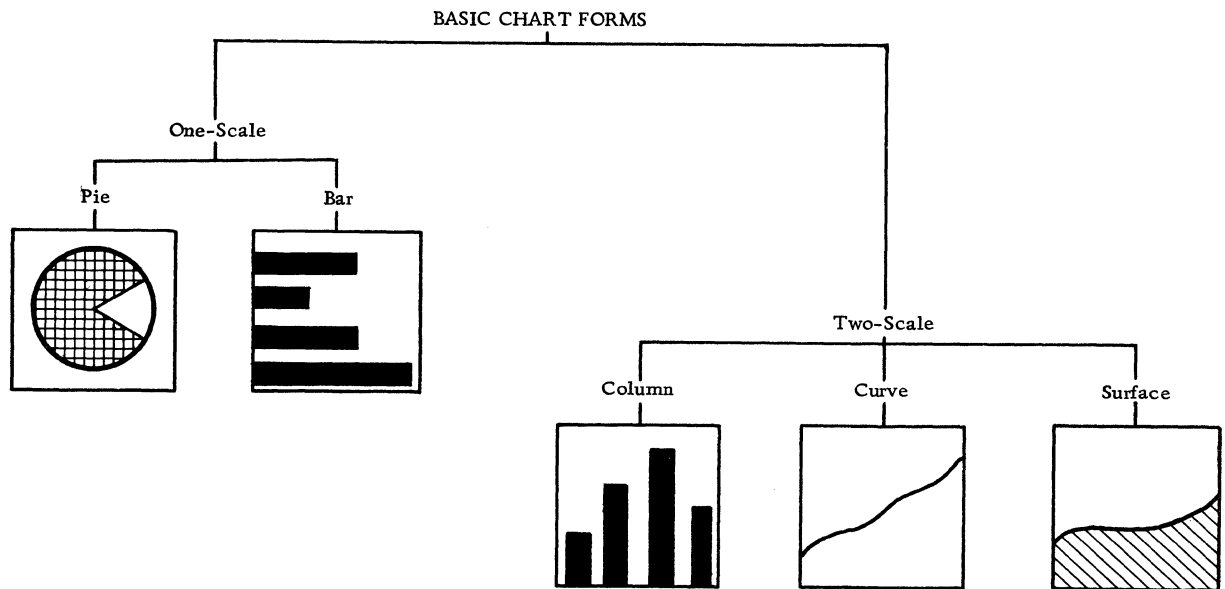
Graphs depict numerical or quantitative relationships. Functionally, there are two basic types of graphs: statistical and mathematical.

Statistical graphs portray data by lengths of bars, lengths and slopes of lines, by areas, volumes and countable units. For completeness, they require some method of estimating values, a title, and a source reference. Statistical graphs are effective tools for condensing, relating and communicating data, and are also designed for efficient and realistic recording and analysis.

A mathematical graph pictures a function in order to reveal the relationship between variables, to assist in the solution of equations, or to speed a computation.

Basic Types of Graphs

Physically, there are two main types of graphs: those which have one scale and those which have two scales. The following diagram shows the major varieties of one-scale and two-scale graphs. These are the basic few from which most graphs stem. One-scale charts are generally simpler than two-scale and are therefore better for an audience which has little experience with graphs. However, for most engineering applications, a two-scale graph is necessary because it more fully explains complicated relationships.



Reproduced, with permission, from "Making the Most of Charts", by K. W. Haemer, published by American Telephone and Telegraph Company, 1960.

One-Scale Graphs

The pie graph is simply a circle divided into sectors. The scale is the circumference divided into suitable scale units such as percentages. The primary use for such graphs is to show the component parts of a whole. They are useful in only a small number of applications.

In the bar graph, the scale extends horizontally along the length of the bars. The vertical dimension does not have a scale, but merely provides space for a series of items and a bar to measure each item.

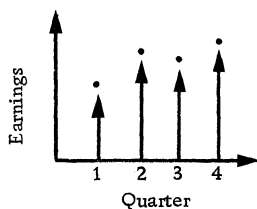
Bar graphs are useful for the comparison of distinct items. There is no graduation of items such that a range or point-to-point scale is required. Each item is measured on one scale only.

Two-Scale Graphs

The two-scale graph allows data to be recorded in such a way that two scales are conveniently cross-referenced to each other. The vertical scale (or ordinate) usually measures a quantity, while the horizontal scale (or abscissa) usually measures a sequence or time factor. Values are derived by measuring data on both scales.

These values form either discrete or continuous sets of points on the graph. Discrete sets imply the existence of gaps between values — for example, an event occurs once or twice, but not one and one-half times. Continuous, as used here, denotes an absence of segregated values — for example, measurement of data where any value can occur.

In the column graph, the plotted points are connected to the horizontal axis. This type of graph is useful in simple applications where the variation among items on the vertical scale is easily recognized by the height of the vertical columns. For example, a column graph could represent the quarterly earnings of a company; the variation among earnings for each quarter is clearly evident from the height of the bars.



The curve graph, or line graph, is the most common type used in engineering applications. It is advantageous where there are a large number of closely spaced points to be plotted, where there are several series of points, or when the level changes only slightly from one point to the next. In this form,

the plotted points are connected to one another instead of to the base, thus forming a curve which shows a point-to-point relationship between the values.

The surface graph is a combination of the column and curve graphs. Like curves, surface graphs connect each plotted point to the next; like columns, they join each point to the base. Thus, an area, or surface, is formed under the curve. Graphs of this type can be used to calculate proportional parts of two-scale functions by comparing individual area segments to the entire area under the curve.

Subtypes: The Histogram

Within each basic graph type there are many subtypes. Detailed discussions of them may be found in "Graphs — How to Make and Use Them", by Herbert Arkin and Raymond R. Colton, published 1936 by Harper Brothers, and in "Graphic Presentation", by Willard C. Brinton, published 1939 by Brinton Associates.

One subtype that is particularly useful to the systems engineer is the histogram. This graphical representation shows a frequency distribution by a series of rectangles which have for one scale (the abscissa) a distance proportional to a definite range of frequencies, and for the other scale (the ordinate) a distance proportional to the number of occurrences within the range. This subtype combines advantages of both the column and curve graphs. The histogram is thus useful where the main interest is centered around the number of times that an event occurs over a range, rather than at discrete points or on a point-to-point basis. It could be used, for example, to show the percentage of input messages that enter a system during specified time intervals of a day.

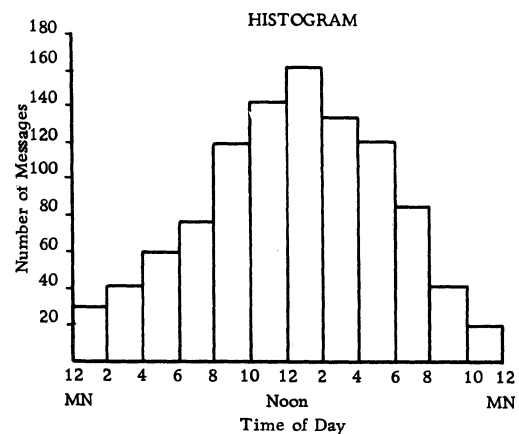
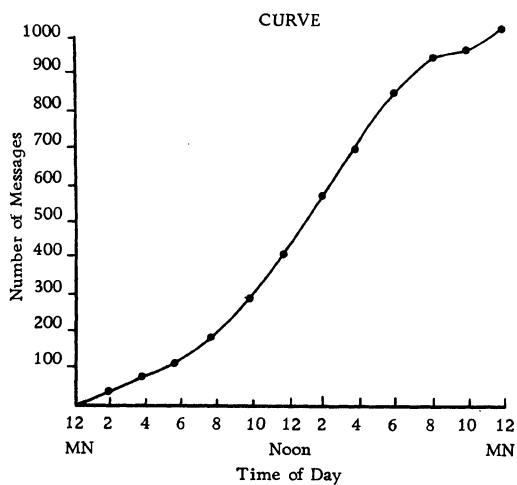
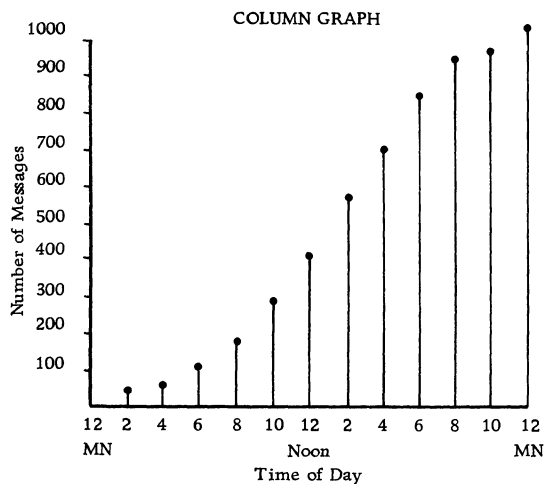
Selecting the Best Graph Form

The graph form selected for use in a given application should be one that precisely fits the information and the purpose for which it is being presented. Often the same information may be correctly displayed in several forms. In such a case, the form selected should be the one which places the best emphasis on the relationship to be stressed.

For example, the systems engineer may have data which shows a cumulative count of input messages into a system during segmented time periods of an entire day. This data can be displayed graphically in the form of the following column graph, curve or histogram.

The column graph would be useful to show the total number of messages that have been in the system through any hour of the day. The curve graph could be used to calculate the rate of increase of

input messages during any time period of the day. The histogram shows the total number of messages that entered the system in each two-hour time period. It clearly shows the peak input periods of the day. Thus, the same information has been displayed in three distinct graph forms which place emphasis on total cumulative count, rate of increase, and individual period count, respectively.



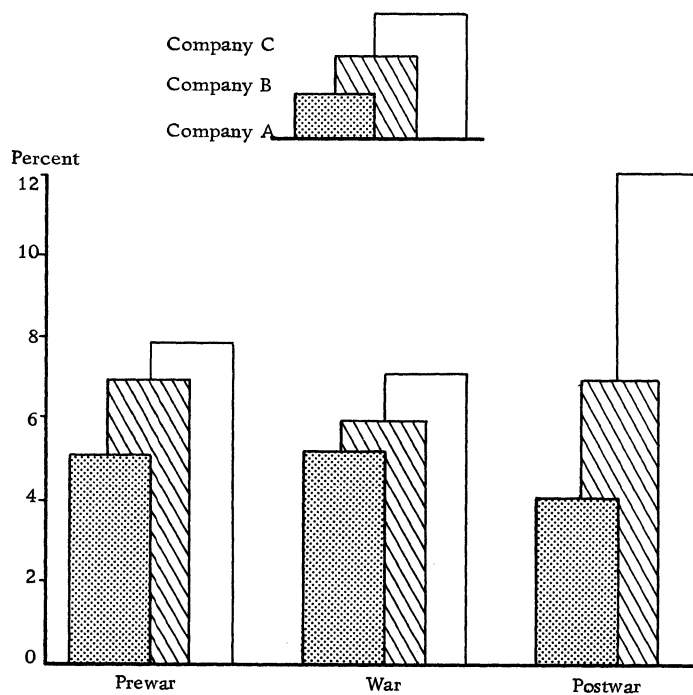
Graph Construction Techniques

An effective graphic display contains the attributes of directness, clarity, simplicity and accuracy. Clarity is achieved by presenting the display in terms and concepts that the audience will readily understand. Simplicity is achieved by functional organization and the avoidance of unnecessary detail.

Directness

The attribute of directness is not always easily attained in the graphic display. Although this simply involves sharp focusing on the main idea of the graph, many graphs that are technically complete and correct are not direct. The following example shows the difference between being direct and being indirect. If three separate column graphs were used to compare the rate of return on capital for three companies, A, B and C, during the prewar, war and postwar periods, they would be simple, clear and accurate, but not direct. The form shown below not only focuses much more sharply on the desired comparison, but also has the advantage of using less space.

RATE OF RETURN ON CAPITAL



Reproduced, with permission, from "Making the Most of Charts", by K. W. Haemer, published by American Telephone and Telegraph Company, 1960.

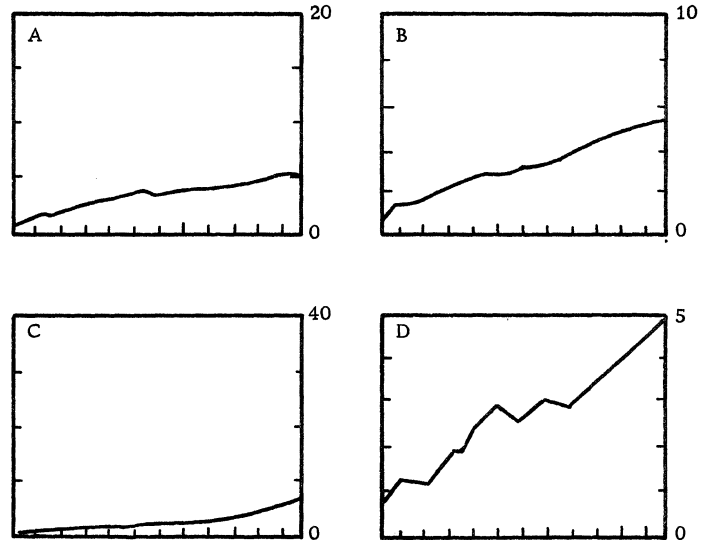
Accuracy — Scale Selection

There are many types of scales used in the construction of graphs. The two most common types used in business applications are the arithmetic scale and the logarithmic, or ratio, scale. In engineering applications, many combinations of these are used (for example, the semilog scale). The arithmetic scale is the one most commonly used; it represents equal quantities by equal differences. In the logarithmic scale, equal distances represent equal ratios. This scale can be used to show the relative importance of changes at different levels or to show the general pattern of relative rate of change. The choice of scale for a given application depends on what emphasis is to be placed on the data.

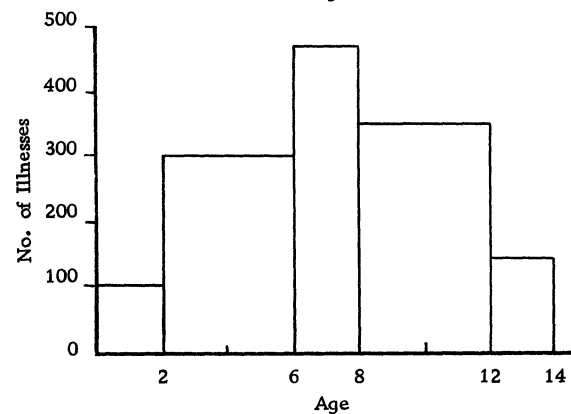
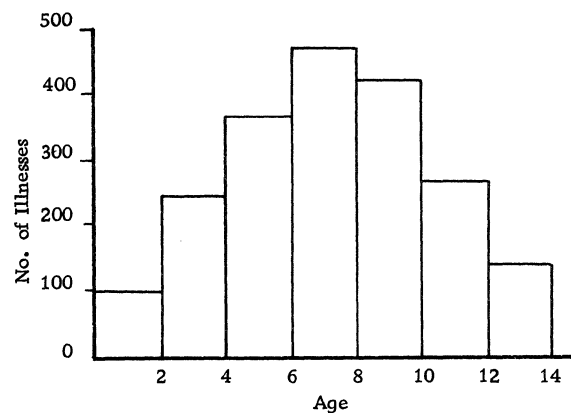
The accuracy of a graphic display goes much further than merely using accurate data and plotting it correctly. The graph must also provide an accurate impression of the situation it pictures. The one feature of graph construction that has the most direct effect on the impression produced is the scale selection. The following paragraphs will reference the techniques of selecting the best scale range and scale unit, and will discuss the effects of removing part of a scale and changing the scale unit within the graph.

The scale range should be chosen so as to give an accurate impression of the data. The discriminate choice of a scale range is especially important when plotting growth or rate functions. In general, an extremely fast growth or rate should be plotted on a small-scale range, and an extremely slow growth on a large-scale range. Relatively fast or relatively slow growths should be plotted on an intermediate-scale range. For example, scales A and B above right indicate relatively slow and relatively rapid growths, respectively, for the same data. Scales C and D show extremely slow and extremely rapid growths, respectively.

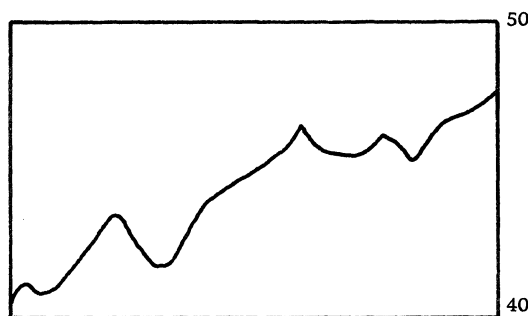
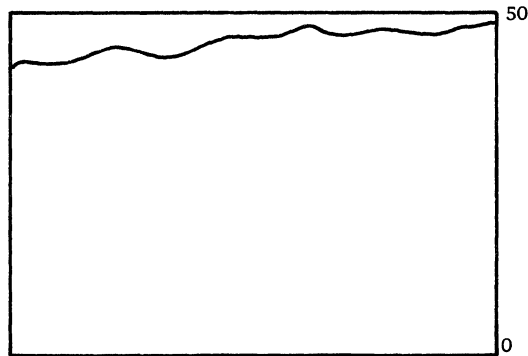
Scale units or intervals do not have to be equally spaced along an axis of the graph. In other words, the master scale can be altered to show smaller or larger intervals. However, when this technique is used, the height of the plotted points must be proportioned exactly to the width employed. For example, the two histograms at right are equivalent. Care must be exercised when this technique is used. If the vertical scale is not proportioned to the horizontal scale, an outright distortion will occur.



Reproduced, with permission, from "Making the Most of Charts", by K. W. Haemer, published by American Telephone and Telegraph Company, 1960.



Removal of part of the scale of a graph is sometimes advantageous. This might be done to emphasize certain parts of the graph or to eliminate an insignificant portion of the graph. This is a valid technique but care must be taken to show the break clearly so that it cannot be missed by the reader. The following two graphs depict the same data:



Reproduced, with permission, from "Making the Most of Charts", by K. W. Haemer, published by American Telephone and Telegraph Company, 1960.

The one on the top presents a picture of total amounts while the other presents a picture of only differences in total amounts; visually, only differences can be compared, not totals. This difference must be clear to the reader or a distorted interpretation of the graph will result.

References

Mode, Elmer B., Elements of Statistics, 3rd edition, Prentice-Hall, Inc., 1961.

Haemer, K. W., Making the Most of Charts, American Telephone and Telegraph Company, 1960.

TABLES AND CHARTS

Structure of the Formal Table

The table is one of the chief ingredients in almost any statistical study. By means of this display technique, a vast body of data is recorded in such a fashion that, while each recorded event retains its individuality, coherent summations can be made. Depending on the way the cells of information are placed in relation to the overall theme of the table, the accompanying totals may lend themselves conveniently to formulae structures; and it is from the formulae used that the systems engineer is able to plan his approach toward a problem-solving situation.

The structural parts of the statistical table are relatively consistent, regardless of the particular table variation involved; however, changes in subject matter or meaning will affect the design of the various display parts to a modest degree.

Above the top line of the representation appears the heading, which consists of the table number, title and headnote.

The table number appears when a series of displays is used so that the relative position of a given piece may be noted. Should a table appear alone, no number is employed.

The title includes information necessary to impart the nature of the table: data description, classification, time reference, etc.

The headnote, which appears in brackets below the title, explains the data contents more fully, perhaps to limit the data or to refer to some dominant theme within the body in greater detail.

The stub, usually found to the left of the page just below the headnote, contains the listed row captions. Centered heads and subheads, which appear as part of the stub, further classify the table data. The stubhead (box) defines the entire listing below it, whereas the center head (subhead) describes only associated segments of the stub. The line caption is the foundation or smallest unit of the stub.

The block is a section of the stub whose integral parts are closely interrelated. A series of line captions together with pertinent heads and subheads — usually an entry within itself — may form the block. However, a stub may consist of a single block.

The boxhead contains the captions of all the individual vertical columns. Spanner heads within

the boxhead may segregate the columns into sub-classifications. The panel is to the boxhead what the stub block is to the stub: it forms related columns into a self-contained unit.

The field falls below the line beneath the boxhead and extends to the bottom rule of the table. The actual data is found in the field.

A cell is the basic table data unit found at the intersection of any line caption with a column head.

The footnote explains some fact in regard to the table above — omitted or added information, even a general statement which may refer to the table as a whole.

TABLE 6.—AGE OF ALL PERSONS AND OF CITIZENS BY SEX, FOR THE UNITED STATES, URBAN AND RURAL: 1940
 [Age classification based on completed years]

Area and age	All persons			Citizens ¹		
	Total	Male	Female	Total	Male	Female
UNITED STATES						
All ages.....				769		
Under 5 years.....				26		
5 to 14 years.....				115		
15 to 24 years.....				139		
25 to 34 years.....				178		
35 to 44 years.....				205		
45 and over.....				106		
21 and over.....	986	475	513	567	302	265
URBAN						
All ages.....				453		
Under 5 years.....				15		
5 to 14 years.....				73		
15 to 24 years.....				86		
25 to 34 years.....				104		
35 to 44 years.....				116		
45 and over.....				59		
21 and over.....				328		
RURAL						
All ages.....				316		
Under 5 years.....				11		
5 to 14 years.....				42		
15 to 24 years.....				53		
25 to 34 years.....				74		
35 to 44 years.....				89		
45 and over.....				47		
21 and over.....				239		

¹Includes both native and naturalized.

The formal table and its major parts. Reproduced from "Bureau of the Census Manual of Tabular Presentation", 1949, a special report of the Bureau of the Census, prepared by Bruce L. Jenkinson, with permission of U. S. Government Printing Office.

TABLE No.—TITLE OF TABLE

(Headnote)

PANEL	Stubhead	Spanner head			Spanner head †			The column Total	
		Column head	Column head	Column head	Column head	Column head	Column head		
BLOCK →	CENTER HEAD								
	Total line caption.....				Cell			769	
	Line caption.....				Cell			26	
	Line caption.....				Cell			115	
	Line caption.....				Cell			139	
	Line caption.....				Cell			178	
LINE →	Line caption.....	Cell	Cell	Cell	Cell	Cell	Cell	205	
	Line caption.....							106	
BLOCK →	CENTER HEAD								
	Total line caption.....				Cell			567	
	Line caption.....				Cell			453	
	Line caption.....				Cell			15	
	Line caption.....				Cell			73	
	Line caption.....				Cell			96	
	Line caption.....				Cell			104	
	Line caption.....				Cell			116	
	Line caption.....				Cell			59	
	Line caption.....				Cell			328	
	BLOCK →	CENTER HEAD							
		Total line caption.....				Cell			316
Line caption.....					Cell			11	
Line caption.....					Cell			42	
Line caption.....					Cell			53	
Line caption.....					Cell			74	
Line caption.....					Cell			89	
Line caption.....					Cell			47	
Line caption.....					Cell			239	
Line caption.....					Cell				
† Footnote.									
→21 and over.....		968	475	513	567	302	265		

The formal table and its major parts. Reproduced from "Bureau of the Census Manual of Tabular Presentation", 1949, a special report of the Bureau of the Census, prepared by Bruce L. Jenkinson, with permission of U. S. Government Printing Office.

The Resource Time Usage Chart

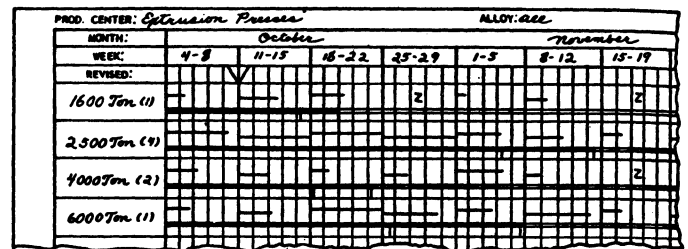
The essential feature of the resource time usage chart, often called the Gantt chart, is that it provides a view of plans and/or progress in relation to time. Charts with this time-scale feature have been devised for a wide variety of applications, including load planning, scheduling and progress reporting. The value of the Gantt chart stems from its ability to show quickly and clearly the relationships among several variables. One such chart can show the relationships among work planned, elapsed time and achievement quantities in one convenient display. An entire situation — performance versus prediction — can be perceived from one chart.

The Gantt chart is thus a valuable tool for the systems engineer in the planning and control of jobs which involve machine or personnel functions in the data processing operations procedures.

The Gantt chart may be prepared in a variety of forms. Basically, the design involves placing the items to be charted in a left-hand column and using the remainder of the chart for time ruling. Any amount of data or notation may be placed on the chart to show relationships among variables, but care must be exercised to make sure that primary data is not obscured by excessive detail.

As with all visual displays, the Gantt chart should stress the important relationships in a given operation. The following are some of the many techniques which can be used effectively to place emphasis on selected relationships:

1. The load planning chart shows, for each machine, the work available each week (light lines) and the total backlog (heavy line). The use of the different line weights shows the comparison vividly.



Load planning chart. Reproduced, with permission, from "Handbook of Industrial Engineering and Management", edited by W. G. Ireson and E. L. Grant, 1957, Prentice-Hall, Inc., pages 478 and 479.

2. The progress control chart for a continuing activity shows several techniques for making displays effective. The use of weekly numbers clearly indicates work scheduled and work actually completed in each week, along with the cumulative work scheduled. The bar for each item gives a dynamic interpretation of progress against schedule-to-date. The combined use of numbers and progress bars provides both a convenient source of data and an effective progress display.

DEPT: Assembly		PRODUCT GROUP: Types Totals						
MONTH	November			December				
WEEK	1-3	8-12	15-19	22-26	29-3	6-10	13-17	
DAYS ELAPSED								
Type R (add. hours)	715	700	675	550	710	710		
	604	553	577					
	715	1415	2070	2640	3350	4060		
Type Z (add. hours)	420	400	400	320	380	360	360	
	416	539	565					
	420	820	1220	1540	1920	2280	2640	
Type P (add. hours)	1800	1760	1760	1310	1640	1600	1620	
	1622	1685	1520					
	1800	3560	5320	6630	8270	9870	11490	

Progress control chart. Reproduced, with permission, from "Handbook of Industrial Engineering and Management", edited by W. G. Ireson and E. L. Grant, 1957, Prentice-Hall, pages 478 and 479.

3. Machine-use schedules show how, for a given job, the planned starting and completion time on each machine can be represented by segmented bars. In these schedules, abbreviations and codes are also used. Machine operators' initials are entered, so that the schedule also serves as a work assignment chart. Codes may represent any factor or condition which should be shown. For example, manpower requirements for given operations can be conveniently represented in the form of numbers placed over the bars.

	MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY
624 KEY PUNCH - 1	AP 1 1	CL 2 1	AR 3 1	CL 4 1	AR 5 1
628 KEY PUNCH - 2	AR 2 2	AR 3 2	AR 4 2	AR 5 2	AR 6 2
636 VERIFIER		AR 1 1	AR 2 1	AR 3 1	AR 4 1
677 COLLATOR	AR 1 1	AR 2 1	AR 3 1	AR 4 1	AR 5 1
680 SORTER	AR 1 1	AR 2 1	AR 3 1	AR 4 1	AR 5 1
519 REPRODUCER	AR 1 1	AR 2 1	AR 3 1	AR 4 1	AR 5 1
62A CALCULATOR			AR 1 1		AR 2 1
62 TAB	AR 1 1	AR 2 1	AR 3 1	AR 4 1	AR 5 1
62 INTERPRETER	AR 1 1	AR 2 1	AR 3 1	AR 4 1	AR 5 1

Machine-use schedule. Reproduced, with permission, from "Handbook of Industrial Engineering and Management", edited by W. G. Ireson and E. L. Grant, 1957, Prentice-Hall, Inc., pages 478 and 479.

4. Target-date schedules, based on existing commitment schedules, can be used in a development operation.

Targets for preceding stages can be set back from the test target date by the time interval required in each phase. Not all targets need be set back. Decisions can be made to determine likely areas of delay and thus more time can be allotted in these areas.

References

Carson, Gordon B., ed., Production Handbook, 2nd edition, Ronald Press Co., 1958.

Ireson, W. G., and E. L. Grant, eds., Handbook of Industrial Engineering and Management, Prentice-Hall, Inc., 1957.

STUDY DOCUMENTATION FORMS

In studying an existing system or designing a new system, it is usually desirable to document the key records and files, operations, applications and activities. It is also useful to associate time, cost and volume data with the various forms and functions, and even to relate the tasks to the organization. The forms used are intended to assist in providing a framework for recording such information when it is necessary to do so.

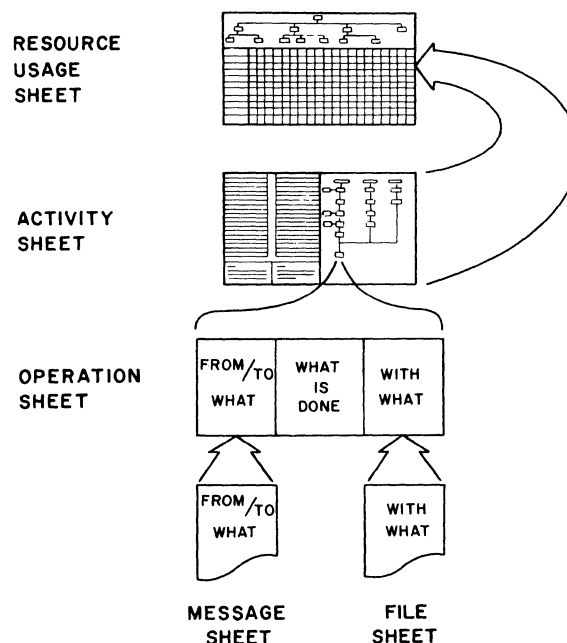
The following five forms, all available from stationery stores, work together to provide a coherent description of a business system with variable magnifying power, since they can be used at several levels of detail:

- Resource Usage Sheet (MO8-4607)
- Activity Sheet (MO8-4608)
- Operation Sheet (MO8-4605)
- Message Sheet (MO8-4604)
- File Sheet (MO8-4603)

The Resource Usage and Activity Sheets together display the dynamic mechanism of an activity; together with the Operation Sheet, they provide a fairly detailed operational view. The Operation Sheet by itself permits a look at critical operations and, when supported by the Message and File Sheets, provides a closeup of information used in or produced by such operations. Message and File Sheets by themselves permit observation and analysis of information inputs, outputs and resources, with again a variable level of detail as desired by the observer.

The amount of detail to be entered on these forms is completely discretionary, and depends on what level of understanding is needed as well as on the time allotted for the survey. Information can be omitted; the observer can single out critical operations from the Activity Sheet and avoid making up Operation Sheets for every operation. Short-cut methods can be used for entering information. Many of the fields on the Message Sheet can be left blank, since a sample of the completed message form is customarily attached. Qualification of content may not be needed on the File Sheet, nor even description of content.

The Activity Sheet traces the flow of a single activity, breaking it down into its major operations. Each Activity Sheet presents as large a group of related operations as can be handled conveniently. It includes a flow diagram of the activity, with individual blocks representing various operations (each of which can be more



Documentation Structure

minutely described on Operation Sheets). Key characteristics such as volumes and times are recorded in tabular form.

The Resource Usage Sheet and Activity Sheet work together to provide a quick look at a business system. In order to dig down for a closer look at the operation of a system, the observer uses forms that permit a more detailed documentation. These forms are the Operation Sheet and the Message and File Sheets.

There is usually an Operation Sheet for each block on the Activity Sheet. It is used for recording the related processing steps that form a logical operation. It describes what is done, with what resources, under what conditions, how often, to produce what specific results. Its primary purpose is to show the relationships between inputs, processes, resources, and outputs. Many people prefer to use annotated flowcharts or decision tables for operation description.

The Message Sheet is one of two forms that support the Operation Sheet. It describes the inputs and outputs — in effect, it displays, to the level of detail desired, from what source or toward what result an operation is working.

The File Sheet describes a collection of messages, an information file; it is the second of the two forms supporting the Operation Sheet and

shows with what stored information the operation works. Preparation of the forms is described in some detail by discussing the specific entries in the various fields. For a more detailed discussion, see "Study Organization Plan -- Documentation Techniques" (C20-8075).

RESOURCE USAGE SHEET

The purpose of the Resource Usage Sheet is to present on a single page the organizational framework and cost analysis of that segment of the business into which the activity under study fits. It is filled in at two points in a phase one survey: the organization chart and overall costs are prepared at the beginning of the study, and the activity cost figures, which emerge during the study, are entered at the last. In the following discussion, key numbers refer to the accompanying exhibit.

1. The structural organization is shown in an organization chart. This chart can be constructed from a company organization table or chart, or can be deduced from interviews. The area allotted dictates a 3- to 5-level chart. Organizational components of the bottom tier are lined up over the vertical cost columns. Hierarchic equivalence of the bottom-tier boxes is valuable but not necessary; if a group of units have little effect on the activity, they can be lumped together on the next-higher management level shown. Where it is necessary but topologically difficult to associate higher-tier units with a cost column, draw two dotted lines from the box to the sidelines of the column. Where possible, the command lines of the organization structure should be traced back until they join in a single management box.

2. The first horizontal band under the organization chart is used to enter inclusive resource costs, summing the costs for each department which affects the activity or activities being studied. There should be a cost tabulation for every box in the bottom tier of the chart. The figures can be historical costs, budgeted costs for the current year, or management estimates. Personnel costs should represent all salaries (including the department head's) and direct fringe benefits. Machine and equipment costs include total annual machine rentals except where machines or equipment are purchased; in this case an annual cost is computed by amortizing or using approved writeoff procedures. If machines are both purchased and rented, the sum of both costs must be developed. Material

costs include annual expenditures for office supplies such as blank forms, paper, and punched cards.

3. The miscellaneous classification allows for unusual and significant costs not otherwise classified, such as personnel training, money costs in a financial enterprise, warehousing costs in a wholesaling organization, and overhead costs if significant and available. Totals in each box are the sum of these inclusive resource costs.

4. In the other horizontal bands, underneath the inclusive summaries, are entered activity resource costs for each activity under scrutiny, which are developed during the survey. The definitions of the entries are the same. The values reflect only that portion which applies to the activity named in the identifying box at the left. For example, all the equipment and material costs of the billing and prebilling departments result from the charge account activity, and are so entered; 38 percent of the total costs of the new accounts department are ascribable to charge accounts, resulting in an entry of \$10,600 in that column.

5. Costs are not entered for organizational components that have nothing to do with the activity, as is the case for the advertising manager. Activity cost figures can be developed in part from unit time and volume figures on the Operation Sheet.

6. Two unlabeled columns at the far right are used for miscellaneous items, and for costs which are difficult to classify or not assignable to organizational units. The first of these columns identifies the item and the second displays the best cost figure available. These entries may fall into the top cost band (for inclusive or departmental summary figures) or into activity bands. Costs of inventories, including raw materials, in-process goods or finished goods; costs of money, including collection costs and bad debts; costs of accounts receivable — these are among items to be listed in these columns.

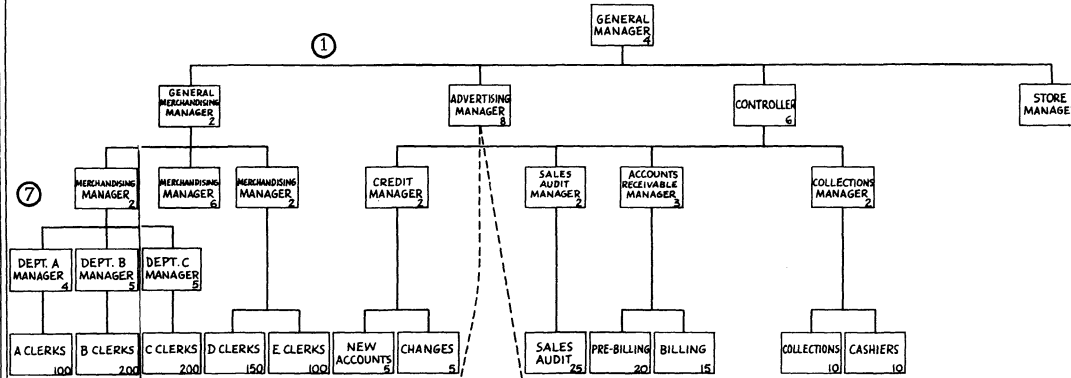
Totals are developed horizontally for the inclusive departmental costs and the various activity costs. Miscellaneous items in the special columns at right are summed into the grand totals. If all activities involving the departments in the bottom tier of the organization chart are presented on the activity cost breakdowns, the sum of the activity totals should be the inclusive departmental total.

7. Notes may be written in either the upper left or upper right corners of the Resource Usage Sheet.

NOTES

1. THE PERSONNEL COST FOR THE SALES CLERKS IS THE AVERAGE NUMBER OF CLERKS X \$ 4500.
2. NEW HIRES ARE SENT TO TRAINING CLASSES. THIS COST IS THE AMORTISED COST PER ATTENDEE.

ORGANIZATION CHART



RESOURCE USAGE SHEET

Study ASSOCIATED RETAILERS, INC.
 Analyst L.H. BAKER, JR.
 Date 29 FEB 61

- NOTES:
3. THESE FIGURES SHOW ONLY A FEW OF THE ORGANIZATIONAL COMPONENTS. THE TOTALS WOULD BE LARGER IF ALL COMPONENTS WERE INCLUDED.
 4. THE VALUES FOR THIS ACTIVITY WOULD BE ENTERED IN THE SAME MANNER AS THOSE FOR CHARGE ACCOUNTS - EXCEPT THAT NOT ALL OF THE SAME ORGANIZATIONAL COMPONENTS WOULD PARTICIPATE.

		TOTALS																			
PERSONNEL	②	450,000	900,000	900,000	675,000	450,000	21,000	21,000	80,000	105,000	84,000	63,000							3,833,000		
MACHINE & EQUIPMENT		15,000	25,000	25,000	20,000	15,000	1,000	1,000	3,000	4,000	40,000	24,000							184,000		
MATERIAL		14,000	23,000	28,000	15,000	14,000	1,500	500	10,000	3,000	10,000	18,000							142,000		
MISCELLANEOUS	③	10,000	18,000	17,000	10,000	5,000	4,000	3,000	4,000	6,000	8,000	10,000							100,800		
FORMAL TRAINING	2	4,500	9,000	9,000	6,750	4,500					3,000	3,000							45,750		
TOTAL		493,500	975,000	979,000	726,750	488,500	27,500	25,500	97,000	118,000	145,000	118,000							4,306,050	⑥	
ACTIVITY NAME	④	150,000	250,000	200,000	175,000	75,000	8,000	15,000		52,500	84,000	63,000							1,124,850	AVERAGE AMOUNT OF ACCOUNTS RECEIVABLE	5,000,000
CHARGE ACCOUNTS		750	1,000	1,000	800	400	300	300	⑤	2,000	40,000	24,000							82,260		
		15,000	20,000	24,000	18,000	14,000	1,200	400		1,500	10,000	18,000							123,850		
		1,000	2,000	2,500	2,100	1,250	1,100	900		3,000	8,000	10,000							34,850		
TOTAL	④	166,750	273,000	227,500	195,900	90,650	10,600	16,600		59,000	142,000	115,000							1,365,810	COST OF FINANCING @ 8%	400,000
ACTIVITY NAME	④																				
MERCHANDISE CONTROL																					
TOTAL	④																				
ACTIVITY NAME																					
TOTAL																					
ACTIVITY NAME																					
TOTAL																					
ACTIVITY NAME																					
TOTAL																					

ACTIVITY SHEET

The Activity Sheet provides both an insight into the dynamic mechanism of an activity and a measure of the magnitude of the processing problems in the activity. It covers five major areas: the operational flow of the activity, the characteristics of inputs, the characteristics of outputs, the characteristics of file usage, and the elapsed times involved in the activity.

1. The flow diagram charts the mechanism of the activity in action. Operations boxes identify each related set of processing steps which can be defined as an operation. Local jargon or more formal operative verb phrase may be used to describe the operations. Space at the bottom of the operation box is used for indicating, in abbreviated form, the organizational component which performs the operation, and for footnote reference numbers. Inputs are shown in an input/output symbol in which are entered the key number and name of the input; outputs are similarly treated. All files referenced during, or used in, the various operations are also shown as input/output symbols. Key numbers are assigned to inputs, outputs and files for cross reference to the tabular grid that takes up the left half of the Activity Sheet. Between-operations key numbers are also assigned for cross reference. The number series used for cross referencing is printed in the LEGEND box (6) on the Activity Sheet.

The four other major areas covered by the Activity Sheet — characteristics of inputs, outputs and file usage, and definitions of elapsed time — all are entered on the tabular grid. Since the space requirements for the four uses vary from one activity to another, no subdivisions or headings are preprinted; the observer subdivides the grid in the manner most suitable to his purposes. Detail level of any and all entries in the grid can vary; only those details which point up significant operating conditions should be entered.

2, 3. Entries describing input and output characteristics should define the inputs and outputs; the two classes of entries are grouped separately. Key number series for inputs is from 2000 to 2999; for outputs, 3000 to 3999. Name of input or output should correspond to the name on a Message Sheet. Sources of inputs and destinations of outputs shown on the Activity Sheet are always in the environment external to the Activity Sheet. Volume of each input or output is expressed as quantity per unit time; if there are significant variations in the volume over the course of a year, these are spelled

out. Causes of volume variation (calendar cycles or seasonal fluctuations) are shown. Frequency per year is shown for each volume level given. If batching is an input or output characteristic, batch size and batch-size variations are shown. If any of these items require additional explanation not easily entered in the grid, footnotes are added.

4. File usage is defined in terms of the size of the file and the access characteristics. The key number references the specific usage in the activity flow diagram; the same physical file may be used in several ways. File name may also change from use to use. Quantity of messages in the file is given for average and peak conditions if the difference is significant; quantity shown may be for all messages in the file or for just the type of message involved in the operation. Type of access (random or cyclic) is shown for each operation which is keyed to the file. Access time may be given as a maximum allowable figure, as a minimum attainable figure, or as a mean operating figure.

5. A tabular format is used to display elapsed time; in the table, combinations of the several controlling conditions are laid out, and the elapsed time for each set of conditions is then entered for every critical operation. Conditions include average and peak days during both average and peak weeks or seasons under various operating conditions. Where elapsed time is not appropriate, the fact that another event takes place may be entered; under certain volume conditions, for example, sub-contracted assistance may be sought. Function name for the elapsed-time tabulation describes a sequence of operations. Frequency is given for the operations within this subset. Key-number designations for inputs or outputs cross-reference the definitions of the inputs and outputs. A table for the various operations is set up and the elapsed times entered; these may be totaled for long operating sequences.

6. The LEGEND box on the Activity Sheet provides space to define the organizational components abbreviated in the operations boxes, and to give the physical locations if pertinent. Key-number series specifications are preprinted in this box. Here, too, the observer enters his name, the date the Activity Sheet was completed, the client's name, and the identifying name of the activity.

7. Footnotes are entered in the three NOTES boxes. Numbers are assigned to footnote references from the series 1000 to 1999 as the necessity to use footnotes arises.

FUNCTION: POSTING A/R										INPUTS									
FREQUENCY: DAILY										KEY NAME SOURCE									
INPUTS: 2000										2000 SALE CUSTOMER									
CONDITIONS										FREQUENCY									
DAY	AVG.	AVG.	PEAK	PEAK	AVG.	AVG.	PEAK	PEAK	2010	RETURN	CUSTOMER	FREQUENCY	2020	PAYMENT	2030	FILE	2040	DUPLICATE	2050
WEEK	AVG.	AVG.	AVG.	AVG.	PEAK	PEAK	PEAK	PEAK	2010	RETURN	CUSTOMER	FREQUENCY	2020	PAYMENT	2030	FILE	2040	DUPLICATE	2050
BILLED?	NO	YES	NO	YES	NO	YES	NO	YES	2020	PAYMENT	2030	FILE	2040	DUPLICATE	2050	FILE	2040	DUPLICATE	2050
1-3	1D	1D	1D	1D	1D	1D	1D	1D	2020	PAYMENT	2030	FILE	2040	DUPLICATE	2050	FILE	2040	DUPLICATE	2050
3-6	1/2D	1/4D	1D	1/2D	1D	3/4D	2D	1/2D	2030	FILE	2040	DUPLICATE	2050	FILE	2040	DUPLICATE	2050	FILE	2040
7-12	1/2D	1/4D	2D	1/2D						FILE	2040	DUPLICATE	2050	FILE	2040	DUPLICATE	2050	FILE	2040
13-18										FILE	2040	DUPLICATE	2050	FILE	2040	DUPLICATE	2050	FILE	2040
19-24										FILE	2040	DUPLICATE	2050	FILE	2040	DUPLICATE	2050	FILE	2040
25-26										FILE	2040	DUPLICATE	2050	FILE	2040	DUPLICATE	2050	FILE	2040
REMAINDER FILLED IN THE SAME WAY										REMAINDER FILLED IN THE SAME WAY									
FUNCTION: BILLING										OUTPUTS									
FREQUENCY: 20 DAYS PER MONTH										KEY NAME DESTINATION									
OUTPUTS: 3000										3000 BILLS CUSTOMER									
CONDITIONS										FREQUENCY									
DAY	AVG.	AVG.	PEAK	PEAK	AVG.	AVG.	PEAK	PEAK	2010	RETURN	CUSTOMER	FREQUENCY	2020	PAYMENT	2030	FILE	2040	DUPLICATE	2050
WEEK	AVG.	AVG.	AVG.	AVG.	PEAK	PEAK	PEAK	PEAK	2010	RETURN	CUSTOMER	FREQUENCY	2020	PAYMENT	2030	FILE	2040	DUPLICATE	2050
BILLED?	NO	YES	NO	YES	NO	YES	NO	YES	2020	PAYMENT	2030	FILE	2040	DUPLICATE	2050	FILE	2040	DUPLICATE	2050

NOTES

1010 - UNIT AUTHORIZER IS SUPERVISOR OR CREDIT MANAGER IF NECESSARY, DEPENDING ON AMOUNT OF SALE.

1020 - UNDER PRESENT SYSTEM, MEDIA FROM A REGISTER INCLUDES CASH SALES, CHARGES, COD, LAYAWAYS, VOIDS, ETC. ALL MEDIA FROM REGISTER ARE BALANCED TO THE REGISTER TOTAL; TOTALS ARE DEVELOPED FOR MERCHANDISE AND NON-MERCHANDISE CLASSES, AND MEDIA SEPARATED BY TYPE. CAUTION SHOULD BE USED IN ANY ATTEMPT TO SEPARATE THESE OPERATIONS FROM THE TOTAL JOB.

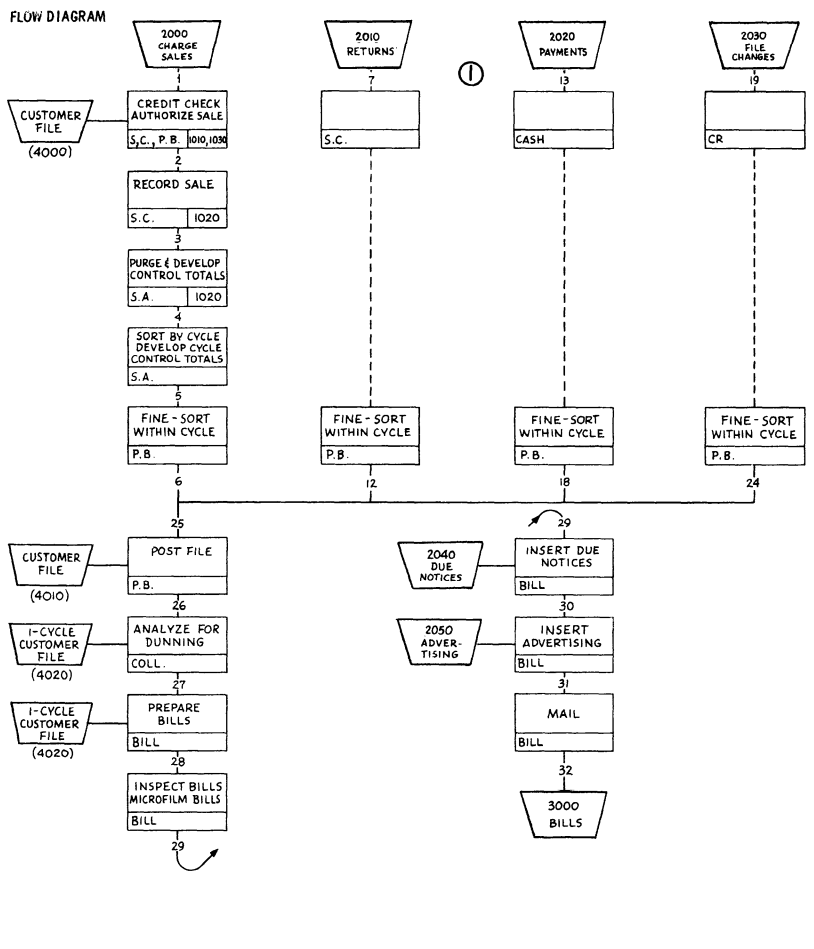
1030 - AT PRESENT, ONLY 60% OF CHARGE SALE TRANSACTIONS ARE AUTHORIZED WITHOUT APPROVAL OF UNIT AUTHORIZER.

NOTES

1040 - THERE ARE 302 WORKING DAYS PER YEAR IN THIS OPERATION.

1050 - THERE ARE 240 WORKING DAYS PER YEAR IN THIS OPERATION.

1060 - DELAY IN BILLING DURING THE CHRISTMAS PEAK RAISES ACCTS. RECEIVABLE INVESTMENT BY \$500,000.



NOTES

LEGEND BOX

ORGANIZATIONAL COMPONENTS

- S.C. - SALES CLERKS
- S.A. - SALES AUDIT
- P.B. - PRE-BILLING
- COLL. - COLLECTIONS
- BILL. - BILLING
- CASH. - CASHIER
- CR. - CREDIT OFFICE

KEY NUMBERS

- 1 - 999 Between Operations
- 1000 - 1999 Notes
- 2000 - 2999 Inputs
- 3000 - 3999 Outputs
- 4000 - 4999 File Usage

ACTIVITY SHEET

Analyst LH BAKER, JR. Activity Name

Date 29 FEB 61

Study ATLANTIC DISTRIBUTORS, INC. CHARGE ACCOUNTS

OPERATION SHEET

The Operation Sheet describes triggers, inputs and outputs; the processing steps that make up each operation; and the resources used in each operation. It explodes the operation boxes from the Activity Sheet. In many cases, the Operation Sheets are made up before the Activity Sheets; it is from the Operation Sheets that the flow diagram is produced.

1. OPERATION: This field consists of two subfields.
2. NO.: This is a six-digit operation number. The first three digits refer either to the department which performs the operation, or to the activity which includes the operation. The last three digits form the unique operation number. Operation numbers need not be consecutive.
3. PERFORMED BY: The name or number of the organizational component responsible for performing the operation is entered here.
4. TRIGGERS, INPUTS AND OUTPUTS: This field consists of five subfields.
5. ID. NO.: This differentiates among triggers, inputs and outputs. T1 identifies the trigger for each operation. Inputs are identified by a series starting with I1. Outputs are identified by a series starting with R1.
6. NAME AND QUALIFICATION: The name of an input or output must correspond to that appearing on a related Message Sheet. Copies or parts of copies involved are indicated in parentheses after the name. If only a selected group of a named input or output is involved, the group qualifications are entered; the qualifying statement specifies constraining values. One and only one trigger must be defined for each operation. Triggers may fall in any of four categories: receipt of an input; a time of day, week, or month, or a frequency per unit time; receipt of multiple inputs; or a combination of inputs and times.

7. RECEIVED FROM OR SENT TO: Enter here the name or the number (or both) of the organizational component from which the input is received or to which the output is sent. If outside the activity under scrutiny, indicate the sender or receiver in general terms: customer, vendor, salesman, government, etc.; if inside, use the lowest-level component involved. If an output is sent solely for filing, indicate destination file by name.

8, 9. VOLUME: Indicate the average quantity for each input, output, and multiple-input trigger, during any given time period.

8. AVG.: Use the arithmetic mean of the volume figures.

9. PER.: In defining time periods, choose a meaningful length of time. Since this data helps determine costs, the time period should be consistent, if possible, across the operations and with the client's costing and accounting policies. Consider cyclic concepts in making entries for such cases as file processing.

10. ELAPSED TIME: This is determined as follows: A zero is entered for the input which normally arrives first; if more than one type of input is needed to start the process, average elapsed time between availability of the first input and availability of subsequent inputs is shown for each subsequent input. For each output, list the average elapsed time between time zero (the availability of the first input) and the arrival of the output at its destination. Where practical, elapsed time should be given in working days or fractions of a working day.

11. PROCESSES: This field includes three subfields.

12. ID. NO.: The identification number segregates the processes that make up the operation. Processes are identified by a number series starting with P1.

13. DESCRIPTION AND QUALIFICATIONS: This subfield consists of a verb and its object, plus occasional qualifying phrases. Verbs should be broad enough to obviate the need for details, clear enough to avoid ambiguity. Such verbs as determine, prepare, reproduce, insert, attach, select, post, arrange, edit, adjust, reject, destroy have been adequately defined and exemplify the type of verb to use. The object of the verb should refer to inputs, outputs or resources, and should tell what information is affected or transformed by the action implicit in the verb. Additional adverbial (to . . . , for . . . , into . . . , etc.) or gerundive (using . . . , etc.) phrases are entered as needed to explain the process.

Conditional clauses (if . . . , when . . . , etc.) are also employed.

14. FREQ.: This field gives the frequency of performance in terms, if possible, of the number of executions per input, output or operation. Frequency may also be expressed as unit operating time, or as a statement of the cycle: MC would indicate an operation performed on a monthly cycle.

15. RESOURCES: This field includes four subfields.

16. ID. NO.: This segregates the various resources used in the operation. Resources for each operation are identified by a separate number series starting in each case with X1.

17. TYPE: Here are shown the classifications of operating personnel and machines or equipment, with the numbers of each in parentheses. Names and classifications of materials are entered, with appropriate volumes; raw materials, work in process, forms, finished goods, all are listed. Resource data includes files; these are identified by name and type of message, using names and numbers corresponding to entries on Message and File Sheets.

18. UNIT TIME: Resource usage per operation is defined here. For a given resource, unit time may be separately defined for each process and again for the whole operation. The figure should represent average time for the total resource named. Thus, ten minutes from each of three clerks would demand an entry 1/2 H/OP under UNIT TIME opposite the entry CLERKS (3) in the TYPE subfield. Unit time is not given for materials, nor, in most instances, for files.

19, 20. TOTAL TIME: This defines the total resource usage per process or operation over a selected time period. The entry represents the average time that the total resource is employed; unit of time selected should be compatible across operations and with the client's accounting policies.

19. AVG.: The number and type of time unit are shown.

20. PER.: List the selected time period.

MESSAGE SHEET

The Message Sheet displays fine detail on inputs and outputs. Entries in the various fields describe and define the message characteristics; a copy of the message is usually appended to the form.

1. MESSAGE NAME: This is a unique name for the message, formed from letters, hyphens, and blanks; numbers may not be used. Hyphens must be preceded and followed by at least one alphabetic character.

2. MESSAGE NO.: This is the unique identifying number for the message, with prefix indicating message type (R for report and S for signal) and suffix numerals indicating copies. The designation R 131.3 identifies the third copy of message 131.

3. OTHER NAMES USED: Frequently used alternate names and synonyms are listed.

4. LAYOUT NO.: Layout is the physical format of a message; each format is identified by a layout number. The layout is independent of both the messages using the format and the data within the format. Several message types may use the same layout number, with the fields differently interpreted on each. Layout number defines physical locations of fields and field characteristics, permitting relatively flexible form design for a series of messages having the same layout number.

5. FORM NO.: This number is assigned by the client to identify the message form; it is usually printed on the form.

6. NO. OF COPIES: This figure includes both the number of copies prepared with the original and the number of copies later reproduced.

7. MEDIA: This entry displays the media employed in original and later reproduction.

8. HOW PREPARED: This pertains to reports, and describes the means by which basic fields are entered on the original message (by hand, type-written, card-punched, etc.).

9. OPERATIONS INVOLVED IN: This is the identifying number of each operation using the message as input or output.

10. REMARKS: Enter here all supplemental data on message definition or handling (security, access, and so forth).

11. NO.: The identifying number of the data element can be any number from 01 to 99. The message number and this number uniquely identify the data element for later reference.

12. DATA NAME: This is the title for the data element. Names may be formed from letters of the alphabet, hyphens, or blank spaces; numbers may not appear. Hyphens must be preceded and followed by at least one alphabetic character.

13. FREQUENCY: Three types of entries appear here. If the data does not appear on every message carrying this message number, the frequency of appearance is entered as a decimal fraction. If the data appears once on each message, the entry is 1; if more than once, the number of

OPERATION										TRIGGERS, INPUTS AND OUTPUTS				PROCESSES				RESOURCES					
NO	PERFORMED BY	ID NO	NAME AND QUALIFICATIONS				RECEIVED FROM OR SENT TO		VOLUME		ELAPSED TIME		ID NO	DESCRIPTION AND QUALIFICATIONS		FREQ.	ID NO	TYPE		UNIT TIME		TOTAL TIME	
(2)	(3)	(5)	(6)				(7)	(8)	(9)	(10)	(12)	(13)		(14)	(16)	(17)		(18)	(19)	(20)			

Figure 22. Operation Sheet

appearances, or the possible range and average, is entered. Whenever a range is entered, the average must also be noted.

14. CHARACTERS: This field shows the maximum number of characters in the data element.

15. A/N: Enter A for alphabetic, N for numeric,

and AN for alphanumeric (including special symbols).

16. ORIGIN: Enter the operation number for the operation that either initially accepts the data element into the system, or originates the data. Operations that merely post the data to the message are not entered.

IBM

Message Sheet

MESSAGE NAME (1)		MESSAGE NO. (2)			
OTHER NAMES USED (3)		LAYOUT NO. (4)			
		FORM NO. (5)			
		NO. OF COPIES (6)			
MEDIA (7)	HOW PREPARED (8)				
OPERATIONS INVOLVED IN (9)					
REMARKS (10)					
CONTENTS					
NO. (11)	DATA NAME (12)	FREQUENCY (13)	CHARACTERS (14)	A/N (15)	ORIGIN (16)

DATE _____ ANALYST _____ SOURCE _____ PAGE _____
STUDY _____ 808-4894-0

Message Sheet

FILE SHEET

The File Sheet is used to display fine detail on information resources. Entries in the various fields describe and define the file characteristics.

1. **FILE NAME:** There is a unique name for each file, formed from letters, hyphens or blanks; numbers may not be used.
2. **FILE NO.:** There is also a unique number

for the file, with prefix F and suffixes indicating copies. The designations F 131.2 and F 131.4.2 identify copies 2 and 4 of file 131; copy 4 is located at a site "2" remote from the main file.

3. **LOCATION:** The name or number is given for the organization housing the file (or portion of the file) and the physical location if pertinent.

IBM

File Sheet

FILE NAME (1)			FILE NO. (2)			
LOCATION (3)		STORAGE MEDIUM (4)				
ACCESS REQUIREMENTS (5)						
SEQUENCED BY (6)						
CONTENT QUALIFICATIONS (7)						
HOW CURRENT (8)						
RETENTION CHARACTERISTICS (9)						
LABELS (10)						
REMARKS (11)						
CONTENTS						
SEQUENCE NO. (12)	MESSAGE NAME (13)	VOLUME		CHARACTERS PER MESSAGE (16)	CHARACTERS PER FILE	
		AVG (14)	PEAK (15)		AVG (17)	PEAK (18)

DATE ANALYST SOURCE PAGE
STUDY

808-4603-0

File Sheet

4. STORAGE MEDIUM: This is the type of housing for the file, such as tub file, tape storage cabinet, three-ring binder, etc. This entry is indirectly related to the medium of the information itself.

5. ACCESS REQUIREMENTS: Several types of information appear here: who is or is not permitted access to the file, classified by job titles or by such entries as "Military — Top Secret" or "Company Confidential"; the availability of the file, in terms of what hours and how long the file is open daily; and access characteristics, including how often and how quickly reference must be made.

6. SEQUENCED BY: File sequence keys are described in this field. File sequence is described by minor key within intermediate keys within major key. A file of open purchase orders might, for example, be sequenced by transaction date within purchase order number within part number. Sequence keys are sometimes not contained in the messages themselves, yet must be described in this field. In the case of the purchase orders, transaction date might be missing from the messages; new transactions would be filed in back of existing transactions within the purchase order number and part number sequences.

7. CONTENT QUALIFICATIONS: Details are displayed on file contents if file name is not sufficiently descriptive. A file named "Purchase Order File" might, for example, be qualified as "purchase orders for vendors within 25 miles."

8. HOW CURRENT: This gives the age of transactions when entered in the file.

9. RETENTION CHARACTERISTICS: Removal rules for each type of message in the file are entered here.

10. LABELS: These identify the file, carrying a code or phrase, such as "Master Payroll" to

uniquely establish the file identity. Other information, such as date, number of records, and so forth, is often carried as well. Entry is particularly used for magnetic tape files.

11. REMARKS: Noted here are miscellaneous data, and problems such as rapidly expanding size, excessive or inadequate retention cycles, or need for duplicate files differently sequenced.

12. SEQUENCE NO.: This gives the relative sequence number for order of messages within the sequence keys of the file (for files in which multiple records are filed together). If report A and report B are to be processed for a common master report C, and report A must be filed in front of report B, then it is given sequence number 1 and report B is given sequence number 2.

13. MESSAGE NAME: This shows name and number of messages appearing in the file. The name should correspond to the name on a related Message Sheet.

14. (VOLUME) AVG.: The average number of this type of message in the file is shown.

15. (VOLUME) PEAK: The peak number of this type of message in the file is shown.

16. CHARACTERS PER MESSAGE: This field displays the size of an average message. The number entered here is the total of each data element's character count multiplied by frequency; if a range and average appear, the average is used.

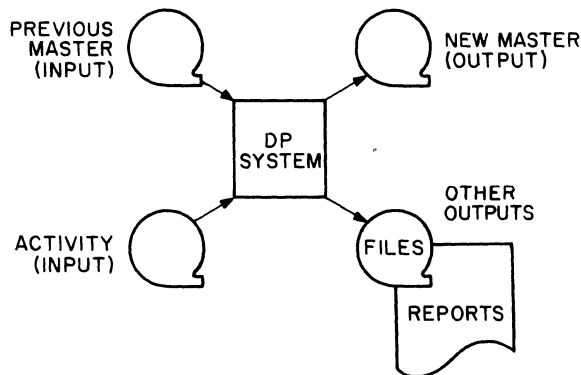
17. (CHARACTERS PER FILE) AVG.: The average file size for this message is given. Multiply the character count per message by the volume average to arrive at this figure.

18. (CHARACTERS PER FILE) PEAK: The peak file size for this message is given. Multiply character count per message by volume peak to arrive at this figure.

SEQUENTIAL FILE RUN DESIGN

A sequential file run as considered here is the operation of a program designed to update and normally revise an ordered master file contained on one or more reels of magnetic tape. During the job a new master file is created from the previous master file after additions, deletions and changes have been made. The basic input to the run is the previous master file and one or more activity files. The output consists of a new master file and additional files or reports as required.

Normal Sequential File Run Design



The above diagram shows the most frequently encountered sequential file program design. Even where there is no activity to include, the master file is duplicated as it is updated. The design assumption is that activity will be present for most records on the master file, and the I/O and process time balanced.

ALTERNATIVE RUN DESIGNS

Some of the variables to be considered in designing a program for a sequential file are:

1. The capacity and features of the data processing system
2. The restrictions imposed on the blocking factor of the master and activity files
3. The process time required for each addition, deletion and change
4. The degree of summarization required for the master file and available in the activity file
5. The relative size of the master and activity files
6. The occurrence of activity (Is most updating done in a small part of the master file? Does it occur in irregular bursts or in a random manner?)

The following alternative approaches may prove more effective in some situations than the standard design.

Split File

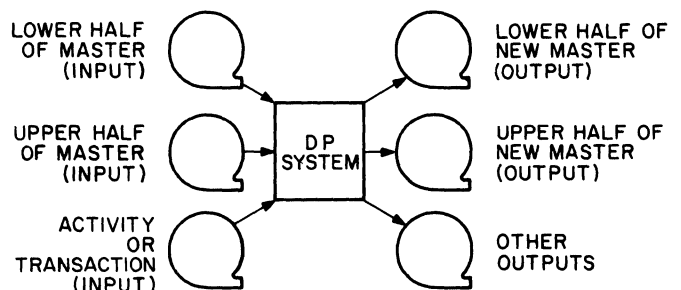
The split file technique requires dividing the master file into two or more segments. The activity file may be arranged on separate files to coordinate with the split masters, or they may be intermixed on a single file. The output, other than the new masters, will not be in sequence unless multiple output tapes are provided.

This design is suitable for configurations with appropriate buffers and channels, and provides for parallel rather than sequential searching.

The split file design compensates for (1) high process time requirements for additions, deletions and changes, (2) irregular activity distribution, and (3) multiple activity records for a single master record.

Additional components may be required to implement this design, and file-blocking factors may have to be reduced.

Split File



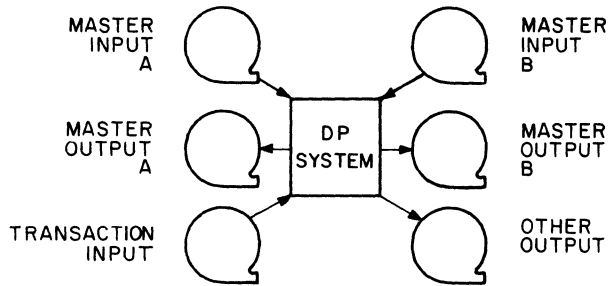
Alternating File

The alternating file design requires assigning alternate blocks of master records to master files A and B. Master files A and B are read and written alternately on separate channels. Two or more buffered channels are required.

This design should be considered when the run has significant periods during which records are being only added or only deleted from the master files. During additions, the master input files are idle, and the output files active. During deletions, the reverse occurs.

Alternating file run design may require additional components, changes in blocking factors, and relatively complex input/output programming to balance the parts of the master files.

Alternating File



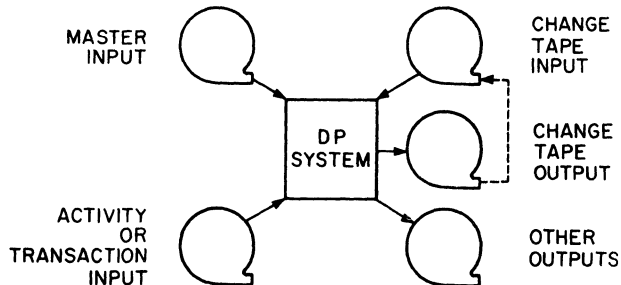
Change Tape

In the change tape design only active records have to be written on the tape. The processing consists of reading both the master file and the change tape, and writing a new change tape. Thus, the change tape is updated from the active records, and records from the master file are added only when they have activity. The advantage is that the entire master file is not rewritten during every run.

The change tape increases in size over a period of time according to the amount of activity. When the tape becomes approximately half the size of the master file, a special run is necessary to update the complete master file by direct replacement from the change tape.

This design should be considered when a small percentage of the master file records have a large percentage of the activity.

Change Tape



Activity Tape

The activity tape design requires two separate runs, one of which may be done on a peripheral computer or at a different time. The processing in the first run consists of extracting active master records

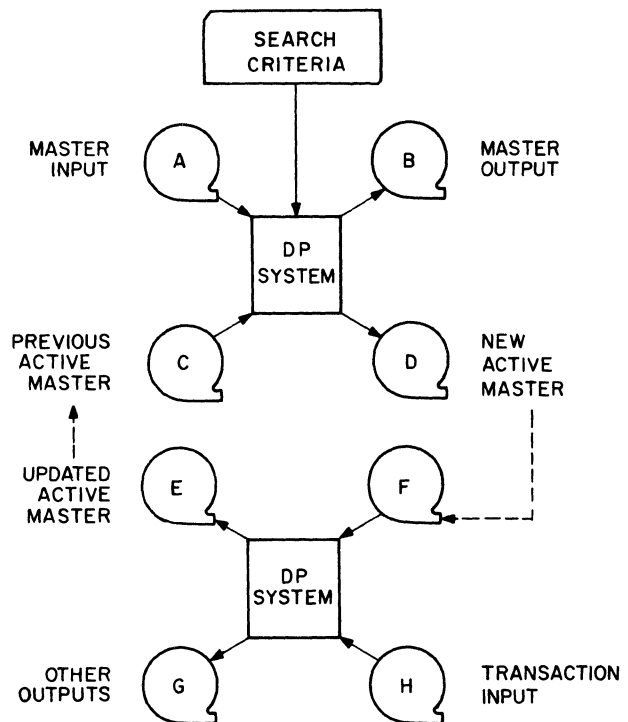
based on search criteria generated cyclically or in a preliminary document-to-tape input run. The input to the first program consists of the master records not used in the previous run (A) and the extracted master records used in the previous run (C). These are internally merged and compared with the search criteria. The master records required are separated (D) from those with no activity (B).

The second run is a standard update with the exception that the master file is an extract (F and E) that is subsequently merged again into the master file (A).

This method should be considered when only a small part of the master file has activity at any one time, and when the update processing allows only low blocking factors due to storage limitations.

The cost of this design is higher because the active master records are read and written three extra times. The advantages include extra flexibility in scheduling and the efficient use of a peripheral system.

Activity Tape

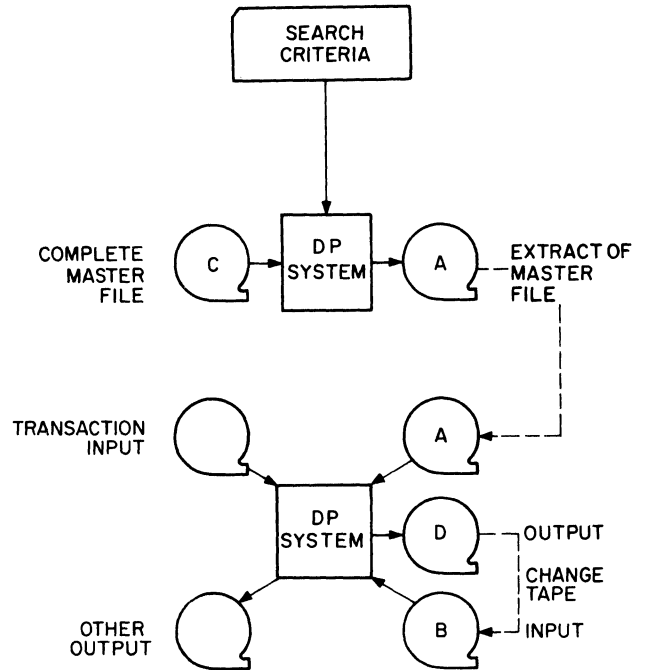


Activity/Change Tape

The technique used in the activity/change tape design involves two runs. The first run copies records from the complete master file (C) based on search criteria. The second run matches the extracted master records (A) with those on the change tape (B). If the extracted master record is already on the change tape, the more current change tape record is used. The other processing in the second run is a normal updating.

Periodically, the current records on the change tape (D) are used to update the master file (C) by direct replacement. This keeps the change tape file small enough for efficient processing.

Activity/Change Tape





OPTIMUM BLOCKING FOR UNBUFFERED SYSTEMS

To minimize total tape passing time for a run, available storage for input and output areas is allocated by the following procedure

Find for each file:

B - the optimum number of records per tape block.

Given:

V - volume of records in the file.

S - start/stop time in seconds for the tape unit on which the file is mounted.

W - the average record length expressed as words or characters.

N - the number of input/output areas to be allocated for all files.

M - the total core storage available for input/output areas.

1. If the number of records per block is fixed by external considerations such as communication with another system, allocate the necessary storage for the block and reduce the total available core storage, M, accordingly.

2. Examine the files for dependent relationships. A dependent relationship exists when two files, such as a master input file and a master output file, must have the same blocking factor. If a dependent relationship exists, use step 3 to calculate A and C. Where an independent relationship exists, calculate A and C from step 4.

3. For dependent files, where one file has the parameters V, S and W, and the other v, s and w, calculate:

$$A = \sqrt{\frac{2(VS + vs)}{N(W + w)}}$$

$$C = \sqrt{\frac{N}{2}(VS + vs)(W + w)}$$

Note: If V, S and W are the same for both files (the usual case) the above equations become:

$$A = \sqrt{\frac{2VS}{WN}}$$

$$C = \sqrt{2VSWN}$$

4. For each independent file, calculate:

$$A = \sqrt{\frac{VS}{WN}}$$

$$C = \sqrt{VSWN}$$

5. Calculate:

$D = \sum C =$ The sum of the C's calculated in steps 3 and 4.

6. Calculate:

$$K = \frac{M}{D}$$

7. Calculate the optimum number of records per tape block B -- from:

$$B = KA$$

using the appropriate value of A for each file as calculated in step 3 and 4.

8. Round results. Calculate:

$$M = \sum BWN \text{ for all files}$$

This will determine whether the blocking factors chosen actually fit the available core storage.

9. If the rounding process has resulted in a requirement for more storage than is available, calculate for each file:

$$E = \frac{VS}{B(B-1)}$$

Reduce by 1 the blocking factor on the files with the smallest value of E until the total storage required is within the value of M.

10. If the rounding process has resulted in some of the available storage being unused, calculate:

$$E = \frac{VS}{B(B+1)}$$

Increase by 1 the blocking factor on the files with the largest value of E until all available storage is used.



PLANNING A DATA PROCESSING ORGANIZATION

A carefully designed organization structure, based on clear delegation of authority and responsibility, is a prime requisite for the effective operation of a data processing section. Planning for the organization begins early in the implementation stage; a structure is established, position descriptions are written, and personnel are recruited and selected to fill these positions.

A data processing organization is developed in five basic steps:

1. Identification of data processing goals and objectives as they apply to a specific business.
2. Designation of the data processing functions or work elements necessary to fulfill these objectives.
3. Grouping individual functions or work elements into personnel positions in accordance with recognized organization criteria.
4. Preparation of complete job descriptions (work content, authority, accountability, etc.) for each position.
5. Documentation of the structure in an organization chart.

Data processing organizations vary considerably in structure among users as they reflect differences in business size, management philosophy, and scope of the work. Further, organization is a dynamic process and will change as requirements of the business change. The fundamental structure should accommodate present needs during the stages of implementation and operation, but be flexible enough to anticipate and handle future growth and change.

ORGANIZATION PATTERNS

A data processing organization, then, should be tailored to the unique demands of a specific business. Figures 1, 2 and 3 represent the organization of large, medium and small companies, respectively. These are not recommended structures; they are simply illustrative of typical situations and are shown here mainly to provide a framework for describing the content of key data processing positions and demonstrating the applicability of certain organization principles and practices.

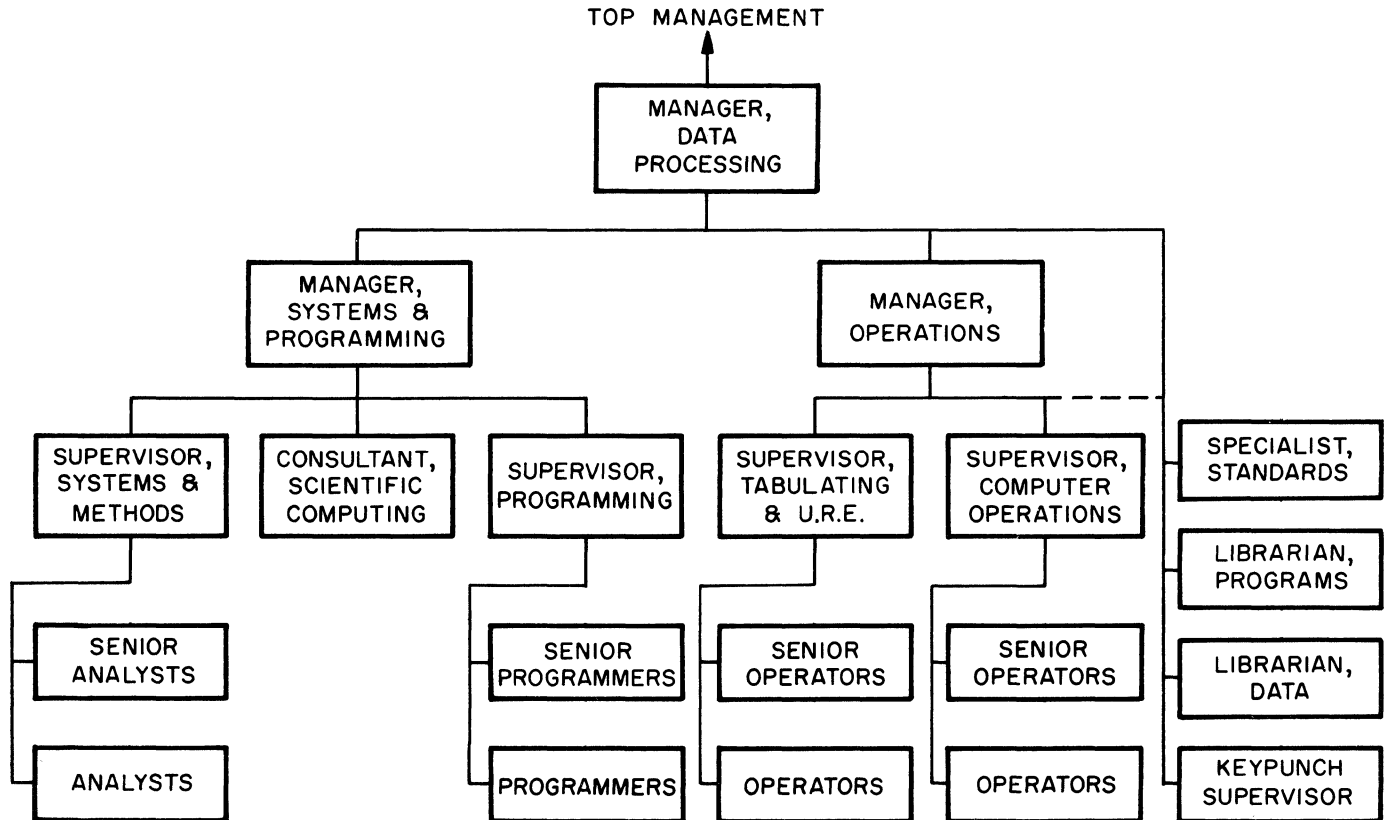


Figure 1. Organization structure -- large installation

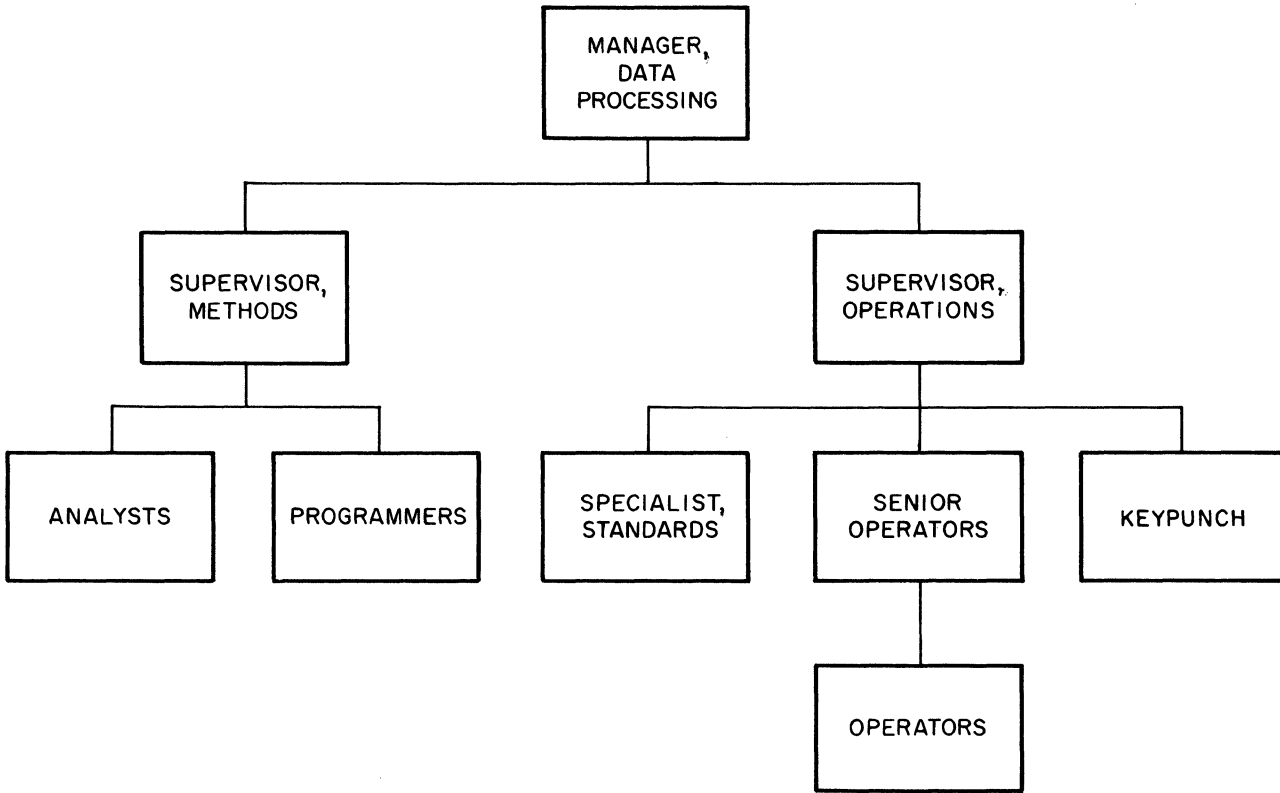
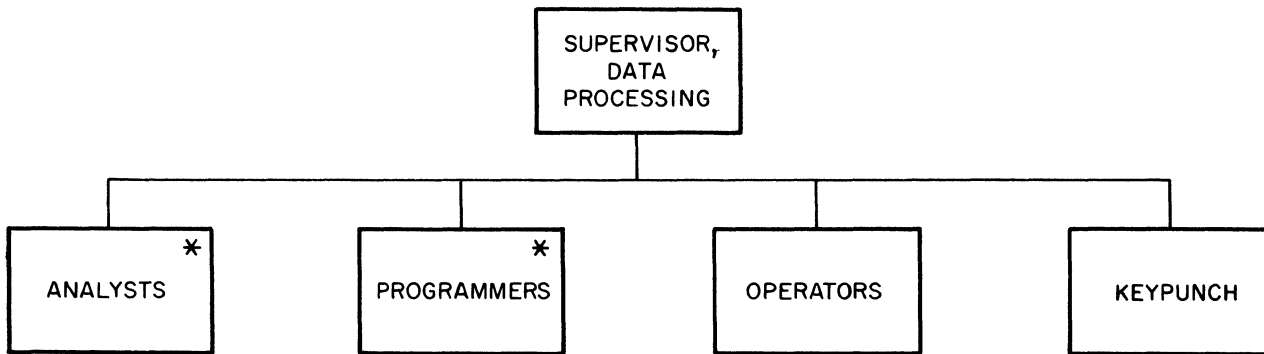


Figure 2. Organization structure -- medium installation



* SOMETIMES COMBINED

Figure 3. Organization structure -- small installation

ORGANIZATION CONCEPTS

Although organization concepts are beyond the scope of this discussion, a few of them should be mentioned to guide the systems engineer in his work:

1. Planning is usually separated from operating work (that is, long range vs. short range). This principle is illustrated in Figure 1, where the manager of systems and programming is assigned planning work, and the manager of operations is responsible for day-to-day operating work.

2. A manager is assigned a span of responsibility in keeping with his ability to properly administer the personnel and the work.

3. The relative degree of integration within the department is determined by the nature of the business and management policies. Thus, in a large, decentralized, multi-plant business, a decision has to be reached on whether remotely located keypunch operators are part of the data processing or the functional organization. A decision also has to be reached on whether programming is to be performed on an open- or closed-shop basis.

The application of these and other tested organization practices is important in the development of logically structured, efficient organizations.

KEY POSITIONS

To maintain a consistent approach, the following description of individual data processing jobs and the relationship of one position to another is based on the data processing organization of a large company, where job specialization is more sharply outlined because of the greater number of positions. Small organizations must perform the same work, but several different functions may be combined into a single position.

Manager -- Data Processing

- Plans, supervises and coordinates electronic data processing analysis, systems design, programming and machine operations.
- Represents data processing department to top management.
- Interprets company policies and directives for data processing personnel.
- Motivates personnel to the accomplishment of established objectives.
- Maintains or approves contacts with other departments.

- Evaluates departmental performance against plans and goals.
- Determines optimum personnel practices, manpower levels, budget requirements and training programs.
- Evaluates feasibility of potential applications, and approves of the amount and type of equipment required to perform the applications.

Manager -- Systems and Programming

- Interprets broad areas to be investigated, general aims of the methods group, and proves feasibility of application projects (includes economic as well as systems and equipment analyses).
- Supervises planning, design, coding and documentation of projects to insure their proper execution.
- Reviews and coordinates procedures employed in implementing a project, and recommends appropriate procedures to support job completion on schedule.
- Endorses application of the latest machine programming systems and systems techniques.
- Maintains liaison between methods and operating groups.

Manager -- Operations

- Plans and administers the various operating sections of the department: computer, tabulating and keypunch.
- Maintains efficient utilization of computing equipment and personnel resources.
- Establishes schedules to meet operating requirements, and determines alternate courses of action as schedules change.
- Verifies adherence to prescribed operating rules and regulations.
- Coordinates the activities of areas under his control, such as keypunching, input/output data control, tape library, standards (the latter two may, for example, report directly to the data processing manager in some instances).
- Determines internal report requirements (for example, machine utilization).
- Sets acceptance standards for production programs and documentation received from the methods group.
- Provides proper training for operating personnel.
- Evaluates quality of personnel and procedures and makes adjustments as necessary.

Consultant -- Scientific Computing

- Applies scientific and operations research techniques where valuable to improve business performance.
- Serves as liaison between data processing and scientific or operations research groups.
- Outlines and develops scientific projects to be handled by data processing department. (Methods is responsible for implementation.)
- Keeps abreast of new developments in scientific computing techniques.

Supervisor -- Systems and Methods

- Supervises assigned projects and coordinates work of the analysts.
- Works with the manager of systems and programming to determine project completion dates and personnel requirements.
- Communicates with other departments on project requirements.

Supervisor -- Programming

- Plans, schedules and supervises preparation of programs to solve technical and mathematical problems, or to process business data.
- Assigns, outlines and coordinates work of programmers.
- Trains subordinates to prepare and write programs, routines, flowcharts and diagrams:
- Reviews, evaluates and reports performance.
- Collaborates with supervisor of systems and methods on schedules and projects.

Specialist -- Standards

- Establishes documentation standards and recommends techniques and methods to secure adequate and consistent documentation.
- Prepares standards instructions on such subjects as record formats, routines, labels, audit and machine controls.
- Updates and maintains IBM programming systems, and evaluates new programming languages and documentation techniques as they become available. (This responsibility might be assigned instead to a selected senior programmer.)
- Validation and editing of input and output to and from the department, and isolation of errors. (This responsibility could also be assigned to operating personnel.)

Note: In some installations responsibility for standards and quality control may be delegated to separate individuals in both the systems and operations groups; in

others, the responsibility may be absorbed in several positions.

Librarian -- Programs

- Maintains all documentation and program information generated by the programming group.
- Maintains literature on installed and proposed equipment.
- Records issuance and receipt of programs, and related data.

Librarian -- Data

- Maintains master and operating decks and operating instructions for programs.
- Issues and receives tapes for daily program runs.
- Prepares operating statistics on such subjects as tape usage, etc.

Other Positions

A maintenance specialist or supervisor might be included in larger organizations to improve the efficiency of operating programs. This work could also be performed by the standards specialist. An operations coordinator is carried in some organizations to maintain work flow through keypunch, tabulating and computing equipment sections. Usually, though, this interfunctional responsibility is assigned to someone in the operations group as an addition to his other work.

Job descriptions for analysts, programmers, operators, etc., can be found in the referenced literature.

Regardless of the approach to organization structuring, the final result should:

1. Promote good vertical and horizontal communication within the business.
2. Provide balanced, substantive content for each data processing position.

REFERENCES

Administration of Electronic Data Processing, National Industrial Conference Board Business Policy Study #98, 1961.

Planning for an IBM Data Processing System, general information manual (F20-6088).

Management of the IBM Department (225-5827-1).

Performance Analysis of Data Processing Services, general information manual (E20-8110-1).

Manuals of Procedure (225-3707-4).

PROGRAM TESTING CHECK LIST

If program testing techniques are properly planned, a minimum amount of machine time is consumed during program checkout. Manual inquiry into a system via the console is extremely expensive in machine and operator cost; little is learned in return for dollars expended. Time spent in desk checking is well invested since the bulk of the logical errors may be detected before the program actually enters the computer testing phase.

In trying to make the maximum number of runs during a test session, the programmer is tempted to make rapid patches without pausing to annotate such changes thoroughly. Such urgency is seldom fruitful in the end.

The program testing phase should be carefully and thoroughly planned, executed and documented. The following check list should be used as a guide to insure maximum productivity for program testing.

After coding the program and preparing revised flowcharts and other supporting documentation, the testing procedure begins.

1. Prepare test data and precalculate results.
2. Punch and verify program cards.
3. List program cards by:
 - a. Label
 - b. Operation code
 - c. Page and line number
4. Desk check the program. Look for:
 - a. Errors in logic
 - b. Endless loops
 - c. Incorrect use of program switches
 - d. Unsatisfied or incomplete coding for problem definition
 - e. Inefficient program (time and storage)
 - f. Incorrect data field lengths
 - g. Improperly limited fields (word marks)
 - h. Improperly signed fields
 - i. A label for each operand
 - j. Improper indexing
 - k. Initialization of routines and storage
 - l. Duplicate labels
 - m. Misspelling and punching errors
 - n. Invalid operation codes
 - o. Necessary control cards
 - p. Improper alignment of card columns
5. Manually simulate the computer process using test data.
6. Assemble the program.
7. Perform error analysis with error listing and master program printout.

8. Correct program

- a. Card programs. -- When the object program exists on cards, error correction can be accomplished by either patching the object deck or correcting the source deck. If there are a large number of corrections to the program, it is usually best to correct the source deck and then recompile the program. Otherwise, patching of the object deck is preferable.

It is recommended that patching be accomplished by overlaying the instructions to be corrected rather than by attempting to correct or modify the original object program cards. The object deck created by the assembly should remain intact.

To facilitate card handling, label the object deck with a marking pen. The first cards produced by the processor are usually the load program and should be so designated. The first and last card of the object deck should be so labeled. The last card produced by the assembly is the Execute card, and should be labeled as such. The top edge of such cards may also be marked.

- b. Tape programs. -- When the program is prepared on tape, corrections are usually made to the source deck. This is accomplished by placing the corrections in the source deck and then recompiling the program.

Alter the master program listing and update the program flowchart to reflect patch or source deck, or tape, corrections.

9. Prepare detailed instructions for machine operation during the test session.
10. Pretest session familiarization
 - a. Console operation
 - b. Input/output devices
 - c. IOCS
 - d. Utility routines such as programmed automatic test programs (like Auto-Test), clear storage and load programs, file generators, trace programs, storage, tape, and disk print programs, sort and merge programs, and checkpoint and restart programs.

11. Test documentation and materials. To reduce confusion, all materials should be clearly labeled with name of user organization, program name, content, and date. Each person should have a list of the items for which he is responsible:

- a. Program flowcharts
- b. Assembled program listings
- c. Test data decks and tapes with test data listings (a duplicate copy may be desirable)
- d. Precalculated results of test data and listing of expected output with each test case
- e. Tape, card, storage, and disk record layouts
- f. Internal storage map
- g. Printer carriage tape
- h. Operator check list, which provides the operator with all the information required to set up the data processing system for running each program:
 - (1) Job or program name
 - (2) Operation name
 - (3) Machine setup
 - (a) tape unit or card stacker assignment
 - (b) input cards or tapes
 - (c) output cards or tapes
 - (d) carriage tape
 - (e) sense switch settings
 - (4) The sequence of events to run the test
 - (5) Listing of all possible messages and halts
 - (6) Switch and index register listings
 - (7) List of paper forms or card stock for auxiliary equipment
- i. Object deck or tape, with patches included
- j. Blank forms, cards, tapes (output or scratch if necessary)
- k. Source deck and listings by label, operation code, and page and line number

12. The test session

- a. Plan the test session in advance. Decide upon the sequence in which programs shall be tested. Programs should take precedence in testing according to their importance and the most important programs should be retested as often as possible until they are completely debugged. Schedule a workload greater than can be accomplished in the allotted

test time. Assign duties (such as handling the card reader, punch, printer, tape units, and console) to each person attending.

- b. Arrive early. Confirm the testing schedule that was established in advance of actual testing.
- c. Be familiar with the latest versions of all programming systems to be used.
- d. Use an automatic testing system if at all possible. Make certain that the test packet is organized properly. Test the higher priority and larger programs first. Each program must have its own input test data; a program should not be dependent upon one run earlier in the same session.
- e. If using tape units, make sure they are clean, and that the tape reels are properly file-protected.
- f. Make sure that all units are in the proper initial status -- for example, printer restored, tape units ready, no leftover cards in the reader or punch.
- g. Debugging at the console is time-consuming, error-prone and generally nonproductive. When the program hangs up, the following steps should be taken immediately:
 - (1) Note the console status -- indicators, lights and registers
 - (2) Take storage print
 - (3) Take tape and disk prints
 - (4) Go on to next program or cease work

Even if a program goes to end-of-job and appears correct, the above steps should be taken in order that the correction of errors which are discovered later may be simplified. When a program hangs up, do not force it on without taking down status information, since the conditions causing the original hangup would then be destroyed.

- h. Label all storage prints, tape prints, console sheet, etc., with date, time, and program identification.
- i. Debug off the console with deliberate speed. With the above items, there is more information to aid in locating the reason for the hangup than is available

Section	Area	Page
03	02	03

on the console. Do not make hurried corrections to a program in a false effort to maximize usage of test time.

Before testing the program again, find all possible bugs, not just the one which caused the hangup. Step further through the program after each test to ensure that the program will not hang up on the next instruction or routine. Correct all errors in output content and format. Strive for perfect output from each test.

- j. Note all corrections on the master program listing. Corrections which affect the halt, switch, or index register listings should be updated accordingly.
 - k. Note the reason for the correction on the patch card itself. Be sure to number and date patch cards. A post-test listing of patch cards is desirable for reference when correcting the source deck.
 - l. Reassemble after an appropriate number of patch cards have been added to the program. Update program flowchart to reflect the current status of the program.
 - m. Keep documentation current; this eliminates the waste of time and effort to try to pick up the changes during testing or debugging.
13. Post-test evaluation. Every test session should be followed by a thorough evaluation:
- a. Was the pretest preparation adequate?
 - b. Were there any areas of preparation that could be improved to yield a more effective test?
 - c. Were there areas of preparation in which we spent too much time and others in which we spent too little?
 - d. Did the test point up any areas of weakness in the coding? If so, are these types of errors documented so that stronger emphasis can be placed upon them during future coding and desk checking?
 - e. Was each machine session used effectively?
 - f. Are there any corrections to the testing techniques that would make the next test more fruitful?
 - g. What is the status of each program tested?
 - (1) Is it completely tested? That is, has every program loop been tested, and do we have any reservations about calling this program complete?
 - (2) Is it tested to the stage where the only changes left are in spacing and editing of the output data?
 - (3) Are there logic errors left in this program?
 - h. Did the test session achieve its objectives? If not, what adjustments in present scheduling are necessary?



PREINSTALLATION CHECK LIST

To implement a data processing system effectively and efficiently, many varied tasks must be properly defined, organized and performed. Strict adherence to defined procedures will enable an orderly and on-time installation.

The following check list is provided as a guide to the tasks involved in the preinstallation procedure. This procedure is usually formulated with respect to a time schedule of events, a target date being set for each event. This check list will define the tasks that are required for the completion of each event. The numbered topics represent events, and the lettered information under each topic represents the required tasks to complete the particular event.

1. Establishment of customer organization
 - a. Establish executive committee
 - b. Appoint project director
 - c. Select planning and programming staff
 - d. Prepare organization planning chart
2. Initial education program
 - a. Obtain pertinent manuals and instructional material
 - b. Present executive seminar and programming classes
 - c. Present programming classes for project director and programming staff
3. General system design
 - a. Establish preinstallation schedule of events (bar charts)
 - b. Define application, with the following items as the principal basis:
 - (1) Review of source documents
 - (2) Analysis of file requirements
 - (3) Determination of report requirements
 - (4) Establishment of due-in and due-out schedules
 - (5) Definition of procedures with regard to workflow
 - (6) Definition of required organizational changes
 - (7) Determination of requirements for operating systems personnel
 - c. Prepare general flowcharts, including:
 - (1) Program planning chart
 - (2) System planning chart
 - d. Determine all basic changes to existing system and agree on a plan to effect these changes.
4. Review of physical installation plans
 - a. Prepare physical planning chart
 - b. Determine space requirements
 - c. Determine power and air conditioning requirements
 - d. Design machine rooms
 - e. Determine cable requirements
 - f. Plan for room construction
 - g. Meet with customer representative and review all physical installation plans
5. Detailed system design
 - a. Prepare detailed flowcharts
 - b. Prepare detailed block diagram of major application
 - c. Determine detailed equipment specifications and supplies requirements
 - d. Prepare related systems work chart
 - e. Review savings and cost analysis
 - f. Review detailed preinstallation schedule of events
 - g. Define program blocking symbols
 - h. Design memory organization
 - i. Design I/O records and forms
 - j. Prepare program timing sheets
 - k. Begin coding
 - l. Design runs controls--halt system
 - m. Define EAM control procedure and log
 - n. Organize run and file numbering
 - o. Organize tape file history
 - p. Define assembly procedures
6. Test session preparation and procedures
 - a. Establish test procedures
 - b. Define EAM desk checking--board wiring
 - c. Prepare overlay (patch) sheets and cards
 - d. Test run book design
 - e. Test data forms
 - f. Maintain utility program decks
7. Establishment of conversion procedures
 - a. Establish installation time schedule
 - b. Analyze conversion planning chart
 - c. Finalize equipment and personnel requirements
 - d. Establish procedures and controls for:
 - (1) Auxiliary operation
 - (2) Library operations
 - (3) Console operations
 - (4) Machine room operations
8. Securing firm commitment on system delivery date

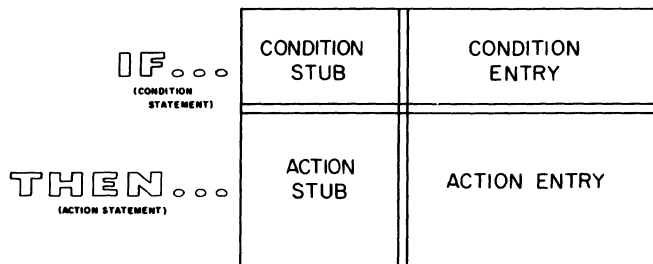
9. Machine room layout and cable order approval
10. Selection and training of operating personnel
 - a. The following personnel must be selected:
 1. Console operators
 2. Tape handlers
 3. Equipment operators and auxiliary operators
 4. Librarian
 5. Others
 - b. Provide training for all operating personnel
11. Conversion completed and application programs tested and ready for volume or parallel pilot runs
 - a. Schedule conversion operations
 - b. Place physical labels on tapes
 - c. Complete internal tape labeling
 - d. Perform EAM/EDP board wiring
 - e. Finalize sorts, merges and spools (tests)
 - f. Prepare final run book
 - g. Confirm system compatibility
 - h. Integrate parallel operations

DECISION TABLES

The basic objective of decision tables is to arrange and present a data processing system's application logic by using tabular presentation techniques so that its meaning can be readily grasped. Decision tables can be used effectively for systems analysis, procedure design and documentation.

In a decision table, the "if" conditions cause some action or actions to be taken -- for example, if an employee worked overtime, then compute his overtime pay. The format of a decision table separates conditions from actions. No action may appear in the condition area, no conditions may be indicated in the action area. No values may be changed by conditions. A condition must be answerable by either a yes or no. Actions are to be written in the order in which they are executed.

The four basic elements which constitute the structure of a decision table are shown in a schematic diagram below:



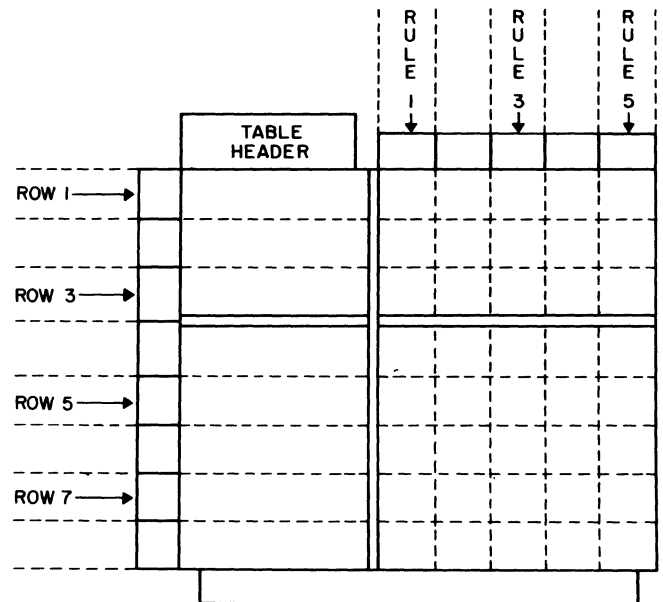
The double horizontal and vertical lines separate the four basic elements. Above the horizontal double line is the condition area; below, the action area. To the left of the vertical double line is the stub; to the right, the entry area.

The relative position of each element should remain consistent -- that is, the stubs are always to the left of the entries, and the conditions are always placed above the actions.

A rule is one vertical column (or group of associated columns numbered as a single rule) of the entry portion of the table and is read downwards in conjunction with the stub.

For a rule where multiple conditions exist, all conditions must be satisfied before the actions are executed.

There are three types of tables: limited entry, extended entry and mixed entry. In limited entry form, the entire condition or action is written in the stub, and the entry is limited to showing, for each case, whether the particular condition is true, false, or not pertinent (Y, N or blank), and whether a particular action should be performed (X or blank).



In contrast, an extended entry table expresses a part of the condition or action in the entry side of the table.

In general, limited entries compress a table horizontally, while extended entries compress a table vertically. The action statement shown below, for example, is compressed vertically by combining what could be three "go to" action statements into one extended entry row:

	Rule 1	Rule 2	Rule 3	Rule 4
GO TO	PREPSHIPREL	ORDER REJECT	ORDER REJECT	BACKORDER

A compromise using rows of both types -- that is, a mixed entry table -- is often useful. Limited and extended entry form may be freely mixed within a table, but a single condition or action row must be in just one format.

Aside from the terms used to describe tables according to the type of entries made, there are two other table types classified by the control they have on the sequence of execution or flow from table to table. Just as actions within a table are executed in the order in which they are stated, there must be similar control over the order in which tables are executed.

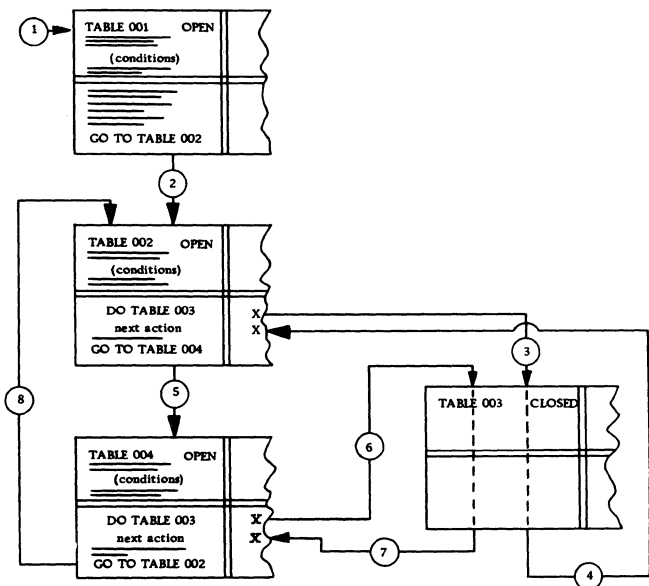
Within a set of tables, it may be desired for one table to permanently pass control to another table. This is accomplished by using appropriate terminology such as "go to". On the other hand, it is often desirable within one table to execute another table and then return to the original table so that

remaining actions can be executed. A notation such as "do" can be used to accomplish this. For example, in a table it may be desirable to call other tables to perform certain functions such as determine net price, compute shipping charges, and prepare invoice. Each of these tables can be called and executed by actions in this calling table -- for example, "do determine net price". The tables to be executed by actions in another table with an implied return to the calling table are designated "closed" tables and may be compared with closed subroutines. Tables which are entered by a "go to" action are designated "open" tables, and no return is implied.

The figure below shows within the numbered circles the sequence of execution of four tables.

Each of the open tables states where to go next. The closed table is referenced with a "do" command in tables 002 and 004. When the closed table is referenced by table 002, it returns control to the next action row of table 002; when it is referenced by table 004, it returns control to the next action of table 004.

Not only may more than one table reference a closed table, but frequently more than one rule within a given table references the same closed table.



Closed tables make it unnecessary to repeat conditions and actions common to more than one table or more than one rule within a table by abstracting the conditions and actions to form a closed table. Further, they permit re-entry to the table which referenced them without starting at the top of the table -- the normal entry point.

Each decision table must indicate where to go next -- to another table or to the table itself.

To illustrate a mixed entry decision table, a typical merging operation is documented. Three files (designated A, B and C) are arranged in numerically ascending sequence. The three files are to be combined in a single sequence with all records bearing the same number from any of the files placed together. This table is not complete, since the end-of-file conditions would have to be considered.

	1	2	3
1. A LE B	Y	N	
2. B LE C		Y	N
3. A LE C	Y		N
4. WRITE	RECORD A	RECORD B	RECORD C
5. READ	RECORD A	RECORD B	RECORD C
6. GO TO TABLE 2	X	X	X

The vocabulary and grammar of the language used within the decision table are left to the discretion of the user. It is his responsibility to insure that the language used in describing the system will be understandable to those who use the documentation.

A decision table form (X28-1630) is available for those including decision tables in their documentation. The basic concepts of decision tables and a minimum set of conventions for their use in systems analysis, procedure design, and documentation are described in the general information manual, "Decision Tables - A Systems Analysis and Documentation Technique" (F20-8102).

The use of decision tables to complement or support flowcharts is recommended when complicated logic or rules are encountered.

RUN DOCUMENTATION CHECK LIST

Decisions in systems and program design are supported by weeks or months of study and analysis. Adequate documentation protects this investment, provides an authoritative reference, and is useful in the design of related systems and programs. Explicit and prompt documentation eliminates any dependency on an individual, helps to avoid misunderstandings, serves as a training aid, and prevents duplication of research and programming effort.

Documentation includes the description of the entire system, the functions of each program, and the details of processing each type of record. Only brief narrative is required if flowcharts, decision tables, record layouts and lists of functions and alternatives are properly employed. To measure the adequacy of documentation, evaluate the degree to which it serves its purpose before, during and after the installation of the system to managers, systems engineers, programmers and operators. Extensive personal communication should not be required.

The principal documentation of a finished program is usually called the run book.

The sequence of documentation material should be functional to facilitate usage. The following check list may serve as a guide to constructing an effective run book which will satisfy documentation requirements:

1. Program name and identification
2. Table of contents
3. System flowchart, with this run highlighted
4. Machine configuration
5. Narrative
 - a. Purpose of run
 - b. Frequency
 - c. Description of run
 - d. Decision tables
 - e. Formulas
 - f. Unusual programming techniques
 - g. Procedure for processing errors
 - h. Control and audit provisions
6. Running time
 - a. Formula
 - b. Volumes
7. Inquiry response time
8. Input documentation
 - a. Source of each input
 - b. Disposition and retention of input
 - c. Volume
 - d. External card format or record layout
 - e. Example of forms; form number

- f. Control panel diagram
- g. Controls established or verified on inputs
- h. Internal record image
- i. Name and description of each field
9. Output documentation
 - a. Internal record image
 - b. Name and description of each field
 - c. Card form, record layout, or printing format
 - d. Form number and volume (quantity)
 - e. Example of form
 - f. Carriage control tape
 - g. Control panel diagram
 - h. Disposition of each output
 - i. Controls established or verified on outputs
10. Storage file description
 - a. Type (tape, disk)
 - b. Type of record (variable, fixed)
 - c. Blocking factor
 - d. Data representation (binary, BCD)
 - e. Storage requirements (I/O and work)
 - f. Source, disposition and retention cycle
 - g. Identification methods and labels
 - h. Controls
 - i. General functions
 - j. Disk techniques
 - k. Record layout
 - l. Name and description of each field
 - m. Record sequence
 - n. Volume
11. Program flowchart labeled with corresponding instruction address
 - a. Main-line logic flowchart
 - b. Detailed flowcharts
12. Storage map indicating location of:
 - a. Input, output and work areas
 - b. Fixed and variable constants
 - c. Tables
 - d. Running programs
 - e. Program overlays
 - f. Key locations
13. Index register listing
 - a. Label
 - b. Use
 - c. Content
14. Switch listing
 - a. Label
 - b. Location
 - c. Purpose
 - d. Settings
15. History log
 - a. Correspondence

- b. Agreements
- c. Program progress reports
- d. Dates of reassembly
- e. Record of changes
- 16. Complete operating instructions
 - a. External file labeling procedure
 - b. Description of control cards
 - c. Other programs required
 - d. File mounting or setup, file source and identification
 - e. Carriage control tape
 - f. Console switch settings
 - g. Console procedure:
 - (1) Load, program identification, halt
 - (2) Halt listing and prescribed action
 - (3) Standard restart procedure
 - (4) Exception restart procedure
 - (5) Error handling
 - (6) End of job -- external tape labels, log entries, control checks, disposition of input and output
 - h. Emergency phone numbers
- 17. Last assembled program listing
- 18. Last storage print
- 19. Last tape and disk prints
- 20. Last test data listing

BANKING

INDUSTRY DESCRIPTION

Commercial Banks

A commercial bank provides full financial services by accepting deposits, lending surplus balances to both individuals and businesses, and acting as a financial counselor to its customers. It is chartered either by the Federal Government or by the state in which it plans to operate. There are two types of commercial bank:

1. Unit bank, operating from a single building and not able to expand by adding locations. All of its business must be transacted from one central office.

2. Central bank with branches, consisting of a central office and one or more branches. Over the past two decades there has been a 300% increase in the number of branch banks. Twelve of the 50 states prohibit this kind of banking by statute. There are 13,500 commercial banks employing over 600,000 people; the largest 53 organizations (in terms of total deposits) do almost 40% of all the commercial banking business in the United States. The following are some of the overall statistics for the area:

Loans and investments: \$184 billion

Cash assets: \$48 billion

Total deposits: \$215 billion

Largest: Bank of America (assets: \$12 billion)

Bank management frequently sits on the boards of directors of other companies. Thus, the successful installation of IBM equipment within a bank sets an influential example to industry in general. It is increasingly more apparent that commercial banking is becoming an important seller of data processing services. This extension of service will further enhance the industry's data processing potential.

Savings and Loan Associations

Savings and loan associations were designed to encourage savings and permit members to acquire homes on easy terms. They are vigorous competitors for savings funds and generally pay a higher dividend rate than the banking establishments within the community.

Savings and loan associations may be chartered either federally or by the state in which they operate. There are 6,400 such organizations in the United States:

Federal Charter 1,800

State Charter 4,600

Federal savings and loan associations were established through the Homeowners Loan Act of 1933. These institutions are supervised by the Federal Home Loan Bank Board. In operation they are similar to state-chartered institutions, lending to home owners on mortgage security and encouraging long-term savings. Federal supervision involves an annual examination of accounts and loan security in addition to the necessary compliance with the charter.

Normally these institutions keep less than 5% of their total assets in cash and about 7% in U. S. Government bonds. The remaining amount is invested in mortgages.

APPLICATIONS

The applications that are performed uniquely in either commercial banks (CB) or savings and loan associations (S&L) are appropriately marked:

Demand Deposit Accounting (CB)

Customer checking accounts are maintained in a daily bookkeeping system. The operation includes receiving deposits and withdrawals, maintaining tight audit controls, preparing internal bank reports such as journals and trial balances, and preparing customer statements.

The American Bankers Association development of MICR (Magnetic Ink Character Recognition) in 1959 has resulted in a special line of IBM equipment to automate the encoding and processing of checks.

Transit Accounting (CB)

A given bank processes checks drawn on other banks as well as its own. These checks are also carefully balanced and controlled and forwarded for collection to the originating bank either directly

or through the Federal Reserve System. MICR processing is also used for this purpose to reduce repetitive manual processing of checks.

Savings Accounting (Time Deposit Accounting)

In this important customer service application, customer savings accounts are maintained, including the processing of deposits and withdrawals, passbook posting, and preparation of internal journals and trial balances. The announcement of special terminal units, such as the IBM 1062 Teller Terminal, has generated interest in online or real-time processing of this service.

Club Accounting

Christmas and vacation club accounts provide for the deposit of fixed amounts at regular intervals until predetermined totals are reached. Final checks are written and accounts are closed.

School Savings

Accounts are updated and maintained in the same manner as in normal savings accounting. The difference is in the physical manner in which the deposits are made. Deposits are usually sent in batches from the schools to the bank where updating is done en masse rather than individually.

Installment Loan Accounting

Loans are granted to customers for the purchase of automobiles, appliances, or for other personal reasons. The preparation and processing of payment coupons is involved, as well as posting customer accounts, identifying and controlling delinquencies and maintaining historical credit records.

Mortgage Loan Accounting

The granting and processing of home loans and other real estate loans secured by mortgages on the assets are involved in mortgage loan accounting. Interest, principal and escrow payments are calculated on the basis of the type of loan granted. Escrow balances (tax, insurance, special) are controlled and updated. Client information and history reports are prepared for both bank and government reporting purposes.

Commercial Loan Accounting (CB)

Loans are granted to businesses or corporations and are usually secured by collateral at a fixed interest rate for a specified period of time. Interest is calculated and billed and controls are maintained over expiration dates and loan status; a variety of principal payment methods are involved.

Personal Trust Accounting (CB)

Personal trust accounting deals with the management of customer portfolios, primarily of stocks and bonds, but often including many other types of assets. There are many legal and tax implications involved in the overall control of the various types of trust accounts. Management fees vary depending on the size and complexity of the trust. An inventory of customer account records is maintained as well as an inventory of the total holdings of each specific stock or bond. Transactions are processed daily to record purchases, sales, dividends received and payment of expenses. Periodically, a statement containing all transactions and current account status is furnished each customer.

Corporate Trust Accounting (CB)

This application involves the control and processing associated with issuing and transferring corporation stockholder records as a service to bank corporate customers. Records are maintained of owner's name and location, dividend payments are distributed, and proxy notices are processed.

SPECIAL CONSIDERATIONS

System selection must consider the number of accounts in each application area as well as average and peak transaction activity. Account numbers are often constructed with self-checking digits built into the code structure. Generally, it is required that all of the day's transactions be processed and reconciled prior to opening for the next day's business. This is particularly true of the demand deposit and transit operations, which usually entail evening processing. Account information must be readily available to answer customer inquiries. There are elaborate government regulations regarding transaction and record formats, reporting frequency, and investment position.

Section	Area	Page
04	01	03

IBM PUBLICATIONS

IBM 1440 Data Processing System and 1412 Magnetic Character Reader for Demand Deposit Accounting, general information manual (E20-0262).

Demand Deposit Accounting with Unit Record Equipment, general information manual (E20-0105).

The Proof Machine (320-4169).

Automated Deposit Proof and Transit, general information manual (E20-0245).

IBM 1440 Data Processing System for Mortgage Loan and Savings Accounting, general information manual (E20-0268).

IBM On-Line Teller Systems for Savings Accounting, general information manual (E20-8019).

Instalment Loan Accounting at Bank of Delaware with the IBM 1401 Tape System, general information manual (E20-0294).

Personal and Corporate Trust Accounting at the First National Bank of Birmingham, general information manual (E20-0109).

STANDARD TEXTS AND PERIODICALS

American Banker, 32 Stone Street, New York 4, N. Y., \$45/year. A daily newspaper, considered the best in the field and containing all the day-to-day happenings in the industry, including frequent coverage of data processing topics and events.

Banking, Journal of the American Banking Association, 12 East 36th Street, New York 16, N. Y., \$5/year. Written primarily for senior banking officials and financial management, and containing information on investments, banking trends, news and operations.

Auditgram, published by NABAC, the Association for Bank Audit, Control and Operation, 205 Touhy Avenue, Park Ridge, Illinois, \$6/year. Pitched mainly toward banking operations that may be considered as EDP applications. It also deals with new products in the field.

Prescott, Paul B., Financing American Enterprise, Harper Bros., New York, 1963. Presents an excellent history of the development of the banking industry, names of leaders and pioneers, and their contributions to the growth of the industry. Easy and enjoyable reading.

Aldorn, R. S., and A. B. Purdy, Automation in Banking, Rutgers University Press, New Brunswick, N. J., 1963. One of the few books on this subject. A compilation of four theses written by bankers and submitted to the Rutgers Graduate School, known as the Stonier Graduate School of Banking. Subjects are magnetic ink character recognition (MICR), the different forms of automation (partial or full), and the determination of automation feasibility.

Munn and Garcia, Encyclopedia of Banking and Finance, Bankers Publishing Co., 89 Beach Street, Boston, Mass., 1962, \$25.00. This prime reference of the industry contains detailed descriptions of banking terms and functions in addition to a thorough bibliography of texts.



BROKERAGE

INDUSTRY DESCRIPTION

The brokerage industry is primarily a service industry providing both the agency and facility for buying and selling securities and commodities. This function is performed through a complex of stock exchanges, brokers, banks, and other dealers.

Stock brokerage firms charge a fee for their services to the public for buying and selling securities through the various exchanges. They range in size from several men in one location to thousands of employees in many cities. There are 650 brokerage firms with membership in the New York Stock Exchange and with some 3,000 branch offices here and abroad.

The principal operations of brokerage firms are:

1. Execution and acknowledgment of customer buy and sell trading orders.
2. Maintenance of records of customer and firm accounts with both money and security balance.
3. Receipt and delivery of securities and/or funds in connection with trades.
4. Collection of the income due from securities held and the carrying out of transfers of stock registration.
5. Rendering of investment research and advisory service to customers. Stocks are usually traded in units of 100 shares, which is referred to as "round lot." An "odd lot" is less than 100 shares and is handled in a special manner by odd lot dealers.

APPLICATIONS

Purchase and Sales Records

Purchase and sales records are needed for the control and billing of customer transactions. Sales and purchases are recorded and statements, billing information and checks sent to clients.

Security Ledger

Control of the inventory of stock certificates held by the brokerage firm is the prime function of the security ledger. The file is maintained by security (e.g., name of stock), showing the disposition of each block of stock as well as the firm's position (ownership) in each security. The term "stock record" is often used to describe the file.

Customer and General Ledger

Each customer has an overall account money balance as well as a record of the securities currently held. An account may be updated by deposit of cash, withdrawal of cash, purchase of security, sale of security, federal borrowing tax, dividend or interest received, deposit of security by customer, or withdrawal of security by customer.

Margin Accounting

Customer accounts in which stocks are not fully paid for are called margin accounts. These accounts are continually checked for both holdings and commitments against current federal margin requirements.

Commission Analysis

Since commission from the sale of securities is often a brokerage firm's principal source of revenue, commission analysis is the firm's method of determining its sources of income.

Dividend Accounting

Dividend accounting is the allocation to the stockholders of corporate dividends paid to the brokerage firm on stock held for customers.

Purchase and Sales Clearance Accounting and Control

Each day purchases are matched against sales for all stocks traded. This settlement controls the transfer of stock to other brokerage houses and odd lot dealers.

SPECIAL CONSIDERATIONS

When a purchase or sale has been consummated, an advice or confirmation, which is the notice of the execution of the order, is prepared for the customer.

A record of every transaction which a broker makes for or with a customer is entered in the customer's account by the bookkeeping department of the firm.

Orders can be classified by size of order, type of transaction, price limits, time limits, stop orders, stop-limit orders, discretionary orders, immediate or cancel orders.

Section	Area	Page
04	02	02

Many brokerage firms maintain their own private wire networks connecting branch offices to the home office as well as to the stock exchanges.

IBM PUBLICATIONS

Series 50 Brokerage Accounting, general information manual (E20-0111).

Brokerage Accounting Guide (Z20-0315).

STANDARD PERIODICALS AND TEXTS

How the Marketplace Works

Leffler, G. L., The Stock Market, 2nd edition, Ronald Press, New York, 1957.

Shultz, B. E., The Securities Market and How It Works, Harper & Bros., New York, 1946.

Odd Lot Manual, Carlisle & Jacquelin, 2 Broadway, New York, privately printed, 1961, 85 pages. A good presentation of the activities on the floor of the stock exchange and behind the scenes with pictures and diagrams.

Understanding the New York Stock Exchange, New York Stock Exchange, 1962. Describes the different kinds of stocks and stock transactions.

Understanding the Modern Securities Market, Association of Stock Exchange Firms, 120 Broadway, New York 5, N. Y., 1960.

Commodities, Bache & Company, privately printed, 1958.

Analysis and Evaluation of Securities

Fisher, P. A., Common Stocks and Uncommon Profits, Harper & Bros., New York, 1958.

Graham, B., The Intelligent Investor, Harper & Bros., New York, 1959.

Graham, B., and Dodd, D. L., Securities Analysis, fourth edition, McGraw-Hill, New York, 1962. One of the foremost textbooks on how securities are evaluated by firms in making investment recommendations to customers.

Loeb, G. M., Battle for Investment Survival, Simon & Schuster, New York, 1957.

Investment Banking

Badger, R. E., and H. G. Guthmann, Investment Principles and Practices, fourth edition, Prentice-Hall, New York, 1951.

Fundamentals of Investment Banking, Investment Bankers Association, Prentice-Hall, New York, 1949.

STATE AND LOCAL GOVERNMENT

INDUSTRY DESCRIPTION

Some governmental departments exist to perform services (police, highway, welfare, unemployment), some to produce revenue (taxation, motor vehicles, ABC boards), and the remainder to exert general control over state functions (purchasing, budgeting, administration).

By organizational type, state and local governments are divided into four categories:

1. Large states with high population density and high revenue expenditure.
2. Medium and small states, thinly settled and with relatively low expenditure per capita.
3. Large cities and counties (about 43), whose expenditure per capita often exceeds that of medium and small states.
4. Small cities and counties, usually thinly settled and with a relatively small central government.

There are 50 state governments, approximately 3,000 counties, and over 17,000 cities and towns in the nation. All 50 states, as well as approximately 1000 counties, cities and towns, are considered IBM prospects. Each state is similar to the Federal Government in that there is an elected chief executive, a legislative branch and a judicial branch. The typical city or county government is operated along lines resembling those of the state; it has a mayor, a city council, possibly a city manager, and a city court system. Counties often are directed by boards of supervisors and seldom have the equivalent of a mayor. The finance committee comprises both the city council and the county board of supervisors and is often the key decision-making group regarding the acquisition of data processing equipment.

APPLICATIONS

Appropriation accounting. -- Amounts appropriated by the legislature are carried in master records along with identifying account codes. Any changes to original amounts, additions or deletions, are reflected by changes in the master record. At the end of the fiscal year, actual expenditures and receipts are balanced against original appropriations.

Revenue accounting. -- The periodic yield of taxes, fees, rents and other forms of government income is controlled by a system of revenue account codes. Many subsidiary applications, such as property tax

billing and licensing procedures, furnish input data to revenue accounts. All receipts are reflected as changes to revenue balances.

Employment security wage benefit. -- Employer contributions control and benefit accounting are only two of many important applications in the Employment Security Agency area:

1. Contributions for worker unemployment insurance are made quarterly to the state by employers. A control file of employer records is maintained.
2. Histories regarding the status of unemployed individuals are kept by the state. Benefit payments are recorded and statistics compiled for the Federal Government.

Driver licensing and control. -- Drivers' applications and licenses are printed before expiration date. Histories of drivers' violations are maintained and licenses suspended or revoked on the basis of offense records.

Vehicle registration. -- Records are kept on all vehicles registered within a given state. Control is maintained over transfer of ownership, new vehicle purchase and, in general, any situation altering owner identity. Registration cards are printed in cycles determined by the state.

Highway engineering. -- Since the problems of highway design and construction are similar from state to state, two major shared program efforts have evolved:

1. The Bureau of Public Roads has a staff writing engineering and administrative programs for the general use of all states. There are currently about 500 programs available to state highway departments through this agency.
2. HEEP is a share-type arrangement where the state highway departments of all states using IBM equipment make their programs available. Presently, there are 100 programs available through HEEP.

The general areas of present applications are:

1. Engineering calculations
 - a. Road design
 - b. Bridge design
 - c. Cut and fill
 - d. Culvert design
2. Payroll
3. Accounting
4. Inventory control
5. Traffic analysis

Additional application areas within state and local government include:

1. Public Health Administration
2. Public Safety and Defense
3. Public Works Administration

IBM PUBLICATIONS

The IBM 1620 for Engineering and Civil Engineering Applications, promotional brochure (520-1160).

The earthwork design program and the survey analysis program available for the 1620 make this computer a potential problem-solver in many government areas.

Series 50 for Local Government, promotional brochure (520-1033). The IBM Series 50 may be used for payroll, tax accounting, budgetary control and utility billing in a small local government. Compensation for the absence of large volumes is made by designing a system around several small-scale applications.

Voter Registration Records, promotional brochure (520-1034). IBM equipment can be used to maintain accurate records of voters and prepare all necessary registration reports by election districts.

Ohio Department of Highway Safety, application brief (K20-0917). The operation of the Statistical Department of the Ohio Department of Highway Safety furnishes a good example of the many potential statistical areas awaiting development.

A Unique, Computable Name Code for Alphabetic Account Numbering, general information manual (F20-8052).

Management by Exception System for Drivers Licensing and Control, general information manual (E20-8051).

Norfolk County Registry of Deeds, application brief (K20-0997).

St. Louis Metropolitan Police Department, general information manual (E20-0089).

1440 for Welfare Payments, general information manual (E20-0267).

1440 for Real Property Tax Accounting, general information manual (E20-0261).

Appropriation Accounting and Budgetary Control, general information manual (E20-8048).

STANDARD PERIODICALS AND TEXTS

The Book of the States. Published biennially by the Council of State Governments, 1313 East 60th Street, Chicago 37, Illinois. Information on the structures, working methods, financing and functional activities of state governments. Deals with legislative, executive, and judicial branches, intergovernmental relations, and major areas of public services. Emphasis is on the previous two years' developments -- comprehensive information on the work of state governments, with current directories of officeholders, both elected and appointed.

Municipal Yearbook. Published annually by the International City Managers Assn., 1313 East 6th St., Chicago 37, Ill. Information on the current problems of cities, with facts and statistics on individual city activities, and analyses of trends by population groups. Included also are salaries of chief municipal officials, directories of officials, city financial data, fire and police departments, forms of city government, taxpayer retirement systems, public relations programs, special planning surveys, and fire department training programs. Emphasis is placed on developments of the past year.

Statistical Abstract of the United States. Published annually by the U. S. Department of Commerce, available from Superintendent of Documents, U. S. Government Printing Office, Washington 25, D. C. This is an enormous source of information, from which many large reports are extracted. Included are statistics pertaining to population, births, deaths, number of families, size of income, education of citizens, percentage of population in each type of industry, number of dwelling units, number of manufacturers, wholesalers, retailers, data on banks, hospitals, government revenue and expenditures, and climate.

State Government News. Published monthly by the Council of State Governments, Chicago, Illinois.

From the State Capitols Newsletter. Published biweekly by Bethune, Jones Co. of Asbury Park, New Jersey.

Capitol, Courthouse and City Hall. Textbook published by Houghton Mifflin Company.

MEDICAL SERVICES

INDUSTRY DESCRIPTION

Expenditures in the biomedical field represent a large and increasing share of the national economy. Total health and medical care expenditures by American families, businesses and government currently amount to an annual rate of \$31 billion and are rising by 7% per year. By 1972, these expenditures are expected to represent 7.3% of the gross national product. Approximately one-third of the medical services industry expenditures are of an information-handling nature.

Medical care institutions are the most important group in this field, accounting for half of the biomedical expenditures for information handling. Included in this area are:

1. Short-term hospitals, characterized by a length of stay of generally less than one month. They are found in most cities and towns in the U. S.
2. Long-term hospitals, usually governmental hospitals for the treatment of the mentally or chronically ill.
3. Nursing homes, long-term residences located in most cities and towns, usually small in size and caring for the aged or physically handicapped.

The next group in importance are the medical practitioners (doctors, dentists, U.S. Public Health service center staffs, etc.). This group currently spends \$2.4 billion for its information-handling requirements. The figure is expected to increase by 50 percent in ten years.

Research is the fastest growing biomedical segment. University medical schools are in the process of expanding their research efforts, particularly in study areas related to data gathering and data reduction: physiological monitoring, patient screening, statistical studies, etc.

Other medical and biomedical groups include:

1. Professional groups
 - a. Group clinics
 - b. Visiting nurses associations
2. Government-aviation agencies
 - a. U. S. Public Health service (NIH)
 - b. Veterans Administration
 - c. NASA
3. State and local government
 - a. Hospitals
 - b. Schools

Of the 5,878 short-term hospitals, the 2,000 with 100 beds or more are considered prime prospects for electronic data processing equipment. Long-term hospitals with a 500-bed (or greater) capacity are also possible users. All 150 medical and dental schools are in the same prospect category.

APPLICATIONS

Accounting

Patient billing: Preparation of bills is based on time in hospital and services and supplies used, each charge being itemized. Medical compensation records are also shown on the bill.

Insurance proration: A master file is maintained containing charge and code for each plan in force. Charges are divided between insurance company and patient according to the limitations of the plan.

Inpatient accounts receivable: Billing establishes the accounts receivable file. Collections are passed against indebtedness, trial balances are run and historical and statistical records are maintained.

Outpatient accounting: Outpatient accounting is similar to inpatient accounting but is usually handled by a separate department within the hospital.

Other accounting functions include payroll, accounts payable, general ledger and budgetary control.

Functional Control

Doctors' orders: Using a TELE-PROCESSING system, doctor's orders are entered, automatically processed and assigned to both the pharmacy and the station responsible for administering them. Control is maintained and the necessary timing messages are sent.

Nurses scheduling: Through an analysis of historical records and the current patient load and status, nursing is dispatched on a need basis.

Pharmacy control: A central inventory of drug supplies is maintained and controlled. Controls as to the disposition of drugs are set up and tested. Statistical and historical records are maintained.

Research

Data analysis: Certain analytical procedures in medicine and biology require enormous manipulation of data in order to carry out complex mathematical analysis.

Simulation of biological system: Factors describing and influencing a chemical reaction can be given to a computer and the reaction simulated. The effects of varying the factors or introducing new factors can be studied much more rapidly and with less risk than they could be in the true situation.

Diagnostic procedures: Diagnostic procedures involve comparing a patient's symptoms with historical data to produce a list of possible ills and suggestions for further examination. Future possibilities include suggested treatment.

Mass screening: Conversion of various medical tests (EEGs, EKGs, etc.) to digital form for computer comparison with normal parameters to indicate exceptions, thus permitting mass detection of cases to be closely watched.

Machine evaluation of psychological test and questionnaires: The computer is used to correlate questionnaire answers with psychological measurement factors to indicate possible mental disorders.

Information retrieval: Technical data is obtained rapidly from medical records or whole documents from a library for technical review.

SPECIAL CONSIDERATIONS

In the installation of data processing systems in hospitals, cost justification is a prime consideration. A 300-bed hospital can justify a computer system for accounting and/or functional control. The hospital administrator is the main contact point for these areas.

Economic justification is not a primary consideration for the sale of equipment in the research area. Normally this area is controlled by a director in charge of research.

A prime consideration in a hospital installation is the quality of the data processing personnel. Generally this area demands heavier than normal support activity. Research institutions, because of their scientific interests, require systems engineering assistance in the area of mathematical techniques.

IBM PUBLICATIONS

Proceedings of the 1st IBM Medical Symposium 1959 and 2nd IBM Medical Symposium 1960 (320-1571).

Proceedings of the 3rd IBM Medical Symposium 1961 (320-1648).

Proceedings of the 4th IBM Medical Symposium 1962 (320-1752).

Hospital Accounting, general information manual, (E20-0007). Designed to acquaint IBM personnel and hospital executives with various proven methods of using IBM equipment. Covers patient billing, accounts receivable, payroll, personnel records, accounts payable, inventory control, financial control and statements and medical records.

The Emma L. Bixby Hospital Data Processing System, application brief (J20-0979). Describes the use of IBM Series 50 for billing, accounts receivable, accounts payable, inventory control and operating reports.

Data Processing at Pekin Memorial Hospital, application brief (K20-0114). Describes accounts receivable, medical records, accounts payable, and inventory control for the small 200-bed hospital.

The IBM 1001 Data Transmission System for Hospital Charge Reporting, general information manual (E20-0069).

The IBM RAMAC 305 for Hospitals, general information manual (E20-2030). Describes patient billing, accounts receivable, revenue and expense accounting, supply accounting, payroll, general ledger accounting and accounts payable.

The Laconia Clinic, Accounting for Receivables, application brief (K20-1373). Describes the use of IBM equipment in small clinic for accounts receivable and data collection applications.

In-Patient Billing - 407 Approach to splitting charges between patient and insurance companies
Source: WRO

Patient Billing and Accounts Receivable - Source:
WRO

Admitting, Charging and Billing Patients - Source:
WRO

Budgetary Account and Control - Source: WRO

Medical Record Statistics - Source: WRO

Out-Patient Charge Procedure - Source: WRO

Unit Record PAL for In-Patient Billing - Source:
WRO

RAMAC 305 for Hospital Accounting at Baptist Memorial Hospital, Memphis, application brief (K20-2060).

Section	Area	Page
04	04	03

1440 DPS for Patient and General Accounting in Hospitals, general information manual (E20-0266).

IBM Programmed Applications Library for Hospital Accounting (520-1508).

IBM Data Processing for Hospitals (520-1086).

IBM 609 Calculator for Aging Accounts Receivable (for a medical clinic), general information manual (E26-1515).

Rees-Stealy Medical Clinic Billing and Accounts Receivable, application brief (K20-1393).

Marymount Hospital - Data Processing, application brief (K20-1400).

Hospital In-Patient Accounting - HI PAU/R

Patient Billing and Accounts Receivable at Barnes Hospital, general information manual (E20-0005).

See also the current edition of the KWIC Index to IBM Marketing Publications catalog of programs.

STANDARD PERIODICALS AND TEXTS

Hospitals Magazine: Journal of the American Hospital Association. Published semimonthly by the American Hospital Association, 840 North Lake Shore Drive, Chicago 11, Illinois. General articles of interest concerning hospital administration, equipment, and safety.

The Journal of the American Medical Association: Published monthly by the American Medical Association, Chicago, Illinois.

1. Directory of approved residences for each type of medical science -- surgery, psychiatry, pediatrics, internal medicine, etc.

2. Essentials of approved internship -- locations.

3. Consolidated list of hospitals, specifying those with approved graduate training.

4. Annual Report.

5. Editorials.

Hospital Progress Magazine: Official journal of the Catholic Hospital Association. Published monthly by the Catholic Hospital Association of the U. S. and Canada, 1438 South Grand Boulevard, St. Louis 4, Mo. Hospital news articles, book reviews, new equipment, hospital floor plans, dietary planning.

Medical Electronic News: Published quarterly by the Instruments Publishing Co., Pittsburgh, Pa.

American Journal of Medical Electronics: Published quarterly by the Yorke Publication Co., New York 17, N. Y.

The Guide Issue of the Journal of the American Hospital Association: Published annually.



CHAIN, WHOLESALE AND RETAIL

INDUSTRY DESCRIPTION

Three important parts of the distribution industry are the chain, wholesale and retail businesses. In many cases companies perform both wholesaling and retailing functions as well as a primary manufacturing role.

There are approximately 2,000,000 retail establishments in the nation. Over 16,000 of these have annual sales of more than \$1 million. Total annual sales of retailers in the United States exceeds \$220 billion per year.

Of the 320,000 firms engaged in manufacturing, 7,200 have annual sales of \$1 million or more. Total sales in this area approximate \$370 billion annually.

Merchant wholesalers function as middle men between retailers and manufacturers. Today there are some 190,000 of these firms with close to 2,000,000 employees. Current sales for this group are in the area of \$150 billion annually. Manufacturers of motor vehicles, parts, metals, chemicals, and drugs frequently sell directly to retailers.

The retailer, subject to competition of mass merchandising with lowered profit margins and rapid expansion, often must modernize his operations at great cost.

Recent trends in the industry have been toward the super-store approach to merchandising. The items inventoried by any one company have increased greatly in number and variety. Since many more stocked units must be carried, the problems of inventory management and control of turnover become acute.

Generally, the distribution industries perform accounting applications that are routine and necessary for their business operation. Penetration of the industry has been through the controllers, with only limited acceptance of data processing equipment to date by the merchandisers.

APPLICATIONS - CHAIN AND WHOLESALE

Inventory Control at the Warehouse Level

Files are maintained showing the status and disposition of inventory items. Purchase records are periodically passed against the file and tested against critical points to determine reorder levels. Information about the disposition of inventory is

maintained for sales analysis purposes. Depending upon the size of the customer, a subfile or disk file approach is most normally used in the mechanization of this application.

Purchase Order Writing

Purchase orders are passed against inventory items to determine price, availability and, in some instances, location. Output may include picking lists, shipping tickets, invoices, sales analysis information, customer statements, and purchase orders.

General Accounting Functions

Included in this area are:

- Payroll
- Accounts receivable
- Accounts payable
- General ledger
- Retail store accounting
- Store operating statements
- Route accounting

APPLICATIONS - RETAIL

Merchandise Control

Central files are maintained on both the seasonal and staple items in the overall retail inventory. The input passed against the file is a sales transaction. The output is a stock status report showing movement of merchandise.

Accounts Receivable

Customer records are stored centrally and are updated with charged purchases and account payments. Output consists of customer statements and aged trial balances.

General Accounting Functions

Included are:

- Payroll
- Accounts payable
- Tally audit
- Office audit
- Floor audit
- Subsidiary ledger
- General ledger
- Commission listing

ADVANCED APPLICATIONS

Inventory Management

The IMPACT (Inventory Management Program and Control Techniques) system allows increased management knowledge of the products sold in the distribution industry. It involves extensive recording and control of inventory to a point at which effective management decisions can be made from available information.

Customer Management

The distribution industries are becoming aware of the need for improved information about the customers they serve. With improved customer management they learn why customers do business with them, thus acquire the tools to attract other customers. Work in this area is just now starting.

SPECIAL CONSIDERATIONS

Retailers and wholesalers are not accustomed to spending large sums for research and preparation for new systems.

The major role played by the controller has largely confined mechanization to traditional accounting applications.

The area with the greatest profit potential, merchandise control, will not show an immediate dollar saving.

Trained personnel are in short supply.

IBM PUBLICATIONS

Chain and Wholesale

Purchase Order Writing for the Chain and Wholesale Industry (E20-0060). The purchase order document is analyzed, the basic approach outlined, major variations described, a typical operation outlined, and significant factors summarized. An excellent basic presentation of this application. Similar techniques can be used by distributors of other merchandise.

Wholesale Food Distribution (K20-1043). Describes the use of unit record equipment in a simple method of warehouse accounting where complicated machine procedures and reports have been eliminated.

The IBM RAMAC 1401 for the Chain and Wholesale Grocery (E20-0223). The IBM card order plan forms the basis of the procedure but manually

punched cards may also be used. The cards, in order book sequence, are read into the 1401, which arranges the order in warehouse location sequence, determines availability, updates the inventory record, and prints the completed invoice in picking sequence.

Control of Inventory and Purchasing with the IBM RAMAC 305 (K20-2055). The significance of this approach to the control of inventory and purchasing is that it is comparable with any mechanized billing procedure: after the invoices are written, the billing cards become input to the 305 for updating inventory and controlling purchasing activity.

Data Processing for a Hardware Wholesaler (K20-1405). Inventory maintenance and control on an IBM RAMAC 305. A detailed record of each inventory item is maintained in disk storage and used daily in pricing and extending orders, updating those inventory items affected by daily transactions, signaling those items reaching a minimum balance or out-of-stock condition and supplying inventory statistics for purchasing.

IBM RAMAC 305 for the Distribution Industry (E20-2061). Contains a complete program for the chain wholesale distribution field for billing, inventory control and sales analysis. Also discusses methods of input preparation, file packing in relation to access time, plus other considerations.

Retail

IBM 1440 Data Processing System for Retail Fashion Inventory Control (E20-0263). The description covers the application from the preparation of input through the printing of each major report required. Management-by-exception techniques for control are discussed and illustrated.

IBM 1401 Tape System at D. H. Holmes Company, Ltd. (K20-0242). Depicts, in flowchart form, some of the initial applications performed on the 1401 tape system at the subject company. Sample reports are included.

IBM 1401 Tape System for Accounts Receivable and Merchandise Management (E20-0241). The accounts receivable section describes a highly mechanized approach to retail accounts receivable using punched card saleschecks, credits, remittance stubs, customer statements and history ledgers. The merchandise control section describes an approach to this function combining management by exception and detailed reporting.

Section	Area	Page
04	05	03

1401 Data Processing System with Tapes for Merchandise Control at Retail Chains (E20-0204).
Describes the 1401 tape system for merchandise control in a retail chain application.

1401 Data Processing System with Tapes for Accounts Receivable for Retailers (E20-0203).
Short description of 1401 components and procedures are included. Companion manual to E20-0204.

Retail Glossary (F20-8006-00-01).

Retail Department Store Analytical Guide (F20-8007-00-01).

STANDARD PERIODICALS AND TEXTS

Chain and Wholesale

Supermarket Merchandiser
Supermarket News
Progressive Grocer

Retail

Woman's Wear Daily
Department Store Economist
Stores Magazine
Chain Store Age
Variety Store Merchandiser

MAJOR ASSOCIATIONS

Chain and Wholesale

Each subdivision of the wholesale industry has its own association. The grocery industry, however, represents the largest market in terms of immediate potential. Its two major associations are:

Supermarket Institute
National American Wholesale Grocers Association

Retail

National Retail Merchants Association



SCHOOLS AND UNIVERSITIES

INDUSTRY DESCRIPTION

The field of education has a variety of institutions: colleges and universities are at the top level; junior colleges and post high schools, mostly technical and business, are next. Secondary schools constitute the third level, with elementary schools forming the lowest group. Administration, teaching, and research are the major areas of interest.

There are about 1,500 accredited colleges and universities -- institutions giving a bachelor's degree after four years. Approximately 700 give a master's degree and 200 a doctor's degree. Accredited junior colleges -- about 600 in number -- grant an associate degree and give credits which may be transferred to four-year institutions. One hundred and fifty 2-year technical or vocational schools, junior college in level, graduate technicians usually not intending to take additional education. Out of 23,000 junior and senior high schools, 5,000 have over 1,000 pupils each. Approximately 5,000 school districts are large enough to employ managers to handle their business matters.

Almost all colleges, secondary and elementary schools, are supported by public tax funds (city, county or state). Colleges also receive many private endowments. Private schools are supported by a combination of endowments and tuition. Therefore, all of these institutions are nonprofit endeavors. Most business schools are private, profit-making organizations. Practically all educational institutions must be licensed or accredited by county, city, state, regional, national or professional bodies. Those supported by public funds are controlled by government agencies.

SPECIAL CONSIDERATIONS

In areas where no centralized data processing unit exists, any secondary school with 1,000 students or more, has need for a record keeping installation. The formation of a data center allows schools to automate important phases of pupil accounting and business applications regardless of size. Technical schools and post-high schools of various types are potential users of equipment for the purpose of teaching data processing curricula.

At least 1,000 colleges or universities are a potential market for a number of applications. Size is not a barrier in this area. One college of 250 students has a computation center.

Every one of the 50 states is doing some kind of punched card statistical application at the state level. Several hundred district or city systems are using punched cards for administrative functions. Not counting test scoring, only a few dozen secondary schools have independent installations on the premises.

Educators usually are not interested in specific machine configurations. They are problem oriented and therefore question an approach on a "dollars and cents" basis. Since the schools are supported by public funds, cost is a problem: obtaining an appropriation to do data processing work is ordinarily time consuming even when justification for a new system is clear. The systems engineer who understands some of the problems confronting the modern educator, will have the best chance of completing a successful study.

APPLICATIONS

Student Record Handling

Student registration and class scheduling. -- Basic information concerning the student (name, address, age, etc.) is recorded at semester registration time, along with academic scheduling data (subjects, instructors, classrooms, and times). This information is used to plan master schedules and form class groups. It is also passed on for use in guidance.

Pupil attendance. -- A record is kept for each pupil, showing times tardy and times absent. This information is accumulated by individual for report card usage. It also forms the basis of statistics included in administrative and government reporting.

Grades reporting. -- Examination grades are submitted to a central location where they are calculated for averages and posted to report cards. Statistics based on individual grades are gathered, to be distributed along various control lines -- age, sex, handicap factors, school level and many other categories.

Financial Record Handling

Payroll. -- Control and verification of lost work time is a preliminary stage in education payroll processing. Vacation and sick leave are audited and posted to master leave records. Increments are granted in light of seniority and credits earned.

Purchasing. -- Purchase orders are prepared in response to inventory replacement needs or school requests. Order files are maintained for control of received items. Reports are necessary to reflect this important aspect of expenditures.

Budget. -- Appropriated amounts are recorded under an appropriation code structure. All changes are added to or deducted from the original scheduled amounts. At the end of the financial year, intended amounts are compared against actual expended sums for all accounts.

School supplies control.

Cafeteria accounting.

Research

Graduate students and school system staffs pursue research projects in almost all subject areas. Other projects frequently involve the gathering of data and the correlation of statistical results for the evaluation of many situations -- teaching effect on learning, relations among student abilities, grades, attendance, etc. The goal is an increased understanding of the learning process.

Teaching

Data processing equipment is increasingly used as a teaching device in varied ways. It is used for problem solution in the given subject area. Mathematics is frequently taught in relation to computer concepts. Such equipment is also used as the vehicle for the study of programming, machine repair and systems analysis.

Test Scoring

Grading has been done for years with IBM equipment. Emphasis on these analyses is now growing, since educators are trying to improve subject teaching through increased testing knowledge. Testing information is being used to establish the validity of specific questions.

Management Information for Decision-Making

Information handling is being increasingly mechanized. Administrators are striving to learn whether their policies are being carried out, or to determine areas where policy changes are needed.

ADVANCED APPLICATIONS

Flexible scheduling. -- By means of computer programming, schedules may someday be on three levels: the individual pupil, the small work group and the large work group. The individual's aptitude and interest in a given subject area determine how many time units he devotes to that subject. The school staff consists of master teachers who coordinate activities and assistants who carry out plans. (Still in experimental stage.)

Test evaluation. -- Objective type tests are graded by computer, and each individual question is evaluated in context with overall test performance. Means and norms are established by which the statistical value of particular questions is ascertained. Question banks may be established for subject areas so that more reliable examinations may be designed.

Programmed learning. -- The computer is programmed to grade student responses to questions and, when a wrong answer is given, guide him into further study until he replies correctly. Text is held in storage. The application will become economically feasible at that future time when many pupils may work with the same computer simultaneously.

IBM PUBLICATIONS

IBM Data Processing Machines at University of Oklahoma, general information manual (E20-0088). Registration and other applications at the university are described in such a fashion that the university officers are furnished with a working guide, and information is given to other universities and colleges.

Public Education Student Records, general information manual (E20-0056). Census records, attendance records, high school programming and registration, grade reporting and posting of permanent records, progress and program evaluation are discussed. Five important phases of public education are treated. A detailed explanation is given of the way in which IBM is helping school administrators in these areas.

Section	Area	Page
04	06	03

Nyack High School Student Accounting, general information manual (K20-1371). Nyack High School's IBM procedures have received a great deal of attention from secondary school administrators and guidance personnel, and are regarded as highly successful. The handling of course changes, attendance and mailings, report card printing, the keeping of permanent record cards, honor rolls, and updated class lists are among the topics discussed.

Data Processing at the Memphis School System with the 1401 and RAMAC 305, general information manual (E20-0288). The Memphis School System, with a student body of 106,000, has mechanized a considerable part of its business and student accounting areas. The 305 locates and updates multiple records pertaining to a given transaction. The tape 1401 is used where fast access to sequentially stored information is required.

IBM RAMAC 1410 for Student Scheduling in Colleges and Universities, application bulletin (H20-8050). The RAMAC 1410 with its large core storage capacity is used effectively for the automatic scheduling of students. Included are short descriptions of the components of the system as well as card and report forms.

IBM 1440 Application Programs for School Systems, application program bulletin (H20-0270). IBM application programs are available for school systems. Each run is described and the basic 1440 system configuration is specified.

STANDARD PERIODICALS AND TEXTS

The Nation's Schools, monthly, F. W. Dodge Corp. (a McGraw-Hill Co.), 1050 Merchandise Mart, Chicago 54, Illinois (\$4.00 year). Addresses

itself to better school administration, offers comprehensive coverage of the field of school management. A regular feature is a page called "Time and Dollar Savers".

The American School Board Journal, monthly, Bruce Publishing Co., 400 N. Broadway, Milwaukee 1, Wisconsin (\$4.50 year). An administrative guide for board members, superintendents, business managers, architects. The material published tends toward school system administration slightly more than toward school building management. The February 1963 issue has an interesting article entitled "Starting an EDP Program in a School System".

Educational Executives' Overview, monthly, Buttenheim Publishing Corporation, 470 Park Ave. South, New York 16, N. Y. (\$5.00 year). Uses a news approach. Its coverage of education tends to be all-inclusive.

Books

Conant, James B., The American High School Today, McGraw-Hill Book Company, New York, 1959. This is a quick and readable introduction to current concerns and the philosophy of American education. The book includes a chapter called "The Characteristics of American Education". It also discusses the comprehensive high school at the small, large, city and suburban levels.

Edmondson, J. B., J. Roemer, and F. L. Bacon, The Administration of the Modern Secondary School, Macmillan Company, New York, 1960. The roots of most systems concerns will be in the school classroom and school office. The book includes chapters on scheduling, finance, and records, all on the individual school building level.



MOTOR FREIGHT

INDUSTRY DESCRIPTION

There are approximately 17,500 regulated inter-city carriers in the United States with a gross annual revenue of about \$7.5 billion. Motor freight carriers account for 22% of the total intercity miles of freight movement. About 3,400 carriers gross over \$200,000 a year, with 1,000 grossing \$1 million or more per year.

With a total expense cost of 96¢ per revenue dollar, including a direct labor cost of 55¢/revenue dollar, the truckers have been forced to initiate a drive for more forceful cost controls. This has led the carriers to an increased use of data processing equipment and communications systems. This increasing use is right in line with the computer-inspired speedup of smaller, more frequent shipments on the customer's part.

APPLICATIONS

At present, the most popular applications are the determination of revenues and payroll. Terminal operating costs and revenue control are other promising areas.

Revenue Accounting -- Audit and Control of Accounts Receivable by Terminal

A central accounts receivable file is maintained by freight bill number within collecting terminal sequence. Periodically, late lists are set up in customer-number sequence for the use of the collecting terminal. Audit control, dunning notices and statistical summaries are run from this file.

Interline Control

Central control is maintained over the receivable-payable position of all connecting carriers. File information includes carrier name, interchange point, receivable amount, payable amount, a net total and a ratio figure. Two methods of payment are used:

1. Contra. -- The receivable-payable balance is calculated and only that difference changes hands.
2. Exchange. -- Settlement is made on a bill-for-bill basis.

Other applications in this area include centralized customer statement writing, revenue and tonnage statistics, and centralized accounts receivable.

Sales Analysis

Information including customer name, salesman, tonnage and revenue statistics by month for last year and this, percentage change and year-to-date totals is desired in analytical reports. Included within this area is a claims analysis to determine the routes and shipments which result in a high claims ratio as well as an analysis of comparing sales and customer trends.

Equipment Accounting

Included within this application area are traffic analysis, tax calculation and tax mile reporting, cost reporting, inventory, preventive maintenance reports, and schedules.

Other Applications

Additional application areas are:

- Payroll
- Accounts payable
- Inventory control
- General accounting
- Financial reports
- Performance reports
- Budgetary control

ADVANCED APPLICATIONS

A total system solution to both the accounting and operational problems is necessary if the motor freight industry is to receive maximum advantages from data processing. A file-oriented TELE-PROCESSING system is one possible route to sound centralization. Terminal locations and break bulk stations can be linked to the CPU via a communications network.

The functions performed centrally by the CPU could be those currently done manually at the terminal locations, such as dispatching, centralized rating, and freight billing. Other terminal work areas are:

1. Data gathering: payroll, terminal statistics, maintenance recording
2. Equipment inventory
3. Maintenance scheduling
4. Terminal control: operations analysis, decision information
5. Revenue accounting
6. Crew scheduling
7. Credit checking

In addition to the applications mentioned above, a TELE-PROCESSING system may also give the carrier centralized control of accounts receivable, a function now performed by collectors at terminal locations. Incorporation of this function would relieve the burden on the terminal and give the carrier an up-to-the-minute control over its monetary position. Billing can be done centrally and receivable controls set up to handle remittances.

The potential for TELE-PROCESSING systems, comprising random access files and terminals linked to a central processor, far exceeds that of presently installed equipment in this phase of the transportation industry.

SPECIAL CONSIDERATIONS

As shipping management becomes more scientific in its approach to inventory control, there is a consistent trend toward more frequent ordering of smaller commodities in order to minimize inventory carrying charges. These charges -- including interest on the money invested in inventory, taxes, and all facility and occupancy costs -- usually run between 18 and 24% per year of the cost of the goods. This figure, by itself, has been shocking enough to stimulate many shippers into a review of their overall systems.

A thorough study of the carrier must be made to determine which application areas can be profitably mechanized, as well as to point the way to the type of equipment that should be used.

IBM PUBLICATIONS

Fox & Ginn, Inc. - Motor Freight Accounting, application brief (K20-0998). The applications being performed by Fox & Ginn, Inc., Bangor, Maine, are done on a 526 Printing Summary Punch, an 82 Sorter and a 402 Alphabetical Accounting Machine. Each application is illustrated with card and report forms. A list of advantages experienced by this small motor freight carrier is included. Billing, accounts receivable, revenue accounting, payroll, sales analysis, maintenance records, and general ledger are done on IBM equipment.

Carolina Freight Carriers Corporation -- TELE-PROCESSING system for Billing, application brief (K20-8068). A large freight carrier estimates costly delays, improves internal operations and provides better customer service with the IBM 65-66 Data Transceiver. Billing is the main application.

Motor Freight Accounting, general information manual (E20-0044). Unit record equipment is used for the preparation of reports on motor freight accounting, statistics, payroll, maintenance, general ledger and operating costs. Reproductions of the document and card forms used in the indicated applications are supplied.

The IBM RAMAC 1401 System with Tapes for Motor Freight Revenue Accounting, general information manual (E20-6214). Some of the advantages accruing to the motor carrier using the RAMAC 1401 with 7330 Magnetic Tape Units are as follows:

1. Combined control of multiple accounting functions.
 2. Action-oriented reporting.
 3. Automatic references to prior records.
 4. Reduction of manual searching through the maintenance on tape of 30 days' accounting of payments received.
 5. Protection of records against removal from the file and subsequent loss or mishandling.
 6. Increased customer goodwill through automatic detection of duplicate payments.
- Flowcharts and report forms accompany the text.

IBM 1440 Data Processing Systems for Revenue Accounting in the Motor Freight Industry, general information manual (E20-0265). Areas covered include maintenance of statistical data, statement writing, cash applications, aged trial balance and interline payables. Disk pack storage is used throughout the procedures. Report forms and record layouts are included.

Transit Scheduling on the IBM 650 at Philadelphia Transportation Company, general information manual (E20-8103). Describes a new concept in handling paperwork at points of origin and destination terminals. The 1050 Data Communications System is utilized.

STANDARD PERIODICALS AND TEXTS

National Highway and Air Carriers and Routes, National Highway Carriers Directory, Inc., 925 West Jackson Blvd., Chicago 7, Illinois. Contains an alphabetic index of carriers, listing the companies' officials, number of units, location of terminals, type of service, etc. Other sections include carriers' maps and advertisements, general information, routing and air carriers' information. Published semiannually in the spring and fall.

Section	Area	Page
04	07	03

TRINCS - Blue Book of the Trucking Industry, published annually, TRINC Associates, Ltd., 240 Woodward Building, Washington 5, D.C. Contains selected information from the annual reports filed with the Interstate Commerce Commission by Class I and Class II motor carriers. The following information is provided on over 3,000 carriers:

1. Industry summaries and statistics
2. Management and control (ownership, stock outstanding, etc.)
3. Cost analysis tables (average haul, average load, revenue per ton, revenue per mile, etc.)
4. Basic discounts and statistics

Traffic World, Traffic Service Corporation, Washington Building, Washington 5, D.C. This weekly news magazine of transportation management highlights current news items on national and international transportation legislation and statistics, with sections devoted to the ICC, the Aeronautics Board, and the Maritime Commission.

Transport Topics, national newspaper of the motor freight carriers, published weekly by Transportation Topics, 1616 P Street N.W., Washington 6, D.C. Contains current news items for the motor freight industry.

Dun's Review and Modern Industry, Dun & Bradstreet, 99 Church St., New York 8, N. Y.



GAS AND ELECTRIC UTILITIES

INDUSTRY DESCRIPTION

Most gas and electric utility companies in the United States are investor-owned; however, publicly owned power-producing organizations, such as the TVA, have reached a significant number. The national postwar yearly growth rate of electric power consumption has averaged 8.6%, which represents a growth rate 2-1/2 times that of the Gross National Product. Net capital invested in utilities exceeds \$38 billion, 12% of the total capital invested in all of U. S. business.

Investor-owned utilities are noncompetitive with the exception of the gas vs. electric power competition. Such organizations are regulated through the Federal Power Commission and various state bodies. A given electric utility generates, transmits and distributes its own power. Frequently, however, in gas production these functions are divided among separate companies.

Of the top 50 utilities, 15 have assets in excess of \$1 billion. Total assets for these top 50 companies exceeds \$45 billion.

Both customer and general accounting applications are highly mechanized in the utility industry. Increasing in importance are the engineering and operating functions, in which the trend is toward digital computer control of power plants and economic dispatch operations.

APPLICATIONS

Customer Accounting

Customer accounting is a broad area that involves the collection and distribution of information concerning customer usage of utility power. Some of the specific applications follow:

Mark-sense meter reading. -- The advantages of direct creation of source documents have caused most utilities to use cards that are mark-sensed with new meter readings and sent without key-punching to the data processing system. The cards contain customer numbers, meter numbers and spaces to mark four or five-digit meter readings.

Customer billing and accounting. -- Mark-sense cards are converted to transaction records. They are validated, sorted and merged with the main file section for billing. The calculated bill is edited and both the bill and supporting bill registers are printed. Discrepancies and errors are set up in memo form for corrective action.

Cash posting. -- Many utility companies render customer bills in punched card form with stubs to be returned with payment. These payments are accepted at any branch of the company, the money deposited and the stubs with balance totals sent to a central processing point. The stub cards serve to update the customer accounts receivable files.

Service orders. -- Service orders processing consists of making necessary changes to the main file (customer name, mailing address, type of service, etc.). These changes are not necessarily dependent upon normal operation, but may be precipitated by a customer request or transaction. Alterations are batched and validated for periodic file updating.

General Accounting

Accounts payable. -- A central cash distribution and master account file is updated by invoices, cash disbursements and general fund canceled checks. Output includes voucher listings, a daily cash listing and the general fund bank reconciliation. In regard to materials and supplies, the central files contain a record of material balance. The file is updated by various orders and adjustments to the inventory. The resultant output is a variety of listings describing the overall control of the inventory.

Plant accounting. -- General accounting work orders, material work orders and units of property constitute the files involved in this application. Property requirements and requests for job analysis and costing act on the files to produce work order listings, plant closing reports and property listings.

Other accounting applications include:

- Payroll and personnel
- General ledger
- Responsibility accounting
- Shareholders records
- Property records

Engineering

Engineering encompasses the planning, design and testing of the physical characteristics and equipment associated with the operation of a public utility. Major subdivisions within this application area include:

- System planning
- Power plant design

Transmission design
Distribution design
Meter testing

Operations

Simulation, statistical analysis and process control are the standard computer applications used in this area of the public utility business. Included are:

System load studies
Economic dispatch
Power plant control

ADVANCED APPLICATION

Management Operating System

Many utility companies are now investigating online random access systems with customer records maintained in a master file at the central office. Terminals at outlying branch and district offices connect to this master file and are used to handle customer inquiries initiating service orders, and in some instances, transmit cash payments.

SPECIAL CONSIDERATIONS

Public utilities with 75,000 or more consumers are potential candidates for computers.

All companies having or anticipating having tape systems for accounting are prospects for random access files and TELE-PROCESSING[®] equipment. Utilities with 100,000 or more consumers are prospects for the 1620 for engineering and the 1710 for economic dispatch.

Any utility having a power plant with steam turbine generators over 100,000 kilowatts in capacity is a prospect for a 1710.

IBM PUBLICATIONS

A Punched Card Service Order Procedure, general information manual (E20-0003). The basic procedure described encompasses two service order department punched card documents: a master service location record card and a customer service order card. The same system is also outlined for application to a decentralized or district office operation.

Monthly Reports -- Ledger Records, application bulletin (H20-0024). The monthly reports and ledger records processed on IBM equipment by the Connecticut Light and Power Company are described in speech format. The sources of journal entry detail are described and the various reports are listed as well as illustrated.

Stores Accounting, application bulletin (H20-0025). The Consumers Power Company's stores accounting application is described in speech format. Stores accounting and control problems are presented together with the solutions incorporated in the punched card procedure.

Plant Accounting, application bulletin (H20-0022). Presented in speech format, the plant accounting application performed by the Commonwealth Edison Company is described. Some of the topics covered are assignment of costs to plant units, maintenance of unit property records, and a comparison of expenditures plus open commitments with budget authorizations.

Transformer Loading Analyzation, application bulletin (H20-0036). The distribution transformer load analyzation procedure performed by the Iowa Electric Light and Power Company is described in speech format. Study factors that must be considered to implement the system are described: type of system, desired results from a load study, available source records, etc.

Mark-Sense Meter Reading, application bulletin (H20-0032). The Hartford Electric Light Company's billing procedure utilizing the mark-sense meter reading card is described. The 604, 419, 514, 403 and 77 are employed.

Centralized Accounts Receivable, application bulletin (H20-0049). A comparison is made of an accounts receivable operation prior to centralization and the same operation after centralization. The company discussed is the Wisconsin Public Service Corporation. Card and report forms are included.

Auto Cost Record, application bulletin (H20-0050).

Responsibility Accounting, application bulletin (H20-0026).

Stockholders Records, application bulletin (H20-0021).

IBM 1418 Optical Character Reader for Public Utilities (320-1315).

Mechanical Cash Posting, application bulletin (H20-0033).

Material and Supplies, application bulletin (H20-8016).

Property Records, application bulletin (H20-0029).

Construction and Retirement Accounting, application bulletin (H20-0030).

Property Records, application bulletin (H20-0028).

Section	Area	Page
04	08	03

IBM 1400 Series Data Processing Systems Using Disk Pack Storage for Materials and Supplies Accounting in Public Utilities, general information manual (E20-0260).

STANDARD PERIODICALS AND TEXTS

Gas Facts. Published annually by the American Gas Association, Bureau of Statistics, 420 Lexington Avenue, New York 17, N. Y. A statistical record of the gas utility industry for the previous year, discussing energy reserves, production, transmission and distribution, underground storage, sales and utilization, finance, labor, prices, Canadian data, and a glossary of industry terms.

Rural Electric Fact Book, National Rural Electric Cooperative Association, 2000 Florida Avenue, N. W., Washington 9; D. C., 1960. Contains tables, charts and maps displaying pertinent information in addition to extensive narrative on such subjects as utility tax benefits, state rural electric programs (by state), publications, and the federal power program.

Statistics of Electric Utilities in the U. S., Federal Power Commission, Washington, D. C.

I Want to Know about the Electric Industry, Edison Electric Institute, 750 Third Ave., New York 17, N. Y.



NEWSPAPER PUBLISHING

INDUSTRY DESCRIPTION

The single most important group in the printing industry is made up of newspaper publishers. There are 1,760 daily and 3,158 weekly newspapers in the United States employing some 850,000 persons. Competition with other media of information has caused newspapers to spend \$100 million during each of the last three years for new equipment and new plant construction.

The greater part of a newspaper's revenue comes from advertising rather than the sale of the paper itself.

A newspaper, like a large corporation, is usually broken down into five major divisions:

1. Circulation has responsibility for building the number of readers and for proper distribution, both local and out of town.
2. Advertising encompasses not only the sale of space, but also editing and costing functions.
3. Newsroom, directed by the editor, handles wire service, city news, editorials, and sports in addition to makeup and proof work.
4. Production is where pages are typeset and made up, and the paper is printed. Inventory control is a major function of this department.
5. Business staff not only deals with finance and accounting, but acts as a service group to other departments. Part of its responsibility is to bill and collect money owed.

APPLICATIONS

Circulation Accounting

Basic route draw records are stored centrally. All statistics, billing, transportation, routing and press requirements are affected by draw record changes. Once the draw changes enter the system and edition deadlines occur each day, the system automatically produces press order requirements, galley route cards and tape labels, mail subscription schedules, transportation routing and loading schedules and statistical summaries.

Display Advertising

Tight credit controls that still allow operational flexibility are a must in display advertising. Billing is based on the advertisements that actually appear in the newspaper. Corrections to advertising orders are entered after printing.

Calculations and expansions of accounts receivables can be made either daily or at billing time. Accounts receivable controls are set up.

Classified Advertising

Classified advertising involves the integration of production and accounting functions. Information on classified set copy is entered in unhyphenated and unjustified form (the printing not spacially arranged as it would be in actual newspaper context). The classified system determines when an ad is to run, when it is to be pulled, and how the content will be arranged. It also determines whether or not the account or person ordering the ad is delinquent in other payments or has exceeded credit limitations. In addition the system ascertains how much should be charged. The production output will be a paper tape of text material in hyphenated and justified form for ads which must be included with yesterday's lineup for input to linecasting machines.

Additional accounting applications include payroll, accounts payable and general ledger.

Production

The use of a computer in the area of hyphenation and justification will increase the productivity of operators perforating linecasting machine tape. This can be done by eliminating several factors: the decision time, the rubout time for light or loose lines, and the extra key depressing for type-setting codes required when justification is done manually.

Future production applications include:

1. Computer-controlled linofilm -- display advertising lineage.
2. Page space bookkeeping -- keeping track of availability and the disposition of page space assigned for editorial department use.
3. Output control and load balancing -- allocating any given story at a point in time to multiple linecasting machines to expedite production.

IBM PUBLICATIONS

Subscription Fulfillment on Magnetic Tape Processing Systems, reference manual (B20-8134). Subscriber data is maintained on a central file, used to produce the labels affixed to the periodical, to bill the customer, to ask for renewal, etc.

Section	Area	Page
---------	------	------

04	09	02
----	----	----

Circulation Accounting for Newspapers, general information manual (E20-0006). A detailed breakdown of the various means of distribution is included in this introduction to newspaper circulation accounting. Many items are covered: circulation billing, balance forward, cash payment, carrier insurance charge, subscriber insurance charge, etc.

Newspaper Accounting for the Providence Journal, general information manual (E20-0099). The Providence Journal Company uses unit record equipment to process newsprint, inventory control, display, contract classified and transient advertising, circulation accounting, payroll and labor distribution and accounts payable expense distribution.

The Houston Post Data Processing System, application brief (K20-1094). Billing, accounts receivable, payroll and sales analysis are illustrated with report and card forms. The Houston Post has a circulation of 200,000.

Payroll and Job Cost Accounting at Fawcett-Dearing Printing Company, application brief (K20-1230). The rate of pay for each type of work is stored in the IBM 305. Hours worked, recorded on punched cards by the employees, are converted into gross pay for 1,300 persons. All taxes are stored until the preparation of periodic reports.

The Good News Broadcasting Association, Inc. -- Contribution and Subscription Accounting, application brief (K20-1264). IBM equipment is used for addressing literature as well as for writing cash listings, receipts to contributors, and statistical reports. Contributions for religious work and subscriptions to a monthly newspaper are encompassed in the data processing installation.

STANDARD PERIODICALS AND TEXTS

Editor and Publisher

Printers Ink

LIFE INSURANCE

INDUSTRY DESCRIPTION

The life insurance industry is made up of a number of major divisions including ordinary life, debit insurance, accident and health, group insurance, and Blue Cross and Blue Shield. There are 1,400 - odd companies in the industry. Some are mutually owned by the policyholders who elect their own boards of directors; others are owned and controlled by stockholders. (Consulting actuaries also form an important segment of the industry.)

The top ten companies do about 75% of the business. Seven of the ten largest are mutuals. The mutuals, representing only 10% of the total companies, hold 65% of the industry's business. The remaining 35% is handled by the stockholding companies, which account for 90% of the total companies.

The industry is very well organized and conducts many group and association meetings. A full-time commissioner of insurance in each state enforces the particular regulations for that state. Several of the more important associations are:

- Life Office Management Association (LOMA) -- primarily interested in home office procedure, administration, and education of personnel. The automation committee in LOMA is very strong and active.
- The Insurance Accounting and Statistical Association (IASA) -- which includes both life and property and casualty insurance companies.
- The Society of Actuaries -- which studies the whole range of problems regarding the actuarial side of the business.

APPLICATIONS

Policy Issue

Insurance policies are automatically written on data processing equipment. Master files are maintained containing all data relevant to policyholders -- name, address, premium amounts, due dates, etc.

Premium Billing

Bills are written whenever insurance premiums are due -- monthly, quarterly, etc. Billing dates as well as premium amounts are controlled by file searching.

Premium Accounting

Records are kept of premium payments with accumulated premium amounts as a by-product.

Valuation

The cash value and paid-up insurance amounts are calculated -- usually annually. These totals are derived from premium accounting records.

Dividend Calculation and Accounting

Dividends are calculated and records updated to reflect latest amount paid to policyholders.

Mortgage Loan

Records are kept of money loaned for mortgage purposes, payments made, outstanding mortgage balance, and accumulated payments.

Agent's Commission

Commissions are calculated on the basis of premium amounts sold by agents. Commission checks are automatically printed.

Policy and Premium Loans

Records are kept of loans made to policyholders as well as payments made by them. Policy and premium amounts accumulated govern the loan limits permissible.

Agency Statistics and Production

Records are kept of policy and premium amounts by salesman distribution. Cumulative production figures are also kept.

Actuarial Calculations

Premium amounts are calculated on the basis of statistics such as age and sex, with actuarial weighting factors such as the risk element as a part of the formulas.

Investments

Records are kept of monies invested by the insurance company in a variety of locations. Expected dividend and interest returns are calculated.

SPECIAL CONSIDERATIONS

- The trend is to include more information on the premium notice such as policy loan balance and interest due, dividend balances and current dividend. Several companies also include the increase in the cash value of the policy.
- Timing consideration is vital in the answering of inquiries on policy status.
- Companies that write participating policies will have a dividend procedure.
- Key factors in judging the data processing potential of a given company are the number of policies it holds and the number of premium billings per month. Frequency of billing will depend on the "mode" selected by the insured.

IBM PUBLICATIONS

IBM Series 50 Data Processing Machines for Ordinary Life Insurance, reference manual (B20-0067). A comprehensive treatment, including wiring diagrams, of the Series 50 approach to ordinary life insurance.

Consolidated Functions at Government Personnel Mutual Life Insurance Company with the IBM RAMAC 305, general information manual (E20-2058). Basic underwriting management reporting, and commission accounting, using the IBM RAMAC 305, are covered. This manual would be very helpful to 1440 users. (Small systems.)

IBM RAMAC 1401 and 1410 with Tapes, for Ordinary Life Insurance Consolidated Functions, general information manual (E20-0216). Applies the 1401 to a 75,000-policy company and the 1410 to a 300,000-policy company. (Medium systems.)

IBM RAMAC 1401 and RAMAC 1410 Data Processing Systems for Combination Life Insurance Companies, general information manual (E20-0222). Applies the 1401 to a company with 50,000 ordinary life and 1,000,000 weekly premium policies. Applies the 1410 to a company with 150,000 ordinary life and 2,500,000 weekly premium policies. (Medium systems.)

Consolidated Functions of Ordinary Life Insurance with the IBM 7070 and IBM 1401 Data Processing Systems, general information manual (E20-8025). Explores the 7070 with offline 1401 for file maintenance, transaction processing, document preparation, and weekly commission procedure. (Large systems.)

The MAPS Index should be checked for a complete list of all life insurance publications, reprints, etc.

STANDARD PERIODICALS AND TEXTS

Best's Insurance News, Life Edition, monthly, Alfred M. Best, Inc., 75 Fulton Street, New York 38, New York. Items of interest on insurance management, sales, education, loss control, office methods, etc., are contained in this news publication.

The National Underwriters, Life Insurance Edition, weekly, Chicago, Illinois. Articles relating to the current state of the business, legislation by the government, personnel changes, editorial comment, are a few of the topics contained in this underwriter's reference work.

Gregg, Davis W., ed., Life and Health Insurance Handbook, Richard D. Irwin, Inc., Homewood, Illinois, 1959. A comprehensive reference source and guide to current practices and procedures in all major phases of life and health insurance by 69 contributing authors and 40 consulting editors. Ninety percent of the text is in narrative form, the remaining 10% comprises appendixes of charts, tables, and report forms.

The Interpreter, monthly, Insurance Accounting and Statistical Association, P. O. Box 139, Kansas City, Missouri. The Interpreter is divided into two sections -- one for fire and casualty, the other for life and accident and health. The magazine frequently contains articles on computer installations and applications.

Angell, Frank J., Insurance, Principles and Practices, Ronald Press, New York, New York. Covers both life and property and casualty insurance.

Dauids, Lewis E., Dictionary of Insurance, Littlefield, Adams, and Company, Paterson, New Jersey.

Best's Life Insurance Reports, annual, Alfred M. Best Company, Inc., 75 Fulton Street, New York 38, New York. Comprehensive statistical reports on the financial position, history and operating results of legal reserve life insurance companies, fraternal benefit societies and assessment associations in the United States and Canada.

Best's Insurance Reports, annual, Alfred M. Best Company, Inc., 75 Fulton Street, New York 38, New York. Comprehensive data on the financial position, history and transactions of over 1,400 insurance companies and associations in the United States and Canada.

FIRE AND CASUALTY INSURANCE

INDUSTRY DESCRIPTION

The fire and casualty insurance industry includes many diversified divisions. Automobile, fire, casualty, homeowner and title policy writing are among the more familiar aspects of the business. Not so well known to the public are workmen's compensation, inland and ocean marine, bonding, surety, welfare and pension underwriting. Accident and health policy writing is done by either casualty or life insurance companies.

Casualty companies fall into two categories: (1) mutual companies owned by the policyholders who elect the board of directors, and (2) corporate organizations whose stockholders elect the board of directors.

Two major methods of marketing are used by the industry. In the direct writer approach the agent works for one company only though he may be independent. The company using direct writing deals directly with the insured in handling premium payments and renewals. Agency companies market their policies through independent agents who own and control the business. In this case premium payments and renewals are usually handled by the agents. Many agency companies, however, have a special department that handles auto policies on a direct writer basis.

Every state has its own fulltime commissioner of insurance whose duty it is to see that state regulations are enforced. An accurate count of property and casualty companies under the jurisdiction of the commissioners is difficult to obtain since many small companies write only in specific counties or cities.

Property and casualty organizations maintain close correspondence with life companies through the Insurance Accounting and Statistical Association to which both types of underwriters belong. The National Board of Casualty Underwriters establishes premium rates for most agency companies in regard to automobile, burglary, general liability and other kinds of insurance.

APPLICATIONS

Rating and Coding

Premium rates are applied according to coverage amounts, risk elements, and other weighting factors. Such rates may be kept in magnetic core, tape, or disk storage. Coverages are coded for company and bureau reports.

Premium Billing, Accounting and Collection

Files containing the receivables, master policy, and statistical information produce the premium billing information and concomitant accounting data as required. Totals by state and line are accumulated. Appropriate information relevant to each policy is written on work tapes or punched in cards for subsequent distribution.

Account Checking

Policy inquiries or changes involve a search of master policy files which must be updated to reflect transactions. Simultaneously, policies due for renewal or expiration are selected. Losses exceeding established ratios are noted for management.

Loss Accounting and Reserves

To accomplish loss distribution accounting, loss data must be summarized by line, term, agent and other factors. Losses are applied against the agent's commission and carried forward to the annual statement, categorized as direct, assumed or ceded.

Statistics

Special accumulations can be gathered from the regular files for use in statistical study: for example, large risks involving numerous policies and losses exceeding established ratios. Such statistics furnished to the actuary lead to realistic premiums.

Agents' Statements (independent agents in the field)

Agents' statements, usually written monthly, take into account premium payments, chargebacks and changes accumulated over the monthly period. These statements, or worksheets, contain data used to calculate commission and/or chargeback, and eventually contribute information used in check writing.

SPECIAL CONSIDERATIONS

In planning for a data processing system some consideration should be given to the number of states in which policies are written, the degree to which the company is decentralized (number of

branches and agents), the different lines of insurance offered, the number of outstanding policies, and the billing frequency. These major criteria will determine the basic approach to be taken in the system design.

PUBLICATIONS

Butteville Insurance Company, application brief (K20-0108). Twelve thousand policies are processed for a small company on unit record equipment consisting of the 26, 82, 402 and 514.

IBM Series 50 Data Processing Machines for Fire and Casualty Insurance Agencies, reference manual (B20-0075). A description of insurance agency operation and procedures from the invoice preparation to final end-of-the-month reports are included. A few of the many reports discussed are the cash listing of premium payments, the producer commission statement, the expiration list, and the invoice, the latter being prepared on the 826 Typewriter Printing Card Punch.

IBM RAMAC 1401 for Fire and Casualty, general information manual (E20-0218). Two approaches are discussed: the card system and the tape system. Among the many functions described are rating and coding of policies, policy writing, renewal notice preparation, and daily distribution to agents' records and control totals.

IBM RAMAC 1410 Tape Systems for Fire and Casualty, general information manual (E20-0220). For those companies which issue policies at central locations, the 1410 provides a means of automatically performing the necessary rating, coding and document preparation functions. Premium accounting and billing, updating the master policy tape, and renewals are among topics discussed.

IBM RAMAC 7010 Tape System for Fire and Casualty Insurance, general information manual (E20-6286). The system discussed provides an approach to fire and casualty insurance for large companies.

IBM 1050 Data Communications System for Fire and Casualty Insurance, general information manual (E20-8130). Describes the equipment itself and configurations to provide compatibility with agencies, claims offices, branch offices and home office departments for data communications reflecting the current status of policyholders and agents.

KWIC Index to IBM Marketing Publications (320-1621-1), current edition.

MAPS Index. Check for complete list of all fire and casualty publications, reprints, etc.

See list of Standard Periodicals and Texts in area entitled "Life Insurance".

MANUFACTURING INDUSTRIES

INDUSTRY DESCRIPTION

The complex of manufacturing industries can be characterized by three basic manufacturing operations. The process industries are concerned primarily with the production of a material, the fabrication industries with the formation of materials into specific shapes, and the assembly industries with the construction of products using basic materials and fabricated products.

Subdepartments within the fabrication and assembly industries are:

1. Transportation equipment -- ordnance, motor vehicles and parts, ship and boat building, railroad equipment, wholesale auto distributors.
2. Machinery and Electrical.
3. Industrial and Consumer Commercial Products -- wood products, furniture, stove, clay, glass products, precision assembly (instruments).
4. Fabrication -- fabricated metal products, construction, engineering services.
5. Primary Metals and Mining.

There are 22 separate billion-dollar fabrication and assembly industry companies, and their total annual sales of \$51.1 billion represent 30% of the total fabrication and assembly industry sales.

Subdepartments within the process industries are:

1. Petroleum and Industrial Chemicals
2. Food, Drug, Chemical Products -- meat, canning, packaged foods, grain, sugar, drugs, soaps, cleaning, cosmetics, paint, agricultural, chemicals, rubber and plastic products.
3. Textile, Apparel, Paper -- textiles, apparel, leather, forestry, pulp and paper.

The 18 individual billion-dollar process industry corporations in the United States have total annual sales of \$40.4 billion which represents 27.8% of the total process industry sales.

The above classifications do not have to be observed as the only organizational breakdown for the manufacturing industries. The criteria used are (1) the similarity in the application of data processing within the industry and (2) the retention of integrated company operations within a given industry department. For this reason the primary metals industry is found associated with metal fabricators,

and the apparel and paper fabricators with the textile and paper industry. With this division the separation could be more correctly identified as durable and nondurable manufacturers.

Total sales for all manufacturing companies amount to about \$337.8 billion annually, representing 25% of the gross national product. The total assets of all the manufacturing companies are \$248 billion.

There are 7,177 manufacturing companies with sales of \$1 million or more. These companies account for \$286.7 billion in annual sales or 83% of all manufacturing sales. They have combined assets of \$225.8 billion or 91% of the total for all manufacturing industries.

Of the 7,177 manufacturing companies with sales of \$1 million or more, 1,819 companies are publicly held and have total annual sales of \$231 billion (67% of the manufacturing total); 5,358 are privately held and have total annual sales of \$55.4 billion (16% of the manufacturing total).

The 40 separate billion-dollar manufacturing corporations represent 80% of all the billion-dollar corporations in the United States. Their total annual sales amount to \$95.6 billion or 28.3% of the total annual sales in manufacturing industries.

FABRICATION AND ASSEMBLY INDUSTRY APPLICATIONS

Management Operating Systems

The Management Operating System is a comprehensive systems approach to company planning and control. Once planned, it need not be implemented all at once. Many companies attack their most serious problems first -- inventory management, for example -- and take on additional functions on a scheduled basis.

In the Management Operating System, applications are not treated as isolated procedures. Major activities or company functions are the basis of data processing planning, rather than the work of specific departments within the company organizational structure.

The functions of a Management Operating System are engineering, forecasting, materials planning, inventory management, purchasing, scheduling, dispatching, quality assurance and operation evaluation. Each function incorporates the related accounting and manufacturing applications.

Management Science

Inventory management simulation
Critical path technique
Production line balancing
Capital investment

Technical Science

Numerical control
Quality assurance
Design automation
Process control

Accounting

Accounts payable
Accounts receivable
Cost accounting
General accounting
Inventory accounting
Personnel
Payroll
Property accounting

Engineering

Research
Development
Design
Technical
Information retrieval

Sales

Order writing
Billing
Sales analysis

PROCESS INDUSTRY APPLICATIONS

Management Science

Inventory management simulation
Capital investment evaluation
Corporate operation simulation

Financial Accounting

Accounts payable
Accounts receivable
General accounting
Inventory accounting
Personnel
Payroll
Property

Research and Development

Research calculations
Process engineering calculations
Process equipment design
Technical
Information retrieval
Statistical analysis
Simulation
Design automation

Operational Planning

Forecasting
Materials planning
Raw material evaluation
Production planning
Plant scheduling

Operational Control

Inventory control
Unit process control
Product quality control
Performance or yield evaluation
Product cost accounting
Facilities utilization
Field performance evaluation

Facilities Engineering

Construction and maintenance scheduling
Job cost estimation
Materials accounting and control
Job cost control
Equipment maintenance accounting and control
Numerical control

Purchasing

Purchase order writing
Purchase order status
Vendor analysis
Supplier conformance

Sales and Distribution

Order writing
Credit control
Billing
Sales analysis
Commission accounting
Distribution inventory control
Delivery scheduling

Section	Area	Page
04	12	03

PETROLEUM INDUSTRY APPLICATIONS

Operational Control

Pipeline scheduling
 Crude evaluation
 Refinery scheduling
 Process control
 Gasoline blending
 Daily yield accounting
 Product cost accounting
 Package product scheduling

Engineering Control

Job scheduling
 Material stores accounting and control
 Job cost control
 Equipment maintenance accounting and control
 Budget control

Financial Control and Reporting -- Production

Run ticket calculation
 Lease production summaries
 Production statistics
 Royalty payments
 Joint venture billing

Financial Control & Reporting -- General

General ledger
 Accounts payable
 Property accounting
 Payroll
 Capital stock accounting
 Pension and welfare fund accounting
 Purchasing
 Tax accounting

Marketing

Order entry
 Price checking
 Wholesale accounts receivable
 Sales analysis
 Commission accounting
 Retail accounts receivable
 Credit control
 Product stock control
 Degree day scheduling

Technical and Research

Exploration survey calculations
 Reservoir calculations
 Equipment design calculations
 Process design calculations
 Statistical analysis
 Plant simulation
 Quality control
 Information retrieval

SPECIAL CONSIDERATIONS

Automation has penetrated the process and fabrication industries to a greater extent than the assembly industries. It has lowered the percentage of labor expense in the total manufacturing cost and has increased the value of plant investment per employee. These factors directly affect the displaceable clerical costs, thus increasing the potential for the conventional data processing applications. High values in capital assets point to the need for applications directed toward control of maintenance functions.

IBM PUBLICATIONS

Production Line Balancing with the 1620, general information manual (E26-5593). The production line balancing problem is described, as well as the IBM 1620 library program designed to meet the situation. The preparation of input data is explained in detail and sample output data is shown in addition to a block diagram of the program.

AUTOLOTTER for the IBM 1620 Data Processing System, application program bulletin (H20-4209). AUTOLOTTER, a program for a card or paper tape 1620, is used for plotting graphs of computed results on a modified 870 system. Examples of the uses of AUTOLOTTER, the format of the data statements and the resultant graphs are included. Machine configurations and execution times are also described.

The Ohio Art Company -- IBM 409 Accounting Machine for Billing and Label Preparation, application brief (K20-1402). The Ohio Art Company is the largest manufacturer of metal lithographed toys in the United States. It uses the IBM 409 Accounting Machine to print and punch shipping labels, thereby lowering the order-filling cycle to 24 hours. Ohio Art's prebilling operation and freight car-loading procedure are described.

Data Collection at Beech Aircraft Corporation, TELE-PROCESSING systems bulletin (H20-8065). The Beech Aircraft Corporation uses the IBM 357

Data Collection System to record employee attendance, job time, production order control data, tooling supplies requisitions information, and tooling orders data. Procedures, card forms and reports are included.

Management Operating System at George Gorton Machine Company, general information manual (E20-0081). The operations of a machine tool company employing 250-500 persons are described. Installed since 1943, George Gorton has developed a completely automated production control system. A glossary of terms peculiar to the industry is included.

Numerical Control Bulletin (H20-6250). A background of machine tools and numerical control information is provided in question-and-answer form. AUTOPROMPT, AUTOMAP, AUTOSPOT and AUTOPROPS programming systems are included. These programs help in preparing instruction tapes that control the operation of machine tools.

Inventory Management Simulation for Manufacturing Industries, general information manual (E20-8063). The IBM Inventory Management Simulators are designed to guide management in selecting policies and making decisions which provide maximum profits from inventory dollars. To obtain these benefits, a reasonable understanding of inventory management simulation is required.

Preventive Maintenance and Cost Control, general information manual (E20-8029). Proper timing of preventive maintenance lowers the number of machine breakdowns, thereby reducing manufacturing costs. Up-to-date records of maintenance and repair activities produced by data processing equipment are applicable to any type of equipment or industry.

Capital Investments, general information manual (E20-8085). Capital investment programs are described. These programs enable management to automatically predetermine the effect which Federal income taxes, different methods of depreciation, and various methods of financing have on additional earnings or savings. Rates of return for various periods of time that a given proposal is operative are also derived.

IBM Data Collection in the Factory, general information manual (E20-8076). The use of data collection in a typical factory is described starting with attendance reporting, then following an order and material through the shop; card layouts and sample reports of major applications are included.

Management Operating System -- Quality Assurance for Manufacturing Industries, general information manual (E20-8081). The objective of the quality assurance function is to define, maintain and improve the quality of finished products at the lowest

cost. Data processing equipment can be effectively used to attain this objective by analyzing and interrelating data from four functions: product design, purchasing of parts, manufacturing and assembly, and product performance.

Engineering Data Processing for Manufacturing Industries, general information manual (E20-8106). The advantages of using IBM data processing systems for the organization, processing and dissemination of engineering and manufacturing technical data are described. The functions of engineering and the flow of engineering information from development to production are shown. A flowchart portrays the interrelationship of engineering functions, the dependency of one function on the output of another, and the typical sources of data.

SUPPORT MATERIAL FOR MANUFACTURING INDUSTRIES

Slide Presentations

MOS for the Process Industries
GARMAC - Garment Industry

Filmstrips

Manufacturing Control - IBM Poughkeepsie
Sales Forecasting
Management Science
Job Shop Simulation
Distribution Accounting
New Power for Numerical Control

Movies

Data Processing at Lockheed Aviation
Computer Programming for N/C Tools
(AUTOSPOT-AUTOMAP)
Design Automation at IBM

Flipcharts

GARMAC - Garment Industry

STANDARD PERIODICALS

Fabrication and Assembly Industries

Factory
Aviation Week and Space Technology
Electronics
American Machinist/Metalworking Manufacturing
Metalworking News
Steel
Automation

Section	Area	Page
04	12	05

Process Industries

Chemical and Engineering News
 Chemical Engineering Progress
 The Oil and Gas Journal
 Women's Wear Daily

Fortune Magazine, Time Inc., 540 N. Michigan Ave., Chicago, Illinois, \$10/year, published monthly. Contains many informative articles on all areas of business. A statistical digest of all manufacturing companies is available from the publisher upon request.

Iron Age, Chilton Publishing Co., Chestnut & 56th Sts., Philadelphia, Pa., \$2/year, published monthly. Covers all phases of the economy that are dependent upon steel.

Industrial Marketing, 740 Rush St., Chicago 11, Illinois, \$3/year, published monthly. Contains general marketing articles applicable to the manufacturing industries.

Instrument and Apparatus News, Rimbach Publications, 845 Ridge Ave., Pittsburgh 12, Pa., published monthly.

Instruments and Control Systems, Rimbach Publications, 845 Ridge Ave., Pittsburgh 12, Pa., published monthly.

Automatic Control, E. H. Nelson Publishing Co., 202 E. 44th St., New York 17, N. Y., \$10/year, published monthly.

American Machinist, McGraw-Hill, 330 W. 42nd St., New York 36, N. Y., published monthly.

Automatic Machining, 65 Broad St., Rochester, New York, published monthly.

Iron & Steel Engineer, 1010 Empire Building, Pittsburgh 22, Pa., published monthly.

Paper Trade Journal, 49 W. 45th St., New York 36, N. Y., published monthly.

Materials in Design Engineering, Reinhold Publication, 430 Park Ave., New York 22, N. Y., published monthly.

Automotive Industries, Chilton Publishing, 56 & Chestnut Sts., Philadelphia 39, Pa., published semimonthly.



AEROSPACE

INDUSTRY DESCRIPTION

The aerospace industry is composed of companies involved in the development, design and manufacture of manned aircraft, missile systems and space systems. The backbone of the industry consists of the older airframe and aircraft manufacturers who pioneered in the design and manufacture of manned aircraft. These companies, along with other companies, are now deeply involved in the research and development of missile systems and space systems as well as the design and manufacture of space vehicles. Some of these older companies no longer manufacture manned aircraft and are involved solely in the missile and/or space system effort.

Few industries have changed direction as dramatically and in so short a time as the aerospace industry. In fact, the name "aerospace industry" has become acceptable only in the past five years.

Key factors that highlight these changes are the increase in research as compared with production, the decline in aircraft, the rapid rise of missiles, and the prospect of a leveling out in missile activity accompanied by a substantial increase in spacecraft.

These changes have been associated with a shift from mass-produced aircraft to fewer but more complex vehicles. For example, in 1953 the industry produced in excess of 10,000 military aircraft; in 1954 this number fell to about 2,000. As a result, the existing floor space in the industry has been sharply reduced in recent years. Massive assembly facilities such as those of Convair at Fort Worth, Boeing at Wichita, and Martin at Baltimore are no longer economically feasible with the industry's change in product mix.

The changes in the industry have produced a significant change in employment. There has been a reduction in the number of production workers accompanied by a growing demand for highly skilled technicians, engineers and scientists. While the change is true of all manufacturing, it has attained substantial significance in the aerospace industry. For example, from 1950 to 1959, production workers in all manufacturing industries declined from 82.3 to 75.7%. In the aerospace industry this decline was from 76.5 to 56.4%.

Although there has been a major reduction in the number of manned aircraft produced, the development, design and manufacture of manned aircraft still accounts for a sizable portion of Department of Defense spending. In 1960, the Department of Defense spent approximately \$6.9

billion in manned aircraft projects. National defense expenditures for missiles in 1960 amounted to \$5 billion.

Spaceprobes and manned space vehicles now make up a vital segment of the aerospace industry. As the manned aircraft effort declines and missile systems activity levels out, procurement of space systems should fill this void. In the fiscal year 1960, costs for space systems were slightly in excess of \$500 million. Estimates for this type of yearly expenditure run in excess of \$6 billion by 1970 or a twelvefold increase. The estimated expenditure for putting an American on the moon is \$20 billion.

SPECIAL CONSIDERATIONS

The aerospace industry spends approximately \$8 million per month for IBM services. Most of the industry (95%) consists of approximately two dozen corporations serviced by three regions. Approximately 53% of IBM's installed points in the aerospace industry are located in the Western Region, 22% in the Eastern Region and 15% in the Midwestern Region. The accounts in the industry represent massive customer installations. The key customers spend \$600,000, \$700,000, \$1 million and \$1.5 million per month for IBM services. These customers are extremely proficient in technical application of our systems. The aerospace industry is highly volatile and defense-oriented. It must react quickly and efficiently to requirements of the various defense agencies and programs.

APPLICATIONS

Forecasting

Forecasting involves the determination of the magnitude and timing of customer requirements and the impact of these requirements on the resources of the firm. The method of forecasting depends on the amount and type of information to be analyzed. Generally, the output will be a report form involving historical information and analysis of:

1. Product line capability
2. Type of contract programs (Apollo, Gemini, etc.)
3. Areas of technology
4. Competitive capability

From an analysis of this information, various confidence factors are developed and used in deciding company strategy in contract negotiations.

Financial Planning

The financial plan is a summarization of the data relating to the contracts and programs projected in the business forecast. The categories of business are:

1. Firm contracts
2. Follow-on business (contract extension)
3. New business

Contract and project data describing the project and establishing the means by which billing and payments will occur includes customer, program, weapon system, contract, status, etc. All the processing and control of the financial information is performed by the computer. Outputs include expenditure projections, departmental burden and general administrative expense information, research projections, dividend and interest information and, in general, all the company's cash-flow information.

Resource Planning

An approach to resource planning involves the use of computers to select contract opportunities which tend to optimize the goals established by management (growth rate, return on investment, long-range opportunities, maintenance of essential skills, etc.). Simulation can be used to develop the combination of programs which will most closely approximate the stated objectives. These objectives are assigned relative weights to indicate the order of preference. On the basis of these statements, the computer examines the various combinations of potential programs and selects the best combination.

Scheduling

Scheduling is the formulation of a plan specifying the desired objectives, the sequence of activities required to meet these objectives, and the dates by which key events or milestones must be reached in order to ensure that the desired objective is completed in the allotted time. Some scheduling techniques using data processing equipment are:

1. Life-cycle analysis: Life-cycle analysis is used to forecast and monitor manpower requirements and time distribution on R&D projects. It is based on the hypothesis that gross manpower utilization patterns on an R&D project exhibit regularities which are a function of the way engineers and scientists work on such projects.

Data processing equipment is used to categorize and summarize historical data in order to develop the life-cycle curves. The forecast is arrived at by applying the curve functions to the actual manpower utilization.

2. Line of balance: Line of balance is a display technique used to present a comparison of scheduling and budgeting data with actual performance data. The use of data processing equipment for line-of-balance reporting provides more current, correct information for effective evaluation of progress.

3. Network scheduling: The best known technique in this area is Program Evaluation and Review Technique (PERT). Project schedules are described in network form for analysis to determine the tasks which are critical to the project's timely completion.

Shop Control

Manufacturing parts lists and assembly and detail routing files are placed in storage by the planning department. An operational scheduling program utilizes these files to prepare manufacturing schedules for each shop order to be released. These scheduled shop orders become the basis for establishing the job order location and machine group queue files. The machine group queue file is then used to prepare shop load forecast reports. Scheduled release dates of shop orders, which take advantage of common setups, tools or material, are adjusted to level the shop loads, and a shop order packet is prepared. Shop order packets include shop orders and shop order cards which accompany each order through the shop. Priorities are assigned either by management or by the scheduling program. A daily work list, showing the shop orders in priority sequence, is prepared for each cost center. Job reporting, done by the operator each time he completes a job, provides the basis for compiling employee and departmental performance reports.

Design Engineering

There are four classes of procedures which occur in the design of an electronic system and for which computer applications have been developed:

1. Analysis: When attributed to design engineering, analysis takes on three meanings:
 - a. Checking the design data against specific rules, with output explicitly indicating violation of those rules.

Section	Area	Page
04	13	03

- b. Evaluating design parameters over a range of increments to arrive at a statistically best configuration or to define the limitation and tolerances of a configuration.
- c. Simulating the system, using the design data, with output implicitly indicating either rule violation or else a relative measure of goodness.

2. Reduction: Reduction is a set of techniques employed to rearrange the logical variables of a function. Terms, in a single logical statement or set of statements describing a function or an entire system, may be manipulated to arrive at a new statement or set of statements logically equivalent to the original representation. However, this new statement or set of statements materially reduces the implementing of hardware requirements or the work to be done.

3. Location and connection: The class of programs in this area includes those which translate the implemented terms of a design statement from their symbolic representation to a physical or geometric representation. This is done by locating the components in space and deriving, or at least recording, the necessary connections between them (etched paths, wires, cables).

Package design locates components on a module. Production design locates modules on a chassis. Circuit and logical design locates blocks on a page so that block diagrams may be printed out. In each case, additional programs route or take note of prespecified routes for the connecting paths. Wiring within airframes and cabling between ground-support modules at launch sites involve so many connection combinations that only a computer can control the volume, keep track of the changes, and provide prompt and useful updated information to the many groups requiring this data.

4. File and maintenance reporting: Master files are originated by design engineering at the time a new design is originated. Changes in design or new input are incorporated into the files, and a report showing such alterations is prepared for the design engineers. The master files, which are used throughout the design process, are the connecting links forming an integrated system.

IBM PUBLICATIONS

The following general information manuals on aerospace information and control systems describe the environments of the functional areas of the aerospace industry and indicate how computer systems can assist in operation and control:

Engineering Summary (E20-8111). Six aerospace engineering manuals and two general information manuals, E20-8112 and E20-8117, are summarized.

Design Information Systems — Structural (E20-8112). Structural design information systems are an important current development in the aerospace industry. A large part of the design information is proprietary, especially the programming data. There is also a great need in the industry for compilation and organization of that information which has been made public.

Design Information Systems — Electronics (E20-8113). Most aerospace companies depend on computer systems to coordinate the efforts and work output of designers of electronic systems. Some aerospace companies are already developing integrated design information systems so generalized as to serve projects that employ varying technologies. The problems and requirements confronting six functional areas of electronic systems design are described: circuit, packaging, system, logic, production and test design.

Technical Information Dissemination and Retrieval (E20-8114). Technical information dissemination and retrieval systems implemented by electronic data processing machines are important functions within the aerospace industry. The sources of information, the functions of an information system, the system itself and further considerations including future systems are discussed.

Engineering Data Control (E20-8115). Engineering data control and release activities are performed by data processing systems. The cycle, which begins with the engineering design process, is completed with implementation procedures.

Configuration Management (E20-8116). A significant characteristic of the aerospace industry is that the design of an aircraft, missile or space vehicle undergoes continual change throughout its development and service life. A consequence of continuing design change is that delivered articles in the same series are rarely identical at any one point in time. The task of controlling, accounting for and reporting the planned actual design of each article throughout its production and service life is called "configuration management". Management control reports, engineering and manufacturing documentation and reliability assurance reports are amply illustrated.

Reliability Assurance (E20-8117). With emphasis on missiles and space vehicles, this manual describes a reliability assurance program based on the analysis of reliability data in (1) prediction (including mathematical models), (2) analysis and evaluation of prototype feasibility models, and (3) surveillance to provide feedback of field performance. Statistical methods for the implementation of a computer analysis of the program are described in detail.

Manufacturing Summary (E20-8118). Relationships of the various major functions in a manufacturing facility are summarized: engineering data control, manufacturing planning and tooling, material control, shop control and product support. Management receives information necessary to control manufacturing facilities from the data processing system output.

Planning and Tooling (E20-8119). Major problems inherent in operational routing in an aerospace company include formulation of the production plan, updating and correcting the production agenda, and communicating the arrangement to the production facility. Included as an adjunct to the system is a procedure for maintaining and reporting the status of all work in the planning department. Estimating the planning hours required for each type of engineering drawing and comparing total estimated planning hours with the actual hours accumulated are important planning tasks.

Shop Control (E20-8121). Aerospace shop control systems are necessarily diverse. The following applications are representative: operational scheduling, shop load forecasting, order writing and release, dispatching of job assignments, job order location, employee performance reporting and machine utilization. Random access storage and two-way data communication terminals are used effectively.

Data System for Limited Production (E20-8123).

Management Summary (E20-8124). This manual summarizes three general information manuals, E20-8125, E20-8126 and E20-8127, and presents, for each area involved, a broad view of the data processing systems applied.

Forecasting and Planning (E20-8125).

Estimating and Pricing (E20-8126). The operations of the estimating and pricing group in proposal preparation, production estimating and gross estimating

are important aerospace functions. In addition, estimating of spare parts prices is done by computer.

Project Scheduling, Budgeting, Evaluation and Control (E20-8127). IBM equipment is extensively applied to project scheduling, budgeting, evaluation and control in the aerospace industry. In the first of these areas, many functions demand attention. Some of them are organization, origination, progress reporting, use of charts, life-cycle, line-of-balance and network-scheduling techniques. PERT-Cost and budgetary systems are other matters of concern. In the evaluation and control area, data processing equipment is used to report variances and to perform other functions

STANDARD PERIODICALS AND TEXTS

Aerospace Facts and Figures, American Aviation Publications, 1001 Vermont Ave., N. W., Washington 5, D. C. \$3.00. Published yearly, this is a facts and figures guide to the aerospace industry.

Aviation Week and Space Technology, McGraw-Hill Publications, McGraw-Hill Building, 33 West 42nd St., New York 36, New York. \$.75/week, \$7.00/year. This is the standard news magazine for the aerospace industry.

Airlift - World Air Transportation, American Aviation Publications, 1001 Vermont Ave., N. W., Washington 5, D. C. \$5.00/yr. Published monthly. News items and facts and figures on the airlines industry, both federal and commercial, round out a current report of the field.

The Official Airlines Guide, 209 West Jackson Blvd., Chicago 6, Ill. A monthly news supplement about the airlines industry, Air Travel, is also published.

ORSA Journal, The Journal of the Operations Research Society of America, Johns Hopkins University, 7100 Connecticut Ave., Chevy Chase 15, Maryland. \$1.50/issue, \$7.50/volume. Published bimonthly, it contains articles in operations research by noted people in the field.

Aerospace Management Magazine, 56th and Chestnut Streets, Philadelphia, Pa.

SERVICE INDUSTRIES

INDUSTRY DESCRIPTION

Service industries produce service to individuals or other businesses rather than manufacture tangible goods for sale. Service industries not discussed here are treated elsewhere for organizational rather than logical reasons. The particular businesses bearing the service caption below belong in a single survey because they are administered as one marketing area.

This single facet of the market includes service bureaus, consulting firms, research and development laboratories, hotels, racetracks and other diverse service industries. Of this heterogeneous group, the service bureaus deal with the widest span of record-keeping applications. Among the reports they produce are entries in their customers' day-to-day accounting ledgers, statistical summations and advisory reports to management. Another type of accounting service uses data processing equipment to produce similar reports but emphasizes the actual bookkeeping.

Consulting firms are essentially divided into two groups — business and management advisory. Management consulting firms, dealing in personnel, time study and similar investigatory functions, have not used data processing to the extent that business consultants have. However, the rapid growth of advisory work suggests that an increase in the use of machines is to be expected. In particular, considerable potential growth lies in the automation of computations performed by engineering consultants. The 13,000-odd firms employing CPAs exert great influence over business in general, and marketing groups are becoming increasingly aware of this fact.

Since the objectives of the consultants and of systems engineers are parallel, mutual understanding between the two is essential to the customer's welfare. Market research and statistical services are also included in the broad consulting structure.

Nonprofit research organizations perform some management consultant functions, but a large part of their effort is concentrated on specialized studies. One example is the Institute of Cancer Research. Research laboratories with profit-making goals are involved in market research, space technology, systems planning and programming systems as well as in management consultant activities.

The remaining service industries cover a broad, diversified group of organizations. Hotels, laundry and linen suppliers, general business services, and nonprofit firms are only representative of the field. Hotels of all sizes — just beginning to outgrow

the traditional accounting applications — are investigating TELE-PROCESSING[®] equipment. They are potential users of data collection and real-time techniques for guest accounting.

For the company providing laundry services to industry or to the consumer, the following criteria are a rough guide in deciding whether or not data processing equipment can be justified: 10 routes, 1,000 accounts or 500,000 gross annual sales.

Variations within the industry are:

- Industrial laundries providing work clothes and other garments to factories, garages, and other businesses.
- Linen supply houses supplying flat goods such sheets and towels to hotels, hospitals, and other industries.
- Family laundries.
- Diaper services.

There are several hundred nonprofit membership organizations in the Eastern Region alone. A list of the more common ones and their purposes follows:

- Business associations engaged in promoting business interests.
- Professional membership organizations committed to advancing the professions of their members.
- Labor unions striving to advance the interest of organized labor.
- Civic and fraternal organizations interested in advancing their causes.
- Religious organizations devoted to spreading their philosophy.
- Charitable organizations providing services to the needy.
- Other nonprofit organizations, such as the AAA, concerned with promoting particular causes.
- Nonprofit educational and research agencies engaged in developing new ideas, methods, or products.

Examples of some other service industries follow:

- Trading stamp companies are a rapidly growing industry. Approximately 15 companies of significant size are already active
- Leasing companies purchase equipment and lease it to their clients.
- Auto repair and service companies primarily rent cars and trucks. Many smaller regional organizations are being formed to compete with the few existing large companies.
- Amusement and recreational services include country clubs and city clubs.

APPLICATIONS

The following list of applications is not complete for any given industry, nor does it cover all categories of service-type businesses. It does indicate the diversity of needs in the service industry field.

Hotels

Guest accounting. — Two vital capabilities in the guest accounting application field are the ability to keep records up to the minute, and the capacity to answer questions on demand. These attributes are especially valuable in reservation booking, guest registration, charge recording from remote hotel locations, billing and guest history maintenance.

Night audit. — Previously the balancing of restaurant, laundry, telephone and room charges required a staff of auditors to perform this function nightly. In an automated system, staff is reduced and control increased because the balancing function is performed automatically whenever payments are made — whether by cash, check, or credit card.

City ledger (accounts receivable). — Billing is done by mail under these circumstances:

- a. To account for services charged by guests who have left the hotel before settling last-minute items.
- b. To account for credit card charges submitted by persons not guests at the hotel.

Food and beverage control. — Almost half of a hotel's revenue comes from the sale of food and beverages. Recently, several hotels have considered control of this income so important that they have automated the process. The purchasing of supplies, the actual food preparation, food and beverage sales analysis, and forecasting future food requirements all necessitate rigid accounting control.

Standard applications. — The standard applications of payroll, accounts payable, and general ledger (expense and revenue) are natural subjects for data processing techniques.

Laundry and Linen Suppliers

Route accounting. — Route accounting includes prebilling and the printing of route lists, two of the basic applications in the laundry and linen supply field. Data processing equipment is especially useful in providing the required volume of repetitive printing.

Accounts receivable. — Accounts receivable are prepared weekly for route drivers. Monthly statements are prepared for customers.

Production control. — Production control in the piecework laundry industry can be accomplished by using prepunched work cards for each employee. A card is attached to each garment processed by the individual worker. A count of the cards each day provides production statistics which furnish the basis for the incentive-type pay system commonly used.

Centralized accounting. — Many companies have multiple branches or depots across the country or throughout given districts. TELE-PROCESSING equipment is being used increasingly to transmit to central locations exceptions to the standard billing routine or normal territory requirements of customers.

Other applications. — Other applications include inventory control, direct mail promotion, payroll, sales analysis, general ledger and accounts payable.

Nonprofit Membership Organizations

For nonprofit membership organizations, the following applications exist: file maintenance, membership billing and direct mail promotion. In addition, labor unions bill members for dues, provide pension lists, and frequently supply lists of temporary employees.

Other Service Industries

Trading stamp companies use normal accounting procedures such as payroll, merchandise control and accounts receivable.

Auto repair and services companies have such applications as billing and automobile maintenance schedules.

Amusement and recreational services use data processing equipment for their extensive billing and accounts receivable activities.

Racetracks apply data processing to on-track betting.

IBM PUBLICATIONS

The Auditor Encounters EDP, general information manual (F20-8057).

Section	Area	Page
04	14	03

Internal Control and Audit In-Line Electronic Accounting, general information manual (F20-2019).

Accountants Service Bureau Financial Statements, application brief (K20-1277).

Hotel Accounting Using the IBM 1440 Data Processing System with TELE-PROCESSING, general information manual (E20-0331).

Domestic Linen Supply and Laundry Company Billing and Route Accounting, application brief (K20-0970).

IBM Equipment for Fund Raising, promotional brochure (520-1475).

A Guide to Processing Campaign Records Mechanically, application manual (320-1394).

Introduction to Racing Associations, application bulletin (H20-0098).

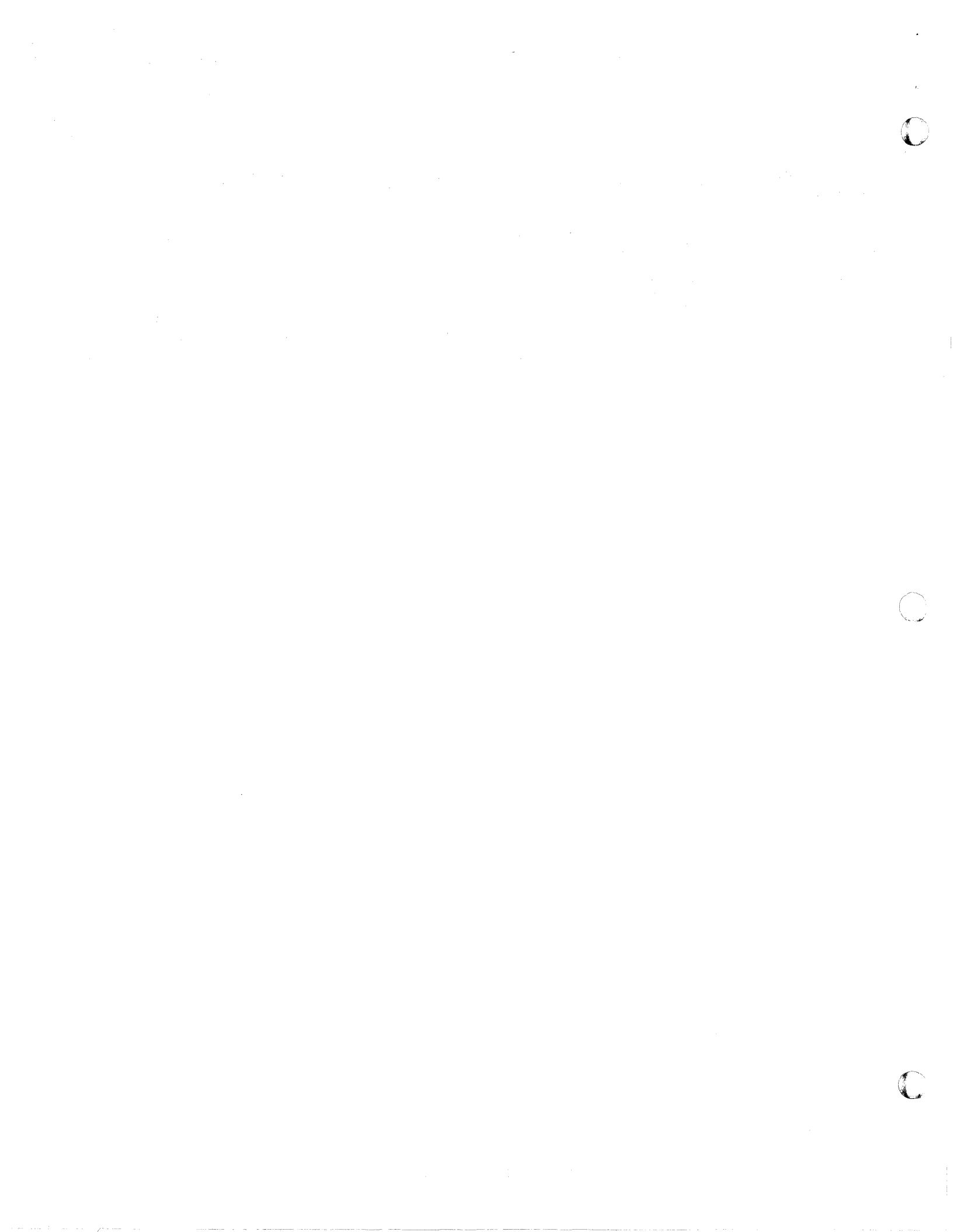
Pari-Mutuel Accounting at Santa Anita Park, application bulletin (H20-0097).

STANDARD PERIODICALS

"Progress in Hotels — The Inn on the Park, Toronto", The Horwath Accountant, Volume 43, No. 6, 1963. A report on recent progress in the hotel processing field.

"Data Processing for the Big and the Small", Hotel Management Review, August, 1963.

"Rental Laundry Management", National Work-Clothes, May, 1963.



STATISTICS

INTRODUCTION

The nature of statistics is such that an enumeration of the specific problems to which it applies is impossible. However, having it in his repertoire, the systems engineer will find that he is equipped with a new approach to problem solving, a rational philosophy for obtaining information and drawing conclusions.

A knowledge of the customer applications of statistics will enable the systems engineer to better understand the customer's operation and place him in a better position to serve his customer effectively. For example, if he understands how a customer uses statistics in such areas as inventory control, data reduction, quality control, forecasting and the establishment of work standards, he can gain a valuable and necessary insight into the customer's requirements.

Statistics is also indispensable in the performance of systems engineering functions such as analysis of data, systems design, and installation problems in data processing systems. For example, in the analysis of data for a potential customer's job requirements, statistics may be used by the systems engineer in load forecasting and the determination of sample sizes. Or, in a full systems study, statistical techniques may be used to formulate projected response times, or to analyze system simulation results. Many uses for statistics are evident in a real-time data processing installation -- for example, the calculation of transaction rates, the analysis of the system's reliability and the determination of decision-making factors such as the effect of significant changes in the performance criteria of a system. In all these areas, the systems engineer will have the opportunity to profitably apply many statistical techniques.

This section provides some formulas and supporting discussion for those branches of the broad field of statistics which have been found to be most useful to systems engineers. These formulas represent a compact summarization of important statistical expressions. The systems engineer is advised not to use these formulas without a substantial understanding of their application. The indiscriminate use of such formulas can lead to many difficulties in the design and analysis of a data processing system.

TERMINOLOGY

Statistics is the science of analyzing and interpreting data. The two subdivisions of statistics are descriptive and inferential. A descriptive statistic is a computed measure of some parameter (property) of a set of values. In the inferential sense, a statistic is an indicator from which a decision can be made.

A variable (sometimes called a variate) is an item which takes on values within some range with a certain probability or frequency, such as height of fourth-grade girls in a school. Variables may be continuous -- measured to any degree of accuracy desired (for example, height); or discrete -- limited in precision to integral amounts (for example, number of children in a family).

A frequency distribution of a variable is a count of the number of occurrences of individual values (the discrete case), or of cases which lie between certain predetermined limits over the range of values the variable may assume (the continuous case). A distribution may be symmetrical or skewed. Positive skewness means that the tail of the distribution has been stretched out to the right; negative skewness, to the left. The degree of peakedness of a distribution is called kurtosis.

SOME DESCRIPTIVE STATISTICS

1. Central Tendency Measures

A set of values of a variable may be analyzed to determine the central tendency of that variable. This analysis may be made by computing such statistics from a set (x_1, x_2, \dots, x_N) as:

- a. The mean (\bar{x})

$$\bar{x} = \frac{\sum x_i}{N} \quad \text{where } N = \text{number of cases.}$$

- b. The median (M)

M = value of the numerically ordered variable which gives an equal number of cases above and below it. (In case N is even, M is the average of the two middle values.)

- c. The mode (m)

m = value of the variable which occurs most frequently.

- d. Geometric mean (G)

$$G = (x_1 \cdot x_2 \cdot x_3 \cdots x_N)^{1/N}$$

(used in rate-of-growth problems to give an average power of change)

- e. Harmonic mean (H)

$$H = \frac{N}{\sum \frac{1}{x_i}}$$

(used in computing a mean where x = a rate, such as miles/hr., \$/gal., etc.)

2. Dispersion Measures

Statistics used to measure dispersion or variability of a variable are:

- a. The range (R)

$$R = x_{\max} - x_{\min}$$

- b. The variance (σ^2)

$$\sigma^2 = \frac{\sum (x_i - \bar{x})^2}{N}$$

$$= \frac{\sum x_i^2}{N} - (\bar{x})^2$$

- c. The standard deviation (σ)

$$\sigma = \sqrt{\text{variance}}$$

- d. The average deviation (A. D.) or mean absolute deviation

$$\text{A. D.} = \frac{\sum |x_i - \bar{x}|}{N}$$

3. Skewness Measures

Some measures of skewness or a symmetry are:

- a. Pearson's 1st coefficient (S1)

$$S1 = \frac{\bar{x} - m}{\sigma}$$

- b. Pearson's 2nd coefficient (S2)

$$S2 = \frac{3(\bar{x} - M)}{\sigma}$$

- c. Third moment measure - relative to

$$\sigma^3 (a_3)$$

$$a_3 = \frac{\sum (x_i - \bar{x})^3 / N}{\sigma^3}$$

A positive value of these coefficients indicates positive skewness.

4. Kurtosis Measures

A common measure of kurtosis is the fourth moment - relative to $\sigma^4 (a_4)$.

$$a_4 = \frac{\sum (x_i - \bar{x})^4 / N}{\sigma^4}$$

Kurtosis is viewed with respect to the normal, or bell-shaped, distribution (see below), for which $a_4 = 3$.

PROBABILITY

1. Concepts

The probability of an event taking place is practically defined as the relative frequency of occurrence of that event -- for example, $P(E) = .7$. By definition, $0 \leq P(E) \leq 1.0$. The conditional probability of E_2 on E_1 is the probability of E_2 occurring, given that E_1 has already happened; this is written as $P(E_2 | E_1)$. If $P(E_2) = P(E_2 | E_1)$, the two events are said to be independent. Compound probability is the probability that a combination of two or more events will occur -- for example, $P(E_1 E_2)$. Computationally, $P(E_1 E_2) = P(E_1) \cdot P(E_2 | E_1)$, or $P(E_1) \cdot P(E_2)$ if E_1 and E_2 are independent. The probability of one or another event occurring is given by

$$P(E_1 \text{ or } E_2) = P(E_1) + P(E_2) - P(E_1 E_2).$$

If the two events E_1 and E_2 are mutually exclusive, this reduces to $P(E_1) + P(E_2)$.

2. Combinatorial Analysis

The number of permutations (differently ordered arrangements) of n items taken r at a time is

$$P(n, r) = \frac{n!}{(n-r)!}$$

Section	Area	Page
05	01	03

The number of combinations (different sets, regardless of order) of n items taken r at a time

$$C(n, r) = \frac{n!}{(n-r)! r!}$$

Note that always $C(n, r) \leq P(n, r)$

The number of unique arrangements of n items of which n_1 are alike, n_2 are alike, . . . n_k are alike is

$$A(n, n_1, n_2, \dots, n_k) = \frac{n!}{n_1! n_2! \dots n_k!}$$

These formulas are useful in determining the total number of possible outcomes of a given set of conditions so that the probability of the occurrence of a certain event can be computed.

3. Distributions for Discrete Variables

- a. Binomial distribution. -- If the probability that an event will occur on a trial is p , and n independent trials are made, then the probability that the event will occur x times in n trials is

$$P(x; n, p) = C(n, x) p^x (1-p)^{n-x}$$

$$= \frac{n!}{(n-x)! x!} p^x (1-p)^{n-x}$$

This is sometimes called the Bernoulli distribution.

- b. Multinomial distribution. -- If the probabilities that events E_1, E_2, \dots, E_k will occur on any trial are p_1, p_2, \dots, p_k respectively, then the probability that in n independent trials there will be x_1 occurrences of E_1, x_2 , of E_2 , etc., is

$$P(x_1, x_2, \dots, x_k; n, p_1, p_2, \dots, p_k)$$

$$= \frac{n!}{x_1! x_2! \dots x_k!} p_1^{x_1} p_2^{x_2} \dots p_k^{x_k}$$

- c. Poisson distribution. -- The Poisson can be derived from the binomial distribution for which $p \rightarrow 0$ and $n \rightarrow \infty$ so that $mp = \lambda$. The probability that x events take place during some period (in which the average number of events is λ) is given by

$$P(x; \lambda) = \frac{\lambda^x e^{-\lambda}}{x!}$$

- d. Hypergeometric distribution. -- This distribution applies in small population problems where the probability that a sample of N items from a lot of size s will contain x items with a certain characteristic, given that there are m such items in the population, is

$$P(x; N, s, m) = \frac{C(m, x) \cdot C(s-m, N-x)}{C(s, N)}$$

4. Distributions for Continuous Variables

Continuous distributions do not permit probabilities to be associated with specific values of a variable, since degree of precision in measurement is unlimited; rather, one must speak of the probability that a value lies between two limits. In general, this gives

$$P \left\{ a < x < b \right\} = \int_a^b f(x) dx$$

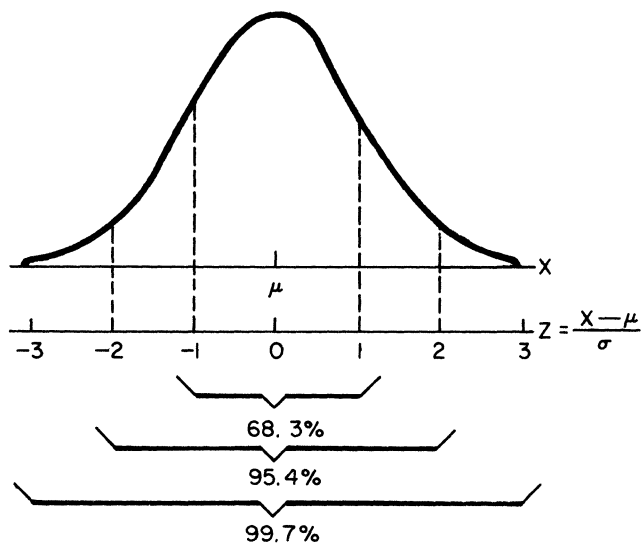
where $f(x)$ is the height of the probability curve for any x and is called the frequency function or density function. Some important continuous distributions are discussed below.

- a. Normal distribution. -- The normal distribution can frequently be used to describe errors, physical properties, or other phenomena which are symmetrically distributed about some average value. The theoretical density function of the normal is given by

$$f(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-1/2 \left(\frac{x-\mu}{\sigma} \right)^2}$$

where the two parameters μ and σ are the population mean and standard deviation.

When the substitution $Z = \frac{x-\mu}{\sigma}$ is made, the density function can be expressed in normalized form in which Z is a new variable which is normally distributed with mean of 0 and variance of 1.0. This standardization of any normal distribution is a valuable aid, since tables of Z and associated areas under the standard curve can be consulted for probability calculations irrespective of μ and σ .



In reality, few variables are normally distributed, but agreement in many cases is close enough to permit acceptable approximation of real-world data by the theoretical normal curve.

- b. Chi-square distribution. -- Some very important statistics follow the theoretical chi-square (χ^2) distribution. For example, if independent samples each of size N are taken from a normally distributed population with mean μ and standard deviation σ , then the statistic

$$Y = \frac{\sum (x_i - \bar{x})^2}{\sigma^2}$$

can be shown to be distributed as χ^2 with $N-1$ degrees of freedom. In general, the degree of freedom of a statistic is the sample size less the number of parameters which must be estimated for the statistic -- for example, \bar{x} in the above formula for Y . The density function of the χ^2 is

$$f(\chi^2; \nu) = K(\nu) \cdot \chi^{\frac{\nu}{2} - 1} e^{-\chi^2/2}$$

where ν = degrees of freedom which must be an integer, and $K(\nu)$ is a function of the degrees of freedom.

- c. Student's t distribution. -- The t distribution is used in small sample problems. Many useful statistics follow the

t distribution, and therefore inferences can be made by consulting the proper theoretical t curve that applies. Like the χ^2 , the t has one parameter, ν , which is its degrees of freedom. The frequency function for any one value of ν is

$$f(t; \nu) = K(\nu) \left[1 + t^2/\nu \right]^{-\frac{\nu+1}{2}}$$

where $K(\nu)$ is a constant for a given ν . The theoretical t is generated by the ratio of a normally distributed variate to one which has a χ^2 distribution (Z and χ^2 independent), namely

$$t = \frac{Z}{\sqrt{\chi^2/\nu}}$$

- d. The F distribution. -- The F distribution is generated by the ratio of two independent χ^2 variables divided by their degrees of freedom.

$$F = \frac{\chi_1^2/\nu_1}{\chi_2^2/\nu_2}$$

The density function of this two-parameter distribution is

$$f(F; m, n) = K(m, n) \frac{\left(\frac{m}{n}\right)^{m/2} F^{(m/2)-1}}{\left(1 + \frac{m}{n} F\right)^{\frac{m+n}{2}}}$$

The F distribution is mainly used in the analysis-of-variance tests which are discussed below to determine whether two χ^2 statistics are significantly different from each other.

SAMPLING THEORY AND ESTIMATION

Sampling is used for both descriptive and inferential purposes. A sample of size N which is drawn from a large population with a certain mean (μ) and standard deviation (σ) may be analyzed to infer the values of μ and σ . The statistics \bar{x} and s are unbiased estimates of these parameters. Unbiased means that the expected value of the estimator is the true value of the parameter.

Section	Area	Page
05	01	05

$$\bar{x} = \frac{\sum x_i}{N} \quad (\text{estimates } \mu)$$

$$s = \sqrt{\frac{\sum (x_i - \bar{x})^2}{N-1}} \quad (\text{estimates } \sigma)$$

Note that when estimating σ by s , a divisor of $N-1$ is used rather than N .

There are two types of estimation: point and interval. A point estimate is a single value, while an interval estimate will specify a range for a parameter with a certain probability that the true value of the population parameter will fall within that range. Interval estimates are generally desired since they indicate precision or reliability. Every sampling statistic will have a mean and a variance. Smaller variances give rise to more precision in the determination of the true value of the parameter which is being estimated.

For example, if a number of samples of size N are drawn as in the case above, the sampling distribution of the sample means (\bar{x} 's) can be expected to have mean μ and standard deviation σ/\sqrt{N} (called the standard error of estimate). The larger the N , the smaller the standard error for any distribution with finite σ . This ability to decrease error by increasing N is called the Law of Large Numbers. Furthermore, irrespective of the original distribution of x 's, the \bar{x} 's will approach a normal distribution as $N \rightarrow \infty$. Confidence limits on \bar{x} can be set by adding and subtracting multiples of the standard error of estimate and referencing the normal probability tables. For example, $\bar{x} \pm 1$ standard error gives 68% probability of containing the true μ ; ± 2 standard errors gives 95%; etc. An assumption of normality of the statistic \bar{x} is predicated upon knowledge of σ , but if σ must be estimated by s (as is usually the case), normality can be assumed if $N > 30$. If N is small, the t distribution with degrees of freedom = $N-1$ applies. In planning experiments, the sample size that is necessary to give desired confidence limits may be calculated by reversing the above procedure.

The sampling distribution of the sum or difference of two variables (each having its own distribution) will have the properties

$$\mu_{x \pm y} = \mu_x \pm \mu_y$$

$$\sigma_{x \pm y}^2 = \sigma_x^2 + \sigma_y^2$$

if the variables are independent of each other. Note that variances are summed in either case.

The sampling distribution of sums or differences of means of two normal populations can be expected to take the form of the normal distribution with

$$\begin{aligned} \mu_{\bar{x} \pm \bar{y}} &= \mu_x \pm \mu_y \\ \sigma_{\bar{x} \pm \bar{y}}^2 &= \sigma_{\bar{x}}^2 + \sigma_{\bar{y}}^2 = \frac{\sigma_x^2}{N_x} + \frac{\sigma_y^2}{N_y} \end{aligned}$$

To test the hypothesis of the difference of two means, the standardized Z variate can be set up as

$$\begin{aligned} Z &= \frac{(\bar{x} - \bar{y}) - (\mu_x - \mu_y)}{\sigma_{\bar{x} - \bar{y}}} \\ &= \frac{\text{actual difference} - \text{hypothesized difference}}{\text{standard deviation of difference}} \end{aligned}$$

The normal tables are then referenced to find the probability that such a discrepancy or a larger one could occur and the hypothesis still be true. A small probability (say $\leq .05$) may be taken to mean that the hypothesis should be rejected. This probability is called the α risk, and is the chance of rejecting a true hypothesis.

The test of difference between two means, when the population σ 's are unknown but are assumed equal, requires computation of the t statistic. Assuming the hypothesized difference is 0, then

$$t = \frac{\bar{x} - \bar{y}}{\sqrt{s^2 \cdot \left(\frac{1}{N_x} + \frac{1}{N_y} \right)}}$$

where

$$s^2 = \frac{\sum (x_i - \bar{x})^2 + \sum (y_i - \bar{y})^2}{N_x + N_y - 2}$$

Reference the t tables with degrees of freedom

$$N_x + N_y - 2.$$

A test for equal means can be made when the σ 's are unknown but presumed equal, and the sample sizes are large:

$$t = \frac{\bar{x} - \bar{y}}{\sqrt{\frac{s_x^2}{N_x} + \frac{s_y^2}{N_y}}}$$

With large N 's, the normal table may be referenced to obtain the probability of the existence of such a difference if the true means are really 0. (t tends to be distributed normally as N 's become large.)

Just as tests regarding means can be made and confidence limits set, similar tests can be applied to variances. These tests will reference the χ^2 distribution to determine confidence limits of the true variance, σ^2 , as based on the s^2 computed from the sample. The 95% confidence limits on σ^2 from a sample of size N are

$$\frac{(N-1)s^2}{\chi_{.975}^2} < \sigma^2 < \frac{(N-1)s^2}{\chi_{.025}^2}$$

where χ_b^2 represents the tabular value of χ^2 (for N-1 degrees of freedom), such that b (in %) of the curve lies between 0 and χ_b^2 .

CORRELATION

Two-variable correlation techniques are profitably used to indicate the numerical relationship between sets of measurements. The most common measure is the product moment correlation coefficient, r. Assuming a sample of size N, each case of which has scores on two variables, then

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{N \sigma_x \sigma_y}$$

$$= \frac{N \sum x_i y_i - (\sum x_i)(\sum y_i)}{\sqrt{[N \sum x_i^2 - (\sum x_i)^2][N \sum y_i^2 - (\sum y_i)^2]}}$$

By nature of its definition, $-1 \leq r \leq 1$. If $r = 0$, then knowledge of one variable contributes nothing to estimating the other. An r of +1 or -1 indicates perfect linear relationship (direct or inverse, respectively). Actually r^2 is more meaningful than r since it indicates the average percentage of variance of one variable which is removed by knowledge of the other. Note that $0 \leq r^2 \leq 1$.

This technique of correlation has a certain relationship with a line of fit which has the equation $y = a + bx$, where

$$b = \frac{N \sum x_i y_i - (\sum x_i)(\sum y_i)}{N \sum x_i^2 - (\sum x_i)^2}$$

$$a = \frac{\sum y_i - b \sum x_i}{N} = \bar{y} - b\bar{x}$$

The variables in the formulas must be reversed if it is desired to predict x as a function of y. The same r, of course, will apply.

The standard deviation of y for a given x is

$$\sigma_{y \cdot x} = \sigma_y \sqrt{1 - r^2}$$

However, when generalizing from a sample to a population, the standard error of estimate of y on x is computed by

$$s_{y \cdot x} = s_y \sqrt{\left(\frac{N-1}{N-2}\right) (1 - r^2)}$$

$$= \sqrt{\frac{\sum (y_i - \bar{y})^2}{N-2} (1 - r^2)}$$

The correction of $\left(\frac{N-1}{N-2}\right)^{1/2}$ adjusts for the fact that two degrees of freedom are lost in estimating the coefficients a and b in the equation. The effect is negligible unless N is small.

MULTIPLE REGRESSION

Multiple regression is an extension of two-variable correlation in which a dependent variable, Y, is to be expressed as a function of several independent variables (x_i 's). The linear equation is:

$$Y = a + b_1 x_1 + b_2 x_2 + \dots + b_k x_k + \epsilon$$

or

$$Y = Y_{\text{est}} + \epsilon$$

in which the a and b_i 's are to be determined. The criterion for "best fit" which is used is called least squares, and involves selecting the coefficients so that the sum of the squared errors between each actual Y and its least squares estimate, Y_{est} , are minimized. Mathematically this is

$$\min \left[\sum \epsilon^2 \right] = \min \left[\sum (Y - Y_{\text{est}})^2 \right]$$

$$= \min \left[\sum (Y - a - b_1 x_1 - \dots - b_k x_k)^2 \right]$$

This minimization is accomplished by taking the partial derivative of $\sum \epsilon^2$ with respect to each coefficient and solving a set of simultaneous equations in k + 1 unknowns. These equations are called normal equations and appear as follows:

$$1) \quad aN + b_1 \sum x_1 + b_2 \sum x_2 + \dots + b_k \sum x_k = \sum Y$$

$$2) \quad a \sum x_1 + b_1 \sum x_1^2 + b_2 \sum x_2 x_1 + \dots + b_k \sum x_k x_1 = \sum x_1 Y$$

$$\begin{aligned}
 3) \quad & a \sum x_2 + b_1 \sum x_1 x_2 + b_2 \sum x_2^2 + \dots + b_k \sum x_k x_2 = \sum x_2 Y \\
 & \cdot \\
 & \cdot \\
 & \cdot \\
 & k+1) a \sum x_k + b_1 \sum x_1 x_k + b_2 \sum x_2 x_k + \dots + b_k \sum x_k^2 = \sum x_k Y
 \end{aligned}$$

A matrix inversion of the coefficients of the unknowns is then applied along with a matrix-vector multiplication to obtain the solution (see matrix section of this Handbook).

The standard error of estimate of Y on the x's is given by

$$s_y \cdot x_1 x_2 \dots x_k = \sqrt{\frac{\sum (Y - Y_{est})^2}{N - (k + 1)}}$$

for the population which the sample represents.

Recalling that the square of a correlation coefficient indicates the proportion of variation of one variable accounted for by knowledge of another variable, this can be generalized to a multiple correlation coefficient, R, so that

$$R^2 = \frac{\text{explained variation}}{\text{total variation}} = \frac{\sum (Y_{est} - \bar{Y})^2}{\sum (Y - \bar{Y})^2}$$

which reduces to

$$R = \sqrt{1 - \frac{s_y^2 \cdot x_1 x_2 \dots x_k}{s_y^2}}$$

Rather than use different independent variables (x_i 's), one may wish to investigate how well Y can be expressed in terms of powers of one variable. This is called polynomial curve fitting, and gives, for example,

$$Y = a + bx + cx^2 + dx^3 \quad (\text{a cubic equation})$$

Here, the higher powers of x may be treated as additional variables so that the preceding discussion on the normal equations applies. Each new term in x adds an additional bend to the curve -- for example, first term, straight line; second, parabola; third, cubic with two bends; etc.

Other possible equations to be considered in curve fitting are

- 1) $Y = ab^x$ (exponential)
- 2) $Y = ax^b$ (geometric)

Most multiple regression computer programs permit such transformations and others to be made to variables. There are some stepwise multiple regression programs which analyze sets of data so as to form an equation which uses only those independent variables which contribute more than a specified minimum amount toward reducing the unexplainable variation of the dependent variable. These "useful" variables are brought in one at a time in order of decreasing significance.

ANALYSIS OF VARIANCE

Analysis of variance is a statistical technique for determining whether certain factors produce a significant change in some criterion measure (for example, of a system's performance). Many of the concepts in analysis of variance were formulated in agricultural experimentation. A typical application is one in which the effect of various plant foods on rate of growth of certain vegetation is to be evaluated. Assuming that rate-of-growth data is collected over a period of time and summarized so that one value (say, average rate of growth) is entered in a results table for each of the r plants treated by each of c food preparations, we have

		Food Treatment			
		1	2	c
Plants	1	x_{11}	x_{12}		x_{1c}
	2	x_{21}			
	.				
	.		$x_{ij} =$	Avg. rate of growth	
	r	x_{r1}	x_{r2}		x_{rc}

There will be an overall average rate of growth, \bar{x} , and an average within each food treatment, \bar{x}_j ($j = 1, 2, \dots, c$). The total sum of squares, $\sum \sum (x_{ij} - \bar{x})^2$, can be partitioned into its two components; the sum of squares between food treatment means, $\sum \sum (\bar{x}_j - \bar{x})^2$; and the sum of squares within treatments, which is called the residual sum of squares, $\sum \sum (x_{ij} - \bar{x}_j)^2$.

Recalling that $\sigma_{\bar{x}}^2 = \sigma_x^2/N$ (from section V) where N is the number in the sample, then an estimate of σ_x^2 is given by $N \sigma_{\bar{x}}^2$. Using $s_{\bar{x}}^2$ as an unbiased estimate of $\sigma_{\bar{x}}^2$, we have one estimate of σ_x^2 as

$$s_1^2 = r s_{\bar{x}}^2 = r \frac{\sum (\bar{x}_j - \bar{x})^2}{c-1}$$

The assumption of an equal number of cases per column is made for simplicity but is not necessary. The other unbiased estimate of σ_x^2 is given by the weighted average of the s^2 calculations within columns or

$$s_2^2 = \frac{(r-1)(s_1^2 + s_2^2 + \dots + s_c^2)}{c(r-1)} = \frac{\sum \sum (x_{ij} - \bar{x}_j)^2}{c(r-1)}$$

These two estimates of σ_x^2 can be shown to be independent. Since, in general,

$$\frac{(N-1)s^2}{\sigma^2}$$

has a χ^2 distribution with $N-1$ degrees of freedom (section IV. D. 2), then s^2/σ^2 is distributed as $\chi^2/(N-1)$. Hypothesizing that the two independent estimates, s_1^2 and s_2^2 both represent the same σ^2 , we have

$$\frac{s_1^2/\sigma^2}{s_2^2/\sigma^2} = \frac{s_1^2}{s_2^2} = \frac{r \sum (\bar{x}_j - \bar{x})^2}{\sum \sum (x_{ij} - \bar{x}_j)^2} \cdot \frac{c(r-1)}{c-1}$$

This new statistic, which is a ratio of two independent χ^2 's divided by their degrees of freedom, is distributed as the F statistic with degrees of freedom $c-1$ and $c(r-1)$. Standard F tables are consulted at a given α risk to determine the probability that such a value of the statistic could be computed and the hypothesis still be true. Thus, the name "analysis of variance" is given to this technique which truly compares variances in order to draw conclusions regarding average effects of variables on some performance measure.

It is important to note that the analysis-of-variance technique has very many variations. It is possible, using a different model, to test not only for column effects as above but also for row factors. For example, each row in the sample problem might have represented a different environmental humidity, thus enabling a test of this variable on plant growth rate to be made. As two or more factors are introduced, additional conclusions regarding interactions may be made. An interaction is the phenomenon in which the effect of a row treatment changes across columns, or vice versa. Sometimes the detection of these interactions is as important as the ascertainment of the main treatment effects themselves.

In general, recognition of factors which reduce the residual variance estimate against which the F test is made tends to increase the sensitivity of the analysis. Caution must be exercised to select the proper statistical test to match the experimental design. In fact, it is necessary to outline the tests and statistical computations when the experiment is in its formative stages in order to avoid later problems of analysis.

BIBLIOGRAPHY

Duncan, A. J., Quality Control and Industrial Statistics, Richard D. Irwin, Inc., Homewood, Illinois, 2nd Edition, 1959.

Feller, W., An Introduction to Probability Theory and Its Application, Vol. I., John Wiley & Sons, Inc., New York, 1957.

Mode, E. B., Elements of Statistics, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 3rd Edition, 1961.

Mosteller, F., R. E. K. Rourke, and G. B. Thomas, Jr., Probability with Statistical Applications, Addison-Wesley Publishing Co., Inc., Reading, Massachusetts, 1961.

Spiegel, M. R., Theory and Problems of Statistics, Schaum Publishing Co., New York, 1961.

Walker, H. M., and J. Lev, Statistical Inference, Henry Holt and Co., New York, 1953.

VALIDATION AND CONTROLS

Validation is the process of checking data to establish the accuracy of its coding and content. The validation function may consist of edit routines in the main processing run or separate edit passes preceding the main processing run.

Controls refer to the control of data as well as systems controls that detect operator errors, machine malfunction and errors not detected by the normal editing function.

REASONS FOR VALIDATION AND CONTROL

- To establish clear and systematic error detection and correction procedures.
- To satisfy audit requirements.
- To realize maximum productivity from the data processing system.
- To minimize rerun time.

COSTS

The costs of incorporating satisfactory data validation and control procedures into a system should be investigated closely in order that an intelligent choice can be made as to the number of such procedures that should be used. Cost areas that should be considered are:

- Programming time required to write edit routines and to incorporate adequate controls in all programs.
- Testing and debugging of such routines.
- Data preparation for testing and debugging.
- Possible increase in memory capacity in order to accommodate such routines.
- Decrease in the "real" throughput of the system caused by extra passes or processing for the edit function and systems checks.
- Just as important, the costs of not putting in proper checks, should also be analyzed, since failure to do so could lead to incorrect results and excessive interruptions and corrections.

LEVELS

In the following discussion, validation and controls will be considered at three levels: departmental, operation-to-operation and run control.

Departmental Control

Control procedures are required between the data processing department and user departments to ensure that the input data is correct and complete, that schedules are adhered to and that output is distributed on time and to the required destination(s).

By the User Department

A procedure manual, containing detailed descriptions of data preparation and disposition, should be maintained. In addition, a control sheet should be prepared and sent to the data processing department with the data. This control sheet should contain:

- Job number or data description
- Date
- Source of input (user department)
- Batch number
- Time when output is required
- Item counts and control totals
- Disposition of input and output
- Special processing instructions

By the Data Processing Department

A procedure manual, containing detailed data descriptions, estimates of processing time, error procedures and other pertinent information, should be maintained.

The control sheet from the user department should be checked for accuracy against the accompanying data, and a separate record maintained of such information as date, time, etc.

Balance figures should be independently generated and balanced against those recorded on the control sheet.

Control card(s) should be prepared to implement special processing instructions.

Operation-to-Operation Control

Controls are required between successive operations within the data processing department, including manual and/or unit record and/or computer operations.

- Balance totals must be maintained and checked between processing runs to ensure processing accuracy.

- Job documentation should be clear and complete. It should include instructions to the operator for error and out-of-balance conditions.
- Operators should be instructed in the use of an operating log.
- Systematic procedures should be followed when modifying programs previously released for production.
- Adequate procedures must be established to govern issuing of programs, master files, etc., from the installation library.

Run Control

This refers to validation and control procedures that can be incorporated into individual processing passes. In this section, data validation and systems controls will be discussed under separate headings.

Data Validation

1. Character checking -- for example, blank, signed, numerical, alphabetic.
2. Field checking:
 - a. Sequence check on control fields.
 - b. Consistency check -- to check two or more data fields (from one transaction) in relation to each other to ensure consistency of the data.
 - c. Limit check -- apply upper and lower limit checks on the value of a field.
 - d. Code validity -- by means of a table lookup procedure, the validity of a code can be determined.
 - e. Completeness check -- to check for the existence and correct sequence of all parts of a multi-record transaction.
 - f. Date check -- to ensure that the input record date is acceptable.
 - g. Self-checking number -- in an application where identification codes such as item number or policy number are used, a self-checking digit should be appended to the field when the record is originally created

When this record is later processed, the self-checking digit should be regenerated and compared against the original to detect keypunch or transmission errors.

Two methods of generating a self-checking digit, modulus 10 and modulus 11, are described in the following bulletins:

- Self-Checking Number Feature, Modulus 11, and Its Associated Self-Checking-Number Generator, Modulus 11 (G24-1022).
- Self-Checking Number Feature (G24-1057).

3. Batch or level checking. Batching is defined as the assembly of a data file into subgroups (or batches) in order that a rerun caused by errors can be restricted to the rerun of a single subgroup instead of the entire file. Each batch should be preceded by a control card containing batch number, record count, control totals or hash totals and other pertinent information.

- a. Batch number check. -- A check should be made to ensure that detail records contain the same batch number as that in the control card.
- b. Batch record count. -- The total number of records in a batch should be balanced against the control card record count.
- c. Batch control totals. -- Control totals should be balanced to those in the control card.
- d. Batch hash totals. -- Hash totals are useful in cases where control totals, as discussed above, are unavailable or not readily obtainable. By subtracting the accumulated hash total from the predetermined (control card) hash total, one can determine whether a record has been inadvertently added or deleted, and also the contents of its hash total field.

4. Rejected data. In some applications, it is possible to process records even though they contain erroneous data. Follow-up procedures are usually required for the techniques discussed below:

- a. Approximation. In applications such as utility billing, the omission or unreasonableness of input data can be corrected by substituting a minimum charge for the erroneous data.
- b. Invalidating part of a record. This technique invalidates the erroneous data by substituting an error code to prevent further processing.
- c. Unscrambling. This technique is frequently used on data received from a data transmission device. Checks that can be made are:
 - Shift record left or right one or more positions to correct for possible offset of the record.
 - Scan a record in a backward direction. This technique is useful when input is created by a serial conversion process such as card to paper tape.
- d. Omission of a record. Omit a record or block of records that cannot be processed (for example, permanent read fail on a tape drive) and maintain a register of such omissions for subsequent follow-up.

- e. Absolute rejection. When the input data contains a multitude of errors, remove the run from the system and return the data to its source for correction.

Systems Controls

Systems controls include: tape and random access systems, card systems, general techniques and specific techniques that apply only to scientific computing. The reader should not construe from this that the techniques discussed earlier are unique to commercial data processing. On the contrary, most of these techniques are similarly applicable to scientific data processing.

1. Magnetic tape and random access systems
 - a. Header label. Each tape file in a system should have a header label as the first record on each reel of a file. This record should conform to IBM tape standards, both in content and format.

A programmed check of the header label will ensure that the correct data file has been loaded on the correct tape drive and that cycled files (grandfather, father, son concept) have been correctly drawn from the tape library.
 - b. Trailer label. Each tape file should have a trailer label as the last record on each reel of the file. The content and format of this record should follow appropriate tape standards.

A programmed check of trailer label counts against counts of reads, writes, adds and deletes will ensure that records have not been inadvertently added or deleted.
 - c. Redundancy tests and recovery. Each tape read or write routine should include a test for redundancy or parity errors. If an error occurs, recommended procedures should be followed. (See IBM tape standards.)
 - d. Noise record bypass. Noise records are extraneous characters found within the interrecord gap or sometimes appended to a valid record. To bypass noise records:
 - Design each record so that its length is greater than maximum noise record length.
 - Incorporate a programmed check to detect noise records as short records. (See IBM tape standards.)

- e. Record length. For systems that do not have an automatic (hardware) record-length check, a programmed check should be made.
2. Card systems
 - a. Card file identification. Data files should be identified with positive coding.
 - b. Program load check. To ensure the completeness of a program deck, a hash total check should be made during program loading.
 - c. Card count. A card count of input records should be maintained for balancing purposes.
 - d. Control totals. Control totals should be maintained and balanced.
3. General techniques
 - a. Checkpoint and restart. In processing runs, checkpoint and restart procedures should be preplanned in writing the program to avoid a complete rerun when a partial rerun would suffice.

At suitable points in the processing run, controls are set up and a checkpoint record is written (tape system only) in order that, in case of an error, the run can be restarted at the most recent checkpoint. With a card system, manual reorganization of a file and a punchout of checkpoint control fields is required.
 - b. Processing log. If sufficient memory is available, all programs should include a routine for logging error conditions on an output device.
 - c. Operator messages. In situations where systems setup is extremely critical (as is often the case in real-time applications), the program should provide messages for the operator on such items as console setup, control cards, etc., and should then check that the operator has correctly executed these instructions.
4. Scientific computing

Discussion in this section is involved primarily with the accuracy of processing. Examples of specific techniques are cited but the user will frequently have to devise his own methods.

 - a. Truncation. In calculations employing series, the significance of successive terms must be evaluated by the program.
 - b. Round-off. The accuracy of the final result should be predetermined to establish to what degree of precision calculations should be carried. Note that this

may be impossible in a complex situation where the number of calculations is very large. In such cases, working backwards from the answer may be necessary to establish the accuracy of the final result. In matrix inversion, for example, reinvert the inverted matrix and compare this with the original matrix.

- c. Verification. In some cases, it is possible to verify the solution by substituting the solution into the original equations and checking to see whether the equations are still valid.

In linear programming problems and the solution of simultaneous equations, for example, it is possible to test the accuracy of the solution by substituting it into the original equations. The difference between the computed right-hand side and the original right-hand side is a measure of the error. Separate summation of the positive and negative terms on the left-hand side may be necessary. In this way, an indication of the errors due to differencing nearly equal numbers can be obtained. In matrix inversion problems, the inverse can be inverted and compared with the original or it can be multiplied by the original matrix and compared against the unit matrix.

- d. Redundant calculation. This technique compares the results of two solutions of a given problem where some parameter which influences the accuracy can be varied between the two solutions. These parameters can be the degree of precision, the use of different noise digits in normalizing operations, the use of rounding, the number of terms used in a series, the use of automatic round-up and round-down, and the variation of step size of the independent variable in stepwise solutions.

This technique is useful in iterative type problems. For example, in the solution of differential equations one can use different size increments or different degrees of precision in successive solutions of the same problem. The difference between the solutions then becomes a measure of the accuracy of the original solution.

- e. In-process reasonableness check. At certain points in the computation, it may be possible to test whether the data falls within a certain range or has a certain sign. Hash totals can also be used with standard sets of input data such as tables, etc.
- f. Convergent iteration procedures. In some instances it may be possible to use iterative procedures which are stable and self-correcting to improve the accuracy of results.

BIBLIOGRAPHY

Price Waterhouse & Company, The Auditor Encounters Electronic Data Processing, IBM Corporation, revised, 1961.

Grabbe, E. M., S. Ramo and D. E. Wooldridge, Handbook of Automation, Computation and Control, Volume 2, John Wiley & Sons, 1959.

Leeds, Herbert D. and Gerald M. Weinberg, Computer Programming Fundamentals, McGraw-Hill Book Company, 1961.

Life Office Management Association

1) EDP Controls - Automation Report No. 5, November, 1961.

2) EDP in Life Insurance -- Proceedings of Automation Forum, 1962.

QUEUING THEORY

FUNDAMENTALS

Queuing theory studies analytically how a system behaves under statistically varying arrival and servicing situations: what idleness the service facility experiences, or what congestion a long queue creates, or what the form of the output distribution is. It shows, for example, that Poisson arrivals (20 items per minute, on average, say) along one channel into a single Poisson service channel (servicing 28 items per minute, on average, say) under a first-come, first-served discipline, will produce a Poisson output of items with the same average rate of output (20 items per minute) as the original input!

If the arrival rate, a , is a great deal less than the service rate, b , the service facility stands idle much of the time (is little utilized) and the queue is of little or zero length. If a is only slightly under b , the queue is rather large and the service facility seldom idle (utilization high). If a is equal to or greater than b , the facility is always busy and the queue becomes infinitely long as time goes on. In the remainder of the paper, a is always taken less than b . The utilization coefficient, $r = a/b$, varies between 0 and 1, between a situation of no queue and minimum utilization of the facility and a situation of a long waiting line and maximum utilization of the facility. If there are N servers in a facility, each with average rate b , then $r = a/Nb$.

Other useful measures of behavior (figures of merit) of a system are:

- Lq , the average number of items in the queue.
- L , the average number of items in the system.
- Wq , the average waiting time spent in queue.
- W , the average time spent in system (queue and facility).
- $Pq (> t)$, the probability of waiting more than a given time, t , in the queue.
- \bar{N} , the average number of servers occupied.

Queuing theory seeks out mathematical formulas for these measures and for the distributions of which the measures are an average.

Lq and \bar{N} may be important measures from the point of view of the service facility, determining to a certain extent the space to be allotted to the queue and service facility, respectively. Wq and $Pq (> t)$ may be crucial from the customer's standpoint. Figure 1 presents the essentials of

the Poisson arrivals to the Poisson servicer (first-come, first-served) situation which was described previously as having a Poisson output. Figure 2 illustrates two system measures for Figure 1, L and Lq , as functions of $r = a/b$. Note how L and Lq both tend to infinity as $r = a/b$ approaches the value 1 (i. e., as a approaches b).

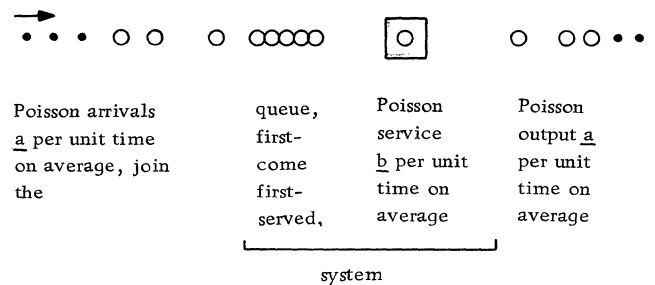


Figure 1. A simple Poisson system

L AND Lq

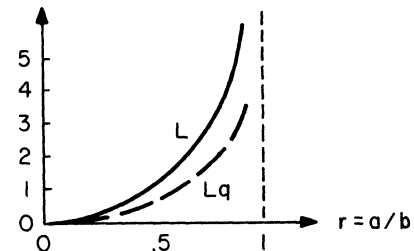


Figure 2. Average number of items in the system, L , and in the queue, Lq , as a function of utilization coefficient, $r = a/b$. The system is that given in Figure 1.

A word of caution is in order. Averages, a major output of applied queuing theory, are deceptive. One should, if possible, attach to the average some measure of scatter or standard deviation from the average. The length of the queue is 6 items \pm 2 items. The number in the system is $L \pm \Delta L$. Figure 3 illustrates the point for the Poisson system in Figure 1. It can be shown that $L = r / (1-r)$ and that $\Delta L = \sqrt{r} / (1-r)$. The standard deviation from the average is larger than the value of the average itself!

The theory is most successful where the Poisson process (e. g., Poisson arrival, servicing) operates, because the mathematical analyses are less complex. The Poisson process gives rise to the Poisson (random) probability distribution

L, AVERAGE NUMBER IN SYSTEM

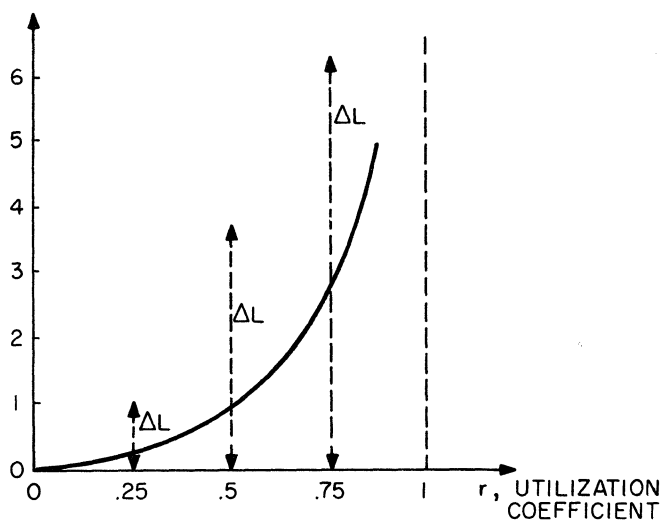


Figure 3. A plot of L vs. r for the Poisson system of Figure 1, showing how the standard deviation from the average, L , is always greater than L . At $r = .25$, $L = 1/3$ and $\Delta L = 2/3$. The negative ΔL is truncated by the horizontal axis (no negative queues).

covered earlier. The process is characterized by independent events in space or in time where short intervals between units cause congestion and long intervals between units create idleness. A Poisson process can be looked upon as a disintegrating scheduled or regular process. The conventional measures of behavior: L , L_q , W , W_q , $P_q (>t)$, ΔL , \bar{N} all increase, as a rule, as one changes from regular to Erlang to Poisson activity. If a complex system is not completely Poisson, but rather has a regular service facility, or regular arrivals here and there, then an idealized queuing model of the system which imposes the Poisson process everywhere yields measures or averages with the quality of conservativeness, for these latter are upper bounds on the Erlang (ranging from the almost regular to the almost random) and regular situation. This point is illustrated in Figure 4.

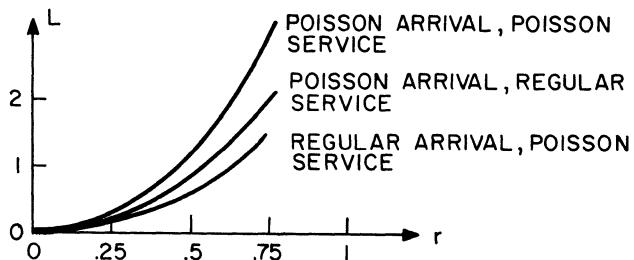


Figure 4. A plot of the average number in a system, L , vs. the utilization coefficient, r , showing how for a given value of r , the full Poisson system has bounded any system more regular, in the value of L .

Poisson (random), the Erlang, and the regular processes are graphed in Figure 5. Note how the probabilities of having 0, 1, 2, ... arrivals or servicings per unit time are distributed with maximum scatter for the Poisson, to zero for the regular.

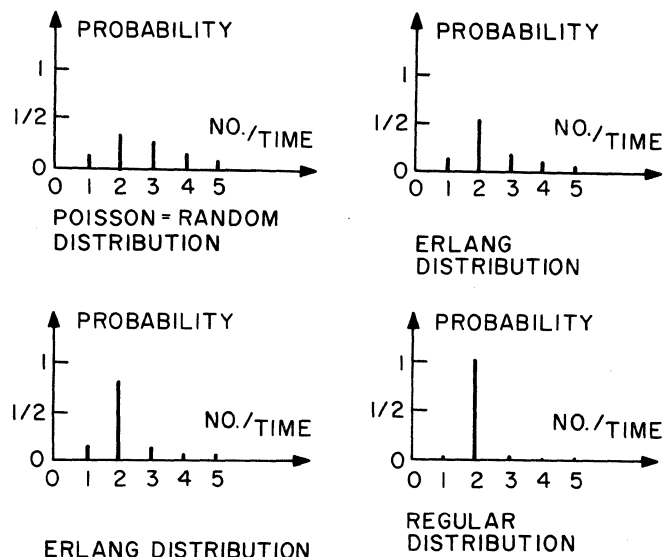


Figure 5. Probability of 0, 1, 2, 3, ... arrivals or servicings per unit time plotted vs. the number arriving or being serviced per unit time for Poisson, two Erlang and regular processes. The average arrival or service rate shown is 2. Probability = 0 is an impossible event; probability = 1 is a certain event.

APPLICATIONS

Queuing theory addresses itself to the problems of congestion and idleness. It has been applied to the analysis of telephone systems, (pioneer investigations by A. K. Erlang, "Use of Waiting-Line Theory in the Danish Telephone System", 1908), traffic light signals, landing of aircraft, railroad classification yards, loading ships, production and inventory control, scheduling patients at clinics, machine breakdowns, bridge and tunnel traffic, survival rates of raiding aircraft, etc.

A USEFUL SYSTEMS TOOL

There is a result in queuing theory due to J. R. Jackson, applicable to the most complex network of waiting lines and facilities, provided it is assumed that the network receives units or items, in Poisson fashion, from outside the network and that each facility (or node) operates in Poisson fashion upon these units, discharging them in Poisson "streams" towards other nodes or completely out

of the system. At first glance, this completely Poisson universe of nodes seems too restrictive for real-world systems and subsystems. Real-world situations involve Erlang or regular distributions, require limitations on queue space, operate on the units in batches, entail time-varying arrival rates, etc. However, if only rough measures are needed, as the systems analyst juggles the black boxes of a possible system configuration, then the Poisson measures of behavior produced by Jackson's theorem (L , L_q , W , W_q , etc.) may satisfy the need. Also, it is known that Poisson measures overestimate (on the safe side) less random measures, and that batching always increases the measures (the Jackson model would underestimate the averages of real-world batching). A Poisson model underestimates certain measures of behavior if it replaces a model in which queue size is finite and blocking exists.

- Jackson's result relates the state of an entire complex network of multiserver nodes with the separate states of each node.
- This, provided Poisson inputs and Poisson internal flows of similar units are assumed.
- Average arrival rates, \underline{a} ; percents of flow from each node to others; average service rates, \underline{b} ; and the number of service channels in a node, n , are required.
- Total average input rates, G , are calculated, and a check made to see that they are less than the total average nodal service rate nb .
- Several useful measures of behavior at each node are calculated with classical formulas, and from these, total system measures are obtained (for example, adding average waiting times, W , for a string or chain of nodes gives the average time to traverse the entire chain).

The following formulas hold for each node:

- I general or total average arrival rate, G
- II utilization coefficient, $r=G/bn$
- III probability of no units at the node (queue and service):

$$P_0 = \frac{1}{\left(\sum_{j=0}^{n-1} \frac{[nr]^j}{j!} \right) + \frac{[nr]^n}{n!(1-r)}}$$

- IV the average queue length: $L_q = P_0 r (nr)^n / n! (1-r)^2$
- V The average number at the node (both queues and service): $L = L_q + nr$

VI the average waiting time in the queue:

$$W_q = L_q / G$$

VII the average waiting time at the node (queue and service): $W = L/G$

VIII probability of waiting more than zero time:

$$P_q (> 0) = P_0 (nr)^n / n! (1-r)$$

IX probability of waiting more than a given time, t :

$$P_q (> t) = P (> 0) e^{-nbt(1-r)}$$

X probability that exactly ν service channels are busy out of n :

$$P_\nu = \frac{(nr)^\nu / \nu!}{\sum_{j=0}^n (nr)^j / j!}$$

XI probability of k units at the node:

$$P_k = P_0 (nr)^k / k!, \quad 0 \leq k \leq n$$

$$= P_0 (nr)^k / n! n^{k-n}, \quad k \geq n$$

XII probability of k or more units at the node:

$$P (> k) = n^{n-k} (nr)^{k+1} P_0 / (n-nr)n!, \quad k \geq n-1$$

BIBLIOGRAPHY

Flagle, C. D., W. H. Huggins, and R. H. Roy, Operations Research and Systems Engineering, Baltimore, The Johns Hopkins Press, 1960.

Goode, H. H., and R. E. Machol, System Engineering, New York, McGraw-Hill, 1957.

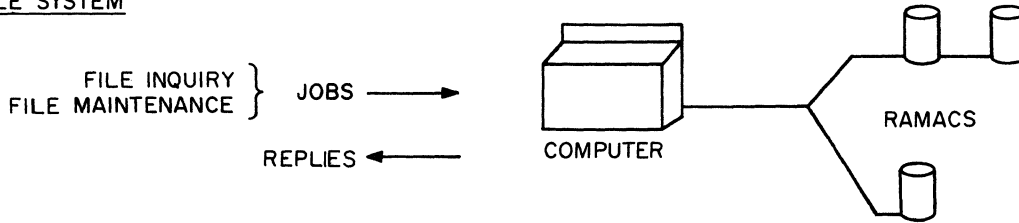
Morse, P. M., Queues, Inventories, and Maintenance, New York, John Wiley and Sons, 1958.

Saaty, T. L., Mathematical Methods of Operations Research, New York, McGraw-Hill, 1959.

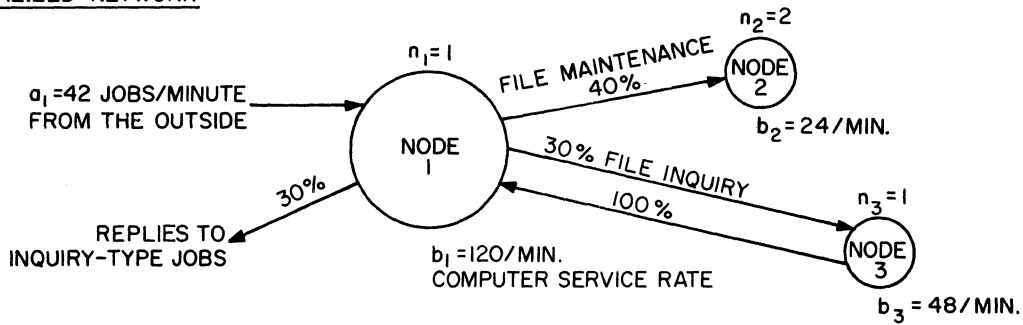
Saaty, T. L., Elements of Queuing Theory with Applications, New York, McGraw-Hill, 1961.

A simple numerical example will clarify the method and objective:

POSSIBLE SYSTEM



IDEALIZED NETWORK

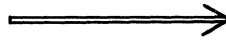


DETERMINATION OF G_1, G_2, G_3

$$G_1 = 42 + G_3 + 0G_2$$

$$G_2 = 4G_1 + 0G_3$$

$$G_3 = .3G_1 + 0G_2$$



$$G_1 = 60/\text{MIN.}$$

$$G_2 = 24/\text{MIN.}$$

$$G_3 = 18/\text{MIN.}$$

CHECKING FOR INFINITE QUEUES

$$r_1 = \frac{G_1}{n_1 b_1} = \frac{60}{1 \times 120} = 1/2 < 1$$

$$r_2 = \frac{G_2}{n_2 b_2} = \frac{24}{2 \times 24} = 1/2 < 1$$

$$r_3 = \frac{G_3}{n_3 b_3} = \frac{18}{1 \times 48} = 3/8 < 1$$

THEREFORE, THERE ARE NO INFINITE QUEUES BUILDING UP

SAMPLE MEASURES OF BEHAVIOR

$$\text{AVERAGE QUEUE IN FRONT OF COMPUTER} = r_1^2 / (1 - r_1) = .5 \text{ JOBS}$$

$$\text{AVERAGE WAIT IN FRONT OF NODE 3 (INQUIRY RAMAC)} = r_3^2 / (1 - r_3) G_3 = 1/80 \text{ MINUTE}$$

PICTORIAL GLOSSARY OF QUEUING THEORY

Units (Customers, items)



Service Facility



Service Facility with many channels (servers)

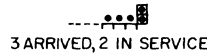


Queue (waiting line) with

- { No queue allowed
- { Fixed-length queue
- { Infinite queue (∞)



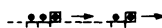
System (Queue + service facility + units in service and on queue)



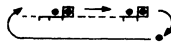
Double queues (a queue of arriving units is served by a queue of arriving servers)



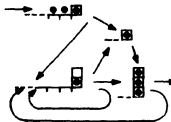
Tandem queues (two or more systems in series)



Cyclic queues (tandem queues with output units of last stage fed back to input queue of first stage)

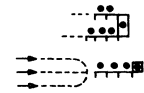


Network of queues (a system with many servers -- plus queues -- each with output going to other servers or to the outside)

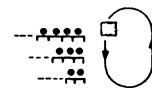


Multiqueue arrival

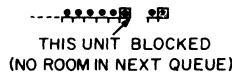
- { Separate queues
- { Consolidated queues



Polling (serving separate queues in cyclic order)

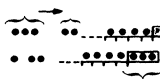


Blocking (hindering unit from advancing and facility from serving next unit)

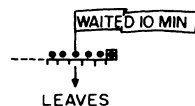


Batch (bulk) processes

- { Batch arrival
- { Batch service



Reneging (quitting queue after waiting a certain time)



Source of Arrivals

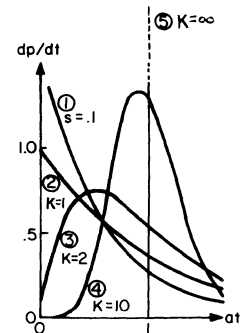
- { Limited (fixed No. which, when thru service can arrive again)
- { Unlimited (infinite No., potentially)

Queue Discipline (Rule for selecting next unit from queue)

- { First come, first served (head of queue)
- { Random (choose uniformly)
- { Priority (take unit of highest priority)
- { Pre-emptive priority (higher priority units interrupt service on lower; latter repeat or resume)

Time Distributions (for interarrival times or service times, t ; a , is average number per unit time)

- { Hyperexponential, $0 < s < .5$ ①
 $dp/dt = 2s^2ae^{-2sat} + 2(1-s)^2ae^{-2(1-s)at}$
- { Exponential ②
 $dp/dt = ae^{-at}$
- { Erlang-K Density ③; ④
 $dp/dt = (Ka)^K t^{K-1} e^{-Kat} / (K-1)!$
- { Regular, Constant Density ⑤
 $dp/dt = \infty$ spike at $at = 1$,
Area = 1



Number Distributions (for number arriving or being served per unit time, n ; a , is the average number per unit time)

- { Poisson (\equiv exponential), $K = 1$
- { Erlang-K, $K = K$
- { Regular, Constant, $K = \infty$

where the probability of n arrivals in a given time, t , is

$$P_n(t) \equiv e^{-Kat} \sum_{j=0}^{n-K} \frac{(Kat)^{j+nK}}{(j+nK)!}$$

Measures of Behavior (in steady state, $t \rightarrow \infty$)

- P_n \equiv Probability of n units in system
- L \equiv Average number in system
- L_q \equiv Average number in queue
- W \equiv Average waiting time in system
- W_q \equiv Average waiting time in queue
- $P(>t)$ \equiv Probability of spending time greater than t in system
- $P_q(>t)$ \equiv Probability of spending time greater than t in queue
- r \equiv Utilization coefficient (in simple systems = average arrival rate, a , divided by average service rate, b .)
- Δ \equiv Standard deviation or measure of scatter of a statistical quantity about its mean.

INDEX TO FORMULAS

SYMBOL KEY	
P	Poisson
E	Erlang - K
R	Random
H	Hyperexponential
G	General
Z	Zero Service Time
B	Binomial
F	First Come
PR	Priority
PP	Preemptive
RA	Random Selection
M	Miscellaneous

OTHER ASSUMPTIONS

QUEUE DISCIPLINE

ARRIVAL DISTRIB.

SERVICE DISTRIB.

SERVERS

QUEUES

SYSTEM

ONE SERVER

MANY SERVER

MANY QUEUE SYSTEM

CYCLIC

DOUBLE

TANDEM

NETWORK

POLLING

- Reneging
- Trimodal Service
- 2 Kinds Units
- K Kinds Units
- Queue Length Dependence
- Limited Queue Space
- Limited Number Sources
- Of Two Types, Same Service
- Of Two Types, Different Service
- Of Continuous Type
- Of R Types, Repeat
- Of 2 Types, Repeat
- Of 2 Types, Limited No. of One
- Reneging
- Non-Reneging
- K = 2
- K - General
- ∞ Queue Allowed
- ∞ Queue Allowed
- No Queue Allowed
- K = 2
- K - General
- K General, Batch Service
- ∞ Queue Allowed
- ∞ Queue Allowed
- No Queue Allowed
- ∞ Queue Allowed
- Of K Types
- Of Continuous Type
- Of 2 Types, Resume
- Arrivals Only at Discrete Times
- C Servers, Same Rates, Reneging
- C Servers, Different Rates
- C Servers, Limited Sources
- C Servers, Same Rates
- ∞ Servers
- ∞ Servers, Limited Sources
- Varying Number Servers, ≤ 2
- Varying Number Servers
- K = 2, 2 Servers
- C Servers
- 2 Stages
- K Stages
- Reneging
- N Stages, Poisson-Like Waiting
- 2 Stages, No Second Queue
- 2 Stages, ∞ Queues Allowed
- 2 Stages, Reneging
- 3 Stages, Reneging
- Any No. Nodes, Many Servers
- Serve All on Each Queue
- Serve Only One on Each Queue

There are Formulas for the Following Queue Characteristics

	P_n	L	L_q	W	W_q	$P_q(>t)$	$P(>t)$	M	
1	•								OR (57) 651
2	•								IBM Res. Report (62)
3	•								M129
4	•								OR (61) 324
5								•	
6	•							•	M18, SQ62, OR (59) 385
7	•								S354
8	•								M124
9	•								M132
10	•								S352
11	•								OR (60) 636
12	•							•	OR (58) 400, OR (59) 678
13	•							•	OR (61) 666
14	•							•	OR (57) 647
15	•								M119
16	•								M89
17	•								M87
18	•								S354
19	•								M96
20	•								M56
21	•								M81
22	•								M75
22A	•								JRSS, B, (54) 84, (55) 260
23	•								S347
24	•								M84
25	•								M55
26	•								S345
27	•								OR (60) 731
28	•								S352
29	•								OR (61) 738
30	•								OR (58) 104
31	•								OR (57) 651
32	•								OR (60) 508
33	•								S362
34	•								S360
35	•								S363
36	•								S364
37	•								OR (62) 219
38	•								OR (62) 222
39	•								M113
40	•								S361
41	•								OR (60) 248
42	•								SQ295
43	•								OR (61) 776
44	•								OR (58) 858
45	•								M100
46	•								S358
47	•								S358
48	•								OR Quarterly (54) 117
49	•								OR (57) 520
50	•								IBM JRD (61) 206
51	•								IBM JRD (62) 248

Legend
M = Morse (see Bibliography, page 03)
S = Saaty "Mathematical" (see Bibliography)
SQ = Saaty "Elements" (see Bibliography)
(The above codes are followed by page.)
OR = Operations Research Journal
JRSS = Journal of the Royal Statistical Society
IBM JRD = IBM Journal of Research and Development
(The above codes are followed by year and page.)

MATRIX METHODS

Disciplines such as economics and electrical engineering encompass problems which can often be expressed mathematically in terms of matrices. Other problems are stated in terms of algebraic or differential equations which may require matrix methods for solution. The basic facts of matrix algebra are presented along with a discussion of the two most commonly posed problems intimately associated with matrices: the solution of simultaneous linear algebraic equations and the finding of eigenvalues and eigenvectors. The presentation is neither complete nor mathematically rigorous. The enclosed bibliography should be consulted for detailed information.

MATRIX ALGEBRA

For our purposes, an $m \times n$ matrix is an array of complex numbers in m horizontal rows and n vertical columns. We shall use English capitals to denote matrices, and subscripted lower-case English letters to denote elements of matrices. Thus, $a_{i,j}$ may denote the element in the i^{th} row and j^{th} column in the matrix A .

$$A \equiv \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix} \equiv (a_{i,j})$$

A vector is an $m \times 1$ or a $1 \times n$ matrix; and often called a column or row vector, respectively. The quantity m (or n) is the dimension of the vector. A square matrix is an $m \times m$ matrix, and the number of rows or columns, m , is called its order. A diagonal matrix is a square matrix such that all of its elements $a_{i,j}$ are zero if $i \neq j$. Consequently,

$$\begin{pmatrix} 3 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & -7 \end{pmatrix}$$

is a diagonal matrix. A diagonal matrix such that $a_{i,i} = 1$ for $i = 1, 2, \dots, m$ is called a unit matrix of order m and is denoted by I . A zero or null matrix is a matrix all of whose elements are zero.

It may be denoted by θ . An upper triangular matrix is a square matrix such that all of its elements $a_{i,j}$ are zero if $i > j$. Thus,

$$\begin{pmatrix} 3 & 4 & 5 \\ 0 & 6 & 7 \\ 0 & 0 & 8 \end{pmatrix}$$

is an upper triangular matrix. A lower triangular matrix is similarly defined.

Sum of matrices: The sum $A+B$ of two $m \times n$ matrices A and B is an $m \times n$ matrix whose elements are the sums of the corresponding elements of A and B . Hence, $a_{i,j} + b_{i,j}$ is the element in the i^{th} row and j^{th} column of the sum matrix. For example:

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} + \begin{pmatrix} 1 & 1 & 1 \\ 2 & 2 & 2 \\ 3 & 3 & 3 \end{pmatrix} = \begin{pmatrix} 2 & 3 & 4 \\ 6 & 7 & 8 \\ 10 & 11 & 12 \end{pmatrix}$$

The operation of matrix summation is associative and commutative; that is, $A+(B+C)=(A+B)+C$ and $A+B=B+A$.

Product of matrices: If A is an $m \times n$ matrix and B is an $n \times p$ matrix, then the product of A and B , denoted AB , is defined as an $m \times p$ matrix such that the element in the i^{th} row and j^{th} column is

$$\sum_{k=1}^n a_{i,k} b_{k,j}$$

(the so-called inner product of the i^{th} row of A with the j^{th} column of B). For example:

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix} \begin{pmatrix} 9 & 8 \\ 7 & 6 \\ 5 & 4 \end{pmatrix} = \begin{pmatrix} 1 \cdot 9 + 2 \cdot 7 + 3 \cdot 5 & 1 \cdot 8 + 2 \cdot 6 + 3 \cdot 4 \\ 4 \cdot 9 + 5 \cdot 7 + 6 \cdot 5 & 4 \cdot 8 + 5 \cdot 6 + 6 \cdot 4 \end{pmatrix} \\ = \begin{pmatrix} 38 & 32 \\ 101 & 86 \end{pmatrix}$$

Multiplication of matrices is not commutative. In fact, the question, "Does $AB=BA$?" is meaningful only if both A and B are square and of the same order. The following properties, however, do hold in general:

$$AI=IA=A, \\ A(BC)=(AB)C, \\ (A+B)C=AC+BC, \text{ and} \\ C(A+B)=CA+CB.$$

Section	Area	Page
05	04	02

Observe that the product of two square matrices of order m requires the calculation of m^2 numbers, each the sum of m products. When the original matrices are initially stored in the primary memory of a computer, their multiplication provides a classical example of the use of index registers.

Scalar multiplication of an $m \times n$ matrix A by a scalar (number) c is that $m \times n$ matrix formed by multiplying each element of A by c . That is, replace each $a_{i,j}$ by $ca_{i,j}$ in order to form cA .

The following are called the elementary row (column) transformations of a matrix, A : (1) multiply each element of any row (column) of A by a scalar; (2) add, element by element, any row (column) of A to any other row (column) of A ; (3) interchange any two rows (columns) of A . Each of these transformations is equivalent to the multiplication of A by a suitably chosen matrix.

The transpose of an $m \times n$ matrix A is an $n \times m$ matrix, A' , such that $a'_{i,j} = a_{i,j}$. That is, the i th row of A' is the i th column of A . The operation of transposition has the property that $(AB)' = B'A'$. A matrix A is called symmetric if $A = A'$.

The determinant, $|A|$, of a square matrix A of order m is a number defined as

$$\sum_{\text{all } f} (\text{sign } f) a_{1,f(1)} \cdot a_{2,f(2)} \cdot \dots \cdot a_{m,f(m)}$$

where $f(1), f(2), \dots, f(m)$ is a permutation of $1, 2, \dots, m$. The summation is over all possible permutations, of which there are $m!$. The term $(\text{sign } f)$ is either $+1$ or -1 , depending upon whether the sequence $f(1), f(2), \dots, f(m)$ can be returned to the natural order $1, 2, \dots, m$ by an even or odd number of interchanges of only two integers at a time. For example,

$$\begin{vmatrix} 1 & 2 & 3 \\ 4 & 3 & 5 \\ 5 & 1 & 2 \end{vmatrix} = 1 \cdot 3 \cdot 2 + 2 \cdot 5 \cdot 5 + 4 \cdot 1 \cdot 3 - 3 \cdot 3 \cdot 5 - 2 \cdot 4 \cdot 2 - 1 \cdot 1 \cdot 5 = 2$$

If A represents the matrix of coefficients of the m unknowns in a set of m simultaneous equations, and X represents the column vector of the unknowns, and C represents the column vector of the constant terms, then the set may be written as $AX = C$. Hence, $X = A^{-1}C$, so that solution of the system is equivalent to a matrix inversion and a matrix-vector multiplication. Note that the solution $A^{-1}C$ exists only if A is nonsingular. Some methods to solve the set $AX = C$ do not explicitly find A^{-1} . However, if the same matrix A is used with various column vectors C , it may be expedient to find A^{-1} once and multiply the various C vectors by it.

Sets of simultaneous linear equations often form by themselves the mathematical statement of many problems. However, they also arise in most

methods to solve partial differential equations by finite difference means, statistical regression analysis (curve fitting), etc.

The direct calculation of a determinant is almost never used in practice except for low orders, since it involves the summation of $m!$ products of m factors each, for example, $20! = 1 \cdot 2 \cdot 3 \dots 19 \cdot 20 = 2 \times 10^{18}$ approximately. The classic algorithm, Cramer's Rule, for solving a set of simultaneous linear equations by means of determinants is similarly impractical.

If the determinant of a square matrix A does not vanish, $|A| \neq 0$, then A is said to be nonsingular and there exists a unique square matrix A^{-1} of the same order such that $AA^{-1} = A^{-1}A = I$. A^{-1} is called the inverse of A . If $|A| = 0$ then A is said to be singular. Henceforth, unless otherwise stated, the matrix A will be assumed square and real; vectors will be real, column vectors.

A set of vectors X_1, X_2, \dots, X_m is said to be linearly dependent if there exist constants c_1, c_2, \dots, c_m not all zero such that $c_1X_1 + \dots + c_mX_m = \theta$ (the zero vector of dimension m). Otherwise, the set is linearly independent.

The set of m column vectors of a nonsingular matrix of order m is linearly independent but any set of $m + 1$ column vectors of dimension m is linearly dependent.

Two vectors X and Y are orthogonal if their inner product vanishes; that is, $X'Y = 0$. A set of vectors which are pairwise orthogonal is linearly independent. A square matrix, A , is called orthogonal if $A'A = AA' = I$; that is, $A' = A^{-1}$.

Let A be a square matrix of order m . Then any scalar λ which satisfies the equation $|A - \lambda I| = 0$ is called an eigenvalue of A . The above equation, which involves a polynomial of degree m in λ , is called the characteristic equation of the matrix A . There are at most m distinct eigenvalues for a matrix of order m . The eigenvalues for a matrix with real elements need not be real but can also occur as pairs of complex conjugate numbers. For example, the matrix $\begin{pmatrix} a & -b \\ b & a \end{pmatrix}$ has $a \pm ib$ as eigenvalues. The eigenvalues for a real symmetric matrix are real and in general A' has the same eigenvalues as A .

For a specified eigenvalue λ , a non-zero vector X such that $AX = \lambda X$ is called a (column) eigenvector of A corresponding to the eigenvalue λ . A matrix may be thought of as defining a linear transformation of an m -dimensional vector space into itself. The eigenvectors of A are those vectors which do not change their direction under the transformation. Such a vector is changed only by a scalar multiple.

A matrix B is similar to a matrix A if there is a nonsingular matrix C such that $B = C^{-1}AC$.

$C^{-1}AC$ is referred to as a similarity transformation of A and it leaves the eigenvalues invariant. If $C^{-1} = C'$ then $C^{-1}AC$ is called an orthogonal transformation of A . For any real symmetric matrix A , there exists an orthogonal matrix S such that $S'AS = D$, a diagonal matrix. This equation can also be written as $AS = SD$ so the diagonal elements of D are the eigenvalues and the columns of S are the corresponding eigenvectors. A highly effective computational algorithm referred to as the Jacobi method creates the matrix S by means of a convergent sequence of 2×2 rotations.

MATRIX INVERSION AND SIMULTANEOUS EQUATIONS METHODS

The methods of solution of simultaneous equations can be broadly classified as (A) direct and (B) iterative. Gaussian elimination is the most commonly used direct method for solving $AX = C$. It is essentially a systematic formulation of the standard elimination procedure which each high school algebra student learns. The method contains two parts: (1) reduction to triangular form by means of elementary row and column operations, and (2) back substitution which successively determines the unknowns starting from one equation in one unknown. The heart of the triangularization algorithm is the formula

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - a_{ik}^{(k)} \frac{a_{kj}^{(k)}}{a_{kk}^{(k)}}$$

The number $a_{kk}^{(k)}$, called a pivot, plays an important role in the accuracy of the method and much consideration has been given to rules for choosing the pivotal elements so as to avoid excessive relative error.

If the matrix A is tri-diagonal, that is, $a_{ij} = 0$ for $|i - j| > 1$, then the Gaussian elimination method is highly efficient in regard to operation count, since it grows linearly with the order m of the matrix. For the general matrix, the operation count grows as the cube of m . For a broad class of tri-diagonal matrices arising from ordinary and partial differential equations A^{-1} is a positive matrix (each element is a positive real number) so it is not economical to deal with A^{-1} directly in this case (operation count of m^2 vs. at most $5m$).

The method of Jordan complete elimination is an extension of Gaussian elimination which uses a sequence of elementary row operations to reduce A to the identity matrix. Since $AA^{-1} = I$, it is clear that these same operations applied to I yield A^{-1} . Again the operation count grows as

m^3 . Economy of storage is possible so that only $m(m + 1)$ memory cells are required for storage of matrix elements. At the end, A^{-1} is stored where A was to begin with.

From a formal point of view the preceding methods can be viewed in terms of the factorization $A = LDU$, where L (U) is a lower (upper) unit triangular matrix (triangular with diagonal elements = 1) and D is the diagonal matrix of pivotal elements. Note that $|A| = |L| \cdot |D| \cdot |U| = |D|$ and $A^{-1} = U^{-1}D^{-1}L^{-1}$.

If A is real symmetric, then both the operation count and storage can be cut almost in half. In this case there is a square root method which consists of the factorization $A = S'S$, where S is an upper triangular matrix.

Where the matrix is too large to handle in main memory, it may be desirable to use partitioning methods, that is, methods which involve matrix operations on sub-arrays of the given matrix. For example, one can treat the matrix A as a 2×2 matrix whose elements are themselves matrices with the diagonal elements being square matrices. This applies in particular where one represents an $m \times m$ complex matrix as a $2m \times 2m$ real matrix.

If B is a matrix which differs from A only in a single element, or in one or more rows or columns, then B^{-1} can be obtained as a modification of A^{-1} . Kron's method for solving network problems involves this principle.

In regard to iterative methods, these often apply to sparse real symmetric positive definite matrices which arise from difference approximations to partial differential equations.

There is a broad pencil and paper art referred to as relaxation methods, but effective digital computer methods require more systematic development and a priori decision-making.

In Seidel's method each equation is uniquely associated with one of the unknowns (for example, first equation with x_1 , second equation with x_2 , etc.) and each equation is solved for its unknown. The resulting form of the system of equations then provides a means of successively correcting an approximate solution. If A is real symmetric and has positive diagonal elements, then convergence occurs if and only if A is positive definite. A substantial improvement in the convergence rate has been achieved by first using corrected values in the equations as soon as they are available (the Gauss-Seidel method) and secondly by introducing a relaxation parameter ω . The effect of the ω is to provide a correction which is a weighted average of the old value and the Gauss-Seidel correction. The convergence of this successive overrelaxation

method was analyzed by David Young (Trans. Amer. Math. Soc. 76, 1954, 92-111) for elliptic partial differential equations, and a formula for the optimum ω was prescribed. Alternating direction methods developed by D. W. Peaceman, H. H. Rachford and J. Douglas (J. Soc. Indust. Appl. Math 3, 1955, 28-41; Trans. Amer. Math. Soc. 82, 1956, 421-439) depend on repeatedly solving tri-diagonal systems of equations.

EIGENVALUES AND EIGENVECTORS

Eigenvalues and eigenvectors are often desired as a part of the solution of partial differential equations. For example, they may specify vibration nodes and resonant frequencies in problems of vibration analysis. In such cases the matrix involved is usually real and symmetric. Statistical factor analysis and reactor problems often give rise to eigenvalue problems with symmetric matrices, whereas flutter analysis often involves complex, nonsymmetric matrices. Sometimes it is required to find only one (such as the largest) eigenvalue; sometimes all are required. This will often govern the choice of computational method.

As indicated earlier, the Jacobi method is a computationally effective algorithm for generating all the eigenvalues and eigenvectors for a real symmetric matrix. Givens' method uses a finite sequence of 2×2 rotations to obtain a real symmetric tri-diagonal matrix similar to the given matrix. The eigenvalues are then determined from this tri-diagonal matrix by various root-finding techniques. Householder's method (see Wilkinson, J. H., Computer J. 3, 1960, 23) also determines a tri-diagonal matrix similar to the given matrix but with a smaller operation count than Givens' method. For general matrices the similarity transformations used in Householder's method yields an almost triangular matrix, that is, a matrix, B , such that $b_{ij} = 0$ for $i - j > 1$ or for $j - i > 1$. By using the matrix B one achieves a data reduction but, more importantly, it is well adapted to the Gaussian elimination which is used in the inverse power method for determining the eigenvalues and eigenvectors of B (see Parlett, B., Technical Report 21, Appl. Math. & Stat. Lab. Stanford University, May 17, 1962). This method gives a correction, X , to an approximate eigenvector y corresponding to an eigenvalue λ .

It is assumed that an approximation $\tilde{\lambda}$ to λ is also given. X is then the solution of the equation $(B - \tilde{\lambda} I) X = y$. Various techniques can be applied to improve λ .

The power method is a classical method which applies to general matrices. It is iterative and finds one eigenvalue and eigenvector at a time, by forming a sequence $X_0, X_1, \dots, X_k, \dots$ where $X_{k+1} = AX_k / \mu_k \cdot \mu_k$ is a normalization factor which, if there is a unique dominant eigenvalue, converges to this value. X_k converges to a corresponding eigenvector. Deflation methods can then be applied to remove this eigenvalue so that subdominant eigenvalues can be found.

BIBLIOGRAPHY

The list below is far from complete but, hopefully, representative. Items (1) and (5) are classics for basic algebraic theory, although both (2) and (9) present necessary algebra in both readable and rigorous form. Item (6) is of a more theoretical nature. Item (3) provides an excellent summary of methods to solve systems of equations and includes a comprehensive bibliography of 131 items. Items (10) and (11) are specifically directed toward machine computation and discuss questions of machine memory and speed as well as algorithmic procedure. White summarizes a number of methods and machine codes; Ralston and Wilf present the Jacobi method for eigenvalues and several procedures for simultaneous equations, complete with flowcharts. Items (4) and (8) discuss physical situations which give rise to matrix problems, and item (7) gives an elementary presentation for beginning business students.

Bodewig, E., Matrix Calculus, Interscience, 1956.

Faddeeva, V. N., Computational Methods of Linear Algebra, Dover Publications, Inc., 1959.

Forsythe, G. E., "Solving Linear Equations Can Be Interesting", Bulletin of the American Mathematical Society, vol. 59, 1953, pp. 299-329.

Forsythe, G. E., and W. R. Wasow, Finite-Difference Methods for Partial Differential Equations, John Wiley and Sons, Inc., 1960.

Frazer, R. A., W. J. Duncan, and A. R. Collar, Elementary Matrices, Macmillan, 1938.

Householder, A. S. , Principles of Numerical Analysis, McGraw-Hill, 1953.

Kemeny, J. G. , A. Schleifer, J. L. Snell, and G. L. Thompson, Finite Mathematics with Business Applications, Prentice-Hall, Inc. , 1962.

Lanczos, C. , Applied Analysis, Prentice-Hall, Inc. , 1956.

Perlis, S. , Theory of Matrices, Addison-Wesley Publishing Co. , Inc. , 1952.

Ralston, A. , and H. S. Wilf. eds. , Mathematical Methods for Digital Computers, John Wiley and Sons, Inc. , 1960.

White, P. A. , "Eigenvalue and Eigenvector Computations of a Matrix", Journal of the Society for Industrial and Applied Mathematics, vol. 6, Dec. 1958, pp. 393-437.

Varga, R. S. , Matrix Iterative Analysis, Prentice-Hall, Inc. , 1962.



LINEAR PROGRAMMING

Programming problems are concerned with the efficient use or allocation of limited resources to meet desired objectives. These problems are characterized by the large number of solutions that satisfy the basic conditions of each problem. The selection of a particular solution as the best solution to a problem depends on some aim or overall objective that is implied in the statement of the problem. A solution that satisfies both the conditions of the problem and the given objective is termed an optimum solution.

One very special subclass of mathematical programming problems consists of linear programming problems. A linear programming problem differs from the general variety in that a mathematical model or description of the problem can be stated, using relationships which are called "straight-line," or linear. Mathematically, these relationships are of the form

$$a_1x_1 + a_2x_2 + \dots + a_jx_j + \dots + a_nx_n = a_0$$

where the a_j 's are known coefficients and the x_j 's are unknown variables. The mathematical statement of a linear programming problem includes a set of simultaneous linear equations which represent the conditions of the problem and a linear function which expresses the objective of the problem.

This area describes the basic concepts of linear programming, reviews standard computational techniques, and summarizes the wide variety of applications. The mathematical presentation will employ the matrix notation of the Matrix Methods area. To take full advantage of the power of this mathematical tool requires an understanding of its foundations; therefore, additional reading is recommended from items listed in the bibliography.

LINEAR PROGRAMMING DEFINITIONS AND BASIC THEOREMS

The linear programming problem is as follows:

Find a set of numbers x_1, x_2, \dots, x_n which minimizes (or maximizes) the linear objective function.

$$(1) c_1x_1 + c_2x_2 + \dots + c_jx_j + \dots + c_nx_n$$

subject to the linear constraints

$$\begin{matrix} a_{11}x_1 + a_{12}x_2 + \dots + a_{1j}x_j + \dots + a_{1n}x_n = a_{10} \\ \vdots \\ a_{i1}x_1 + a_{i2}x_2 + \dots + a_{ij}x_j + \dots + a_{in}x_n = a_{i0} \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mj}x_j + \dots + a_{mn}x_n = a_{m0} \end{matrix}$$

and the non-negativity constraints

$$(3) \begin{matrix} x_1 & \geq & 0 \\ & x_2 & \geq & 0 \\ & & \ddots & \\ & & & x_j & \geq & 0 \\ & & & & \ddots & \\ & & & & & x_n & \geq & 0 \end{matrix}$$

In other words, in a linear-programming problem one wishes to find a non-negative solution to a set of linear constraints which optimizes (i. e., minimizes or maximizes) a linear objective function. The c_j are called cost coefficients and the a_{i0} are called right-hand-side coefficients. The linear constraints may be equations, as noted above, or linear inequalities of the form

$$(4) a_{i1}x_1 + \dots + a_{in}x_n \leq a_{i0}$$

or

$$(5) a_{i1}x_1 + \dots + a_{in}x_n \geq a_{i0}$$

Inequalities like (4) and (5) can be transformed to equations by the suitable addition or subtraction of a non-negative slack-variable. For (4) we would have

$$(6) a_{i1}x_1 + \dots + a_{in}x_n + x_{n+1} = a_{i0} \\ x_{n+1} \geq 0$$

and for (5) we would have

$$(7) a_{i1}x_1 + \dots + a_{in}x_n - x_{n+1} = a_{i0} \\ x_{n+1} \geq 0$$

Note that each inequality has a different slack-variable associated with it. Each slack-variable measures the difference between the left and right-hand sides of the given inequality.

As the form of a linear programming problem requires that all variables be non-negative, we should note that a variable not restricted to be non-negative can always be expressed as the difference between

two non-negative variables -- for example:

$$x_i = x_i' - x_i''$$

where

$$x_i' \geq 0 \text{ and } x_i'' \geq 0$$

A linear programming problem may contain any mixture of linear constraints. For computational purposes, the basic constraints (2) of a linear programming problem must always be expressed in terms of equations such that the number of equations (m) is less than the number of variables (n). This causes (2) to be an underdetermined set of linear equations which has many possible solutions. As each equation can be considered as a hyperplane in n-dimensional space, the solution space of the set of linear equations is a convex polyhedron. The computational algorithms of linear programming determine from all possible solutions one that optimizes the objective function. Since the convex polyhedron may be unbounded in the direction of optimization, the optimum value of the objective function can also be unbounded. We will only discuss the minimizing case, as maximizing a linear function is equal to minimizing the negative of the linear function.

In matrix notation linear programming may be described by

$$\begin{aligned} cX &= \min \\ \text{subject to} \\ AX &= b \\ X &\geq 0 \end{aligned}$$

where $c = (c_1, c_2, \dots, c_n)$ is a row vector;

$$X = (x_1, x_2, \dots, x_n)$$

is a column vector; A is the $m \times n$ matrix of coefficients; $b = (a_{10}, a_{20}, \dots, a_{m0})$ is a column vector; and $0 = (0, 0, \dots, 0)$ is an n-rowed column vector. As it is convenient to consider the columns of A as points in m-dimensional space, the linear programming problem can be written as

$$\begin{aligned} cX &= \min \\ \text{subject to} \\ x_1 P_1 + x_2 P_2 + \dots + x_n P_n &= P_0 \\ x_j &\geq 0 \end{aligned}$$

where $P_j = (a_{1j}, a_{2j}, \dots, a_{mj})$ is a column vector for $j = 0, 1, \dots, n$.

A feasible solution to the linear programming problem is a vector $X = (x_1, x_2, \dots, x_n)$ which satisfies conditions (2) and (3).

A basic feasible solution is a feasible solution with no more than m positive x_i .

A nondegenerate basic feasible solution is a basic feasible solution with exactly m positive x_i .

A minimum feasible solution is a feasible solution which also minimizes (1).

A basis is a linearly independent set of vectors $B_1 B_2 \dots B_m$ chosen from the rectangular matrix $A = (P_1 P_2 \dots P_n)$.

This set of vectors forms a square matrix B of size m and since $B_1 B_2 \dots B_m$ are independent the inverse of B, B^{-1} , exists. The vector $X_0 = (x_{10}, x_{20}, \dots, x_{m0})$ are the unknown variables associated with the vectors $B_1 B_2 \dots B_m$. Hence the equations $BX_0 = P_0$ have a solution: $X_0 = B^{-1} P_0$.

A feasible basis for the linear programming problem is a vector $X_0 = B^{-1} P_0$ such that each element of X_0 is non-negative. This is sometimes written $X_0 \geq 0$ or $B^{-1} P_0 \geq 0$ or $(x_{10}, x_{20}, \dots, x_{m0}) \geq 0$. The solution to the linear programming problem for this basis is $X = (x_{10}, x_{20}, \dots, x_{m0}, 0, \dots, 0)$ where the vector X is understood to be reordered. Finding a feasible basis corresponds to selecting a determined square set of equations from the given undetermined rectangular set and letting those variables not in the basis be equal to zero.

A convex combination of the vectors V_1, V_2, \dots, V_n is a vector

$$V = \alpha_1 V_1 + \alpha_2 V_2 + \dots + \alpha_n V_n \text{ where the } \alpha_i \text{ are scalars, } \alpha_i \geq 0, \text{ and } \sum \alpha_i = 1.$$

A subset of points C of Euclidean space is a convex set if, and only if, for all pairs of points V_1 and V_2 in C, any convex combination

$$V = \alpha_1 V_1 + \alpha_2 V_2 = \alpha V_1 + (1 - \alpha)V_2 \quad (1 \geq \alpha \geq 0)$$

is also in C. A convex set is one which contains the straight line joining any two points in the set.

A point V in a convex set C is called an extreme point if V cannot be expressed as a convex combination of any other two distinct points in C.

Theorem 1. The set of all feasible solutions to the linear programming problem is a convex set.

Theorem 2. The objective function (1) assumes its minimum at an extreme point of the convex set C generated by the set of feasible solutions to the linear programming problems. If it assumes its minimum at more than one extreme point, then it takes on the same value for every convex combination of those particular points.

Theorem 3. If a set of $k \leq m$ vectors $P_1 P_2, \dots, P_k$ can be found that are linearly independent and such that

$$x_1 P_1 + x_2 P_2 + \dots + x_k P_k = P_0$$

and all $x_i \geq 0$, then the point $X = (x_1, x_2, \dots, x_k, 0, \dots, 0)$ is an extreme point of the convex set of feasible solutions. Here X is an n-dimensional vector whose last n-k elements are zero.

Theorem 4. If $X = (x_1, x_2, \dots, x_n)$ is an extreme point of C , then the vectors associated with positive x_i form a linearly independent set. From this it follows that, at most, m of the x_i are positive.

Theorem 5. $X = (x_1, x_2, \dots, x_n)$ is an extreme point of K if and only if the positive x_i are coefficients of linearly independent vectors P_j in

$$\sum_{j=1}^n x_j P_j = P_0$$

Theorem 6. If a feasible solution exists, then a basic feasible solution exists.

Theorem 7. If the objective function possesses a finite minimum, then at least one optimal solution is a basic feasible solution.

These theorems enable us to restrict the search for an optimum solution to extreme points of the convex set C of all possible solutions.

If the linear programming problem is stated as the following primal problem:

$$\begin{aligned} cX &= \min \\ \text{subject to} \\ AX &\geq b \\ X &\geq 0 \end{aligned}$$

then the corresponding dual problem is given by

$$\begin{aligned} Wb &= \max \\ \text{subject to} \\ WA &\leq c \\ W &\geq 0 \end{aligned}$$

where $W = (w_1, w_2, \dots, w_m)$ is the row vector of unknowns for the dual problem.

The Duality Theorem. If either the primal or the dual problem has a finite optimum solution, then the other problem has a finite optimum solution and the extremes of the linear functions are equal, i. e., $\min cX = \max Wb$. If either problem has an unbounded optimum solution, then the other problem has no feasible solutions.

The concept of duality and the duality theorem have much significance in the theoretical and computational aspects of linear programming.

LINEAR PROGRAMMING TECHNIQUES

A. The Simplex Method

The basic computational procedure for solving any linear programming problem is the simplex method. With the simplex method, we can, once a first basic (extreme-point) feasible solution has been determined, obtain a minimum basic feasible solution in a finite number of steps. These steps, or iterations, consist in finding a new basic feasible solution whose corresponding value of the objective

function is less than (or at worst equal to) the value of the objective function for the preceding solution. This process is continued until a minimum solution with either a finite or infinite value of the objective function has been reached. The mathematical description of the standard simplex method follows. The name simplex was given to this procedure as one of the first examples solved with this technique contained the equation $x_1 + x_2 + \dots + x_n = 1$, which is the equation of a simplex (generalized tetrahedron) with unit intercepts in n -dimensional space.

Assume that all coefficients of vector P_0 are non-negative. We can always multiply an equation by -1 to make the corresponding $a_{i0} \geq 0$. Let B be a feasible basis -- that is, a basic feasible solution has been found from the equations $B_1 X_{01} = P_0$, where $X_{01} = B_1^{-1} P_0 \geq 0$. In practice, B_1 is usually a unit matrix of order m , and the corresponding first feasible solution is readily obtained since the inverse of a unit matrix is a unit matrix. For this case, $X_{01} = P_0$. In those situations where a suitable unit matrix is not given as part of the problem, an artificial unit basis is attached to the problem to start it off. This device is discussed below.

Assuming a unit basis for the first feasible solution and reordering the vectors of the basis so that $B_1 = (P_1 P_2 \dots P_m)$ (this step is not necessary, but is done here to aid in the discussion), the simplex tableau (computational tableau) assumes the following form (1 represents the letter l):

Basis	P_0	$P_1 \dots P_l \dots P_m$	$P_{m+1} \dots P_j \dots P_k \dots P_n$
P_1	x_{10}	1 ... 0 ... 0	$x_{1,m+1} \dots x_{1j} \dots x_{1k} \dots x_{1n}$
\vdots	\vdots	\vdots	\vdots
P_l	x_{l0}	0 ... 1 ... 0	$x_{l,m+1} \dots x_{lj} \dots x_{lk} \dots x_{ln}$
\vdots	\vdots	\vdots	\vdots
P_m	x_{m0}	0 ... 0 ... 1	$x_{m,m+1} \dots x_{mj} \dots x_{mk} \dots x_{mn}$
	x_{00}	0 ... 0 ... 0	$x_{0,m} \dots x_{0j} \dots x_{0k} \dots x_{0n}$

where $x_{ij} = a_{ij}$, for $i = 1, \dots, m; j = 0, 1, \dots, n$; the basic feasible solution is $X_{01} = (x_{10}, \dots, x_{l0}, \dots, x_{m0}) = B_1^{-1} P_0$; and in general we can define $X_{j1} = (x_{1j}, x_{2j}, \dots, x_{mj}) = B_1^{-1} P_j$. The value of the objective function is $x_{00} = \sum_i c_i x_{i0}$.

The numbers x_{0j} for $j = 1, \dots, n$ are defined by $x_{0j} = \sum_i c_i x_{ij} - c_j$. The summation term is called indirect cost and is sometimes denoted by $z_j = \sum_i c_i x_{ij}$. Note that $x_{0j} = 0$ for any j in the basis.

The following theorems indicate the need and the use of the x_{0j} .

Theorem 1. If, for any j , the condition $x_{0j} > 0$ holds, then a set of feasible solutions can be constructed, such that $x'_{00} < x_{00}$ for any member of the set, where the lower bound of x_{00} is either finite or infinite (x'_{00} is the value of the objective function for a particular member of the set of feasible solutions).

Case I. If the lower bound is finite, a new feasible solution consisting of exactly m positive variables can be constructed whose value of the objective function is less than the value of the preceding solution -- that is, $-\infty < x'_{00} < x_{00}$.

Case II. If the lower bound is infinite, a new feasible solution consisting of exactly $m+1$ positive variables can be constructed whose value of the objective function can be made arbitrarily small -- that is, $\lim x'_{00} = -\infty$.

Theorem 2. If for any basic feasible solution $X = (x_{10}, x_{20}, \dots, x_{m0})$ the conditions $x_{0j} \leq 0$ hold for all $j = 1, 2, \dots, n$, then the solution is a minimum feasible solution.

Theorem 1 assumes non-degeneracy -- that is, all basic feasible solutions to the problem are strictly positive (all $x_{i0} > 0$). The assumption is required from a theoretical point of view as it enables one to prove that the simplex method will converge in a finite number of steps. If a particular problem can have a degenerate basic feasible solution then there is the possibility that the procedure will cycle, that is, return to the same basis after a finite number of steps, and hence, not converge to the optimum solution. Although examples have been constructed which do cycle, it is quite uncommon and a problem, degenerate or not, usually converges. Although not usually employed, computational devices exist which will guarantee the convergence of any problem.

To determine a new basic feasible solution, the following steps are carried out. These steps change the basis one vector at a time until a stop condition is encountered.

1. Compute all x_{0j} .

2. Are all $x_{0j} \leq 0$ for $j = 1, 2, \dots, n$? (This set of inequalities is called the optimality criterion.) If yes, the current solution is an optimum solution and the procedure stops. For any $x_{0j} = 0$ with P_j not in the optimum basis, an alternate optimum solution can be obtained by introducing this vector into the basis. If no, select for the vector to be introduced into the new solution the vector P_k whose $x_{0k} = \max_j x_{0j} > 0$. If ties occur, select any one.

3. To insure feasibility of the new solution, the vector to be eliminated from the basis is the vector P_1 corresponding to $x_{10}/x_{1k} = \min_{x_{ik} > 0} x_{i0}/x_{ik}$. If ties occur, select any one. If all $x_{ik} \leq 0$, then the problem has an unbounded optimum solution and the procedure stops. The ratio $x_{10}/x_{1k} \geq 0$ and if it happens to equal zero (this implies the degenerate case with $x_{10} = 0$) the value of the objective function for the new solution will be the same as before. The element x_{1k} is called the pivot element.

4. Determine the new solution and new simplex tableau by applying the following formulas (Gaussian elimination):

$$x'_{ij} = x_{ij} - \frac{x_{1j} x_{ik}}{x_{1k}} \quad i \neq 1$$

$$x'_{1k} = \frac{x_{1j}}{x_{1k}}$$

These formulas hold for $i = 0, 1, \dots, m$ and $j = 0, 1, \dots, n$ and the x'_{ij} for $j = 0$ is the new basic feasible solution; x'_{00} the new value of the objective function; x'_{0j} the new indirect minus direct cost numbers. This transformation is equivalent to determining a new feasible basis B_2 such that the new solution vector is $X_{02} = B_2^{-1} P_0$ and the $X_{j2} = B_2^{-1} P_j$.

The above steps are repeated for the data in the new tableau. Note that the transformation will cause the unit matrix of the initial tableau to be transformed to the inverse of the current basis.

If a unit basis is not explicitly contained in the original statement of the problem, a set of artificial non-negative variables is attached to the system, one new variable for each equation. In some instances a full set of m artificial variables will not be required as the original problem contains a partial set of unit vectors. The costs coefficients for the artificial variables are assumed to be infinite, and hence, if a minimum feasible solution to the original problem exists, the simplex method will drive the values of the artificial variables to zero. If the original problem does not have any feasible solutions, the simplex method will terminate with artificial variables in the optimum solution at a positive level. The computational tableau and steps of the simplex method can be readily modified to take care of the infinite costs.

B. Revised Simplex Method

The kernel of the computational aspects of linear programming resides in the calculation of the inverse of the current basis. Given the inverse, all quantities required in a simplex iteration can be determined. Letting B_t be the feasible basis of the t^{th} iteration we have

$$X_{0t} = B_t^{-1}P_0$$

$$X_{jt} = B_t^{-1}P_j$$

$$\pi_t = c_t B_t^{-1}$$

$$\pi_t P_j = c_t B_t^{-1} P_j$$

$$\pi_t P_0 = c_t X_{0t} = c_t B_t^{-1} P_0$$

where X_{0t} is the solution vector for the t^{th} basic feasible solution; X_{jt} are the vectors which express the given vectors P_j as linear combinations of the basis vectors; c_t is the row vector of cost coefficients of the vectors in the t^{th} basis; the elements of the row vector π_t are called the simplex multipliers; $\pi_t P_j$ is the indirect cost for the P_j vector; $\pi_t P_0$ is the value of the objective function for the t^{th} basis. From a computational point of view using the explicit representation of the inverse and the simplex multipliers has a number of advantages. These include the reduction of the amount of computation and the reduction of the amount of information that has to be recorded per iteration. Where the standard simplex method transforms and records the complete simplex tableau, the revised procedure needs only to record the new inverse and solution vector. Note that the revised procedure uses the original data in each step and if, as is the case in many problems, the data contains many zeros, computation time can be saved.

The product-form of the revised simplex method uses the fact that the inverse of the feasible basis (which starts out as an identity matrix) can be expressed as the product of elementary transformation matrices. Each such matrix, which differs from a unit matrix in the $\underline{1}^{\text{th}}$ column ($\underline{1}$ corresponds to the row position of the vector

eliminated from the basis), contains the information necessary to determine the row inverse. For the t^{th} iteration let

$$E_t^{-1} = \begin{pmatrix} 1 & \dots & y_{\underline{1}\underline{1}} & \dots & 0 \\ \vdots & & \vdots & & \vdots \\ 0 & \dots & y_{\underline{1}\underline{1}} & \dots & 0 \\ \vdots & & \vdots & & \vdots \\ 0 & \dots & y_{\underline{m}\underline{1}} & \dots & 1 \end{pmatrix}$$

where

$$y_{\underline{i}\underline{1}} = \frac{x_{\underline{i}k}}{x_{\underline{1}k}} \quad i \neq \underline{1}$$

$$y_{\underline{1}\underline{1}} = \frac{1}{x_{\underline{1}k}}$$

The inverse for the t^{th} basis is given by

$$E_t^{-1} E_{t-1}^{-1} \dots E_2^{-1} E_1^{-1} = B_t^{-1}$$

with $E_1 = I$. With this condensed form only a limited amount of information needs to be recorded. It has been demonstrated that for most linear programming problems, the product form of the revised simplex method is the most efficient. Computational experience has shown that the number of iterations required to find an optimum solution can vary between m and $3m$. The number appears to be a function of the number of equations more than the number of variables. The number varies with algorithm used, method of finding first feasible solution and the criterion used to select a vector to go into the new basis.

C. The Transportation Problem Algorithm

Because of the structure of the matrix of coefficients which defines the transportation problem (see applications discussion), the calculations of the simplex method as applied to the transportation problem are greatly simplified. Letting m = the number of origins and n = the number of destinations, each basic feasible solution of the transportation problem corresponds to an $(n + m - 1)$ triangular basis.

This enables one to readily compute the solution and, since each coefficient of the basis is 0 or 1, the solution will be in integers if the original availabilities and requirements are integers. The simplex algorithms for the transportation problem utilize either the solution to the dual problem or the stepping-stone method. The main difference between the two techniques is the method of computing the indirect costs. The dual method is the one used in most computer codes. Other procedures exist for solving this problem -- for example, the Hungarian method and the flow method. A number of variations to the transportation problem exist which also have specialized computational procedures. These include the capacitated transportation problem in which the amount shipped between any origin and destination has an upper bound; and the generalized transportation problem or machine loading problem in which the coefficients in the equations are not restricted to being either 0 or 1.

D. The Dual Simplex Algorithm

In some instances, it is easier to find a basis for which the optimality criterion is satisfied, that is, all $x_{oj} \leq 0$, and for which the feasibility criterion is not, that is, not all $x_{io} \geq 0$. If this is the case, the dual simplex algorithm can be employed. The only difference is in the criteria used to select the vector (variable) to be introduced into the basis and the one that leaves the basis. Here the vector to leave is determined first and corresponds to

$$x_{\underline{1}o} = \min x_{io} < 0.$$

The vector introduced corresponds to the index k for which

$$\frac{x_{ok} - c_k}{x_{\underline{1}k}} = \min_{x_{\underline{1}j} < 0} \frac{x_{oj} - c_j}{x_{\underline{1}j}} > 0$$

The pivot element is $x_{\underline{1}k}$ and the elimination transformation is the same as in the standard simplex method. The above analysis causes at least one of the negative solution variables to become positive and also keeps the transformed $x_{oj} \leq 0$ (that is, the transformation improves the feasibility of the solution while still satisfying the optimality criterion).

The original simplex method (primal method) and the dual method have been combined in some instances in a primal-dual or composite simplex

computer code. These techniques place no restrictions on the signs of the c_j or b_i and enable one to obtain an initial basic solution which satisfies either the feasibility or optimality conditions.

E. Integer Programming

Many linear programming problems require the solution to be in terms of integers -- for example, whole units to be manufactured. Unlike the transportation problem, the simplex algorithm will not guarantee an integer solution for the general linear programming problem. However, variations of the simplex method exist which will guarantee the finding of an integer optimum solution, if one exists. This integer programming procedure adds, in a systematic fashion, new constraints or cutting-planes to the original set of constraints. The new constraints change the convex set of solutions so that it will contain as an optimum extreme point a point with integer coordinates. Integer programming codes do exist, but as the number of iterations required to solve a particular problem appears to rest heavily on the structure and data of the problem, the utilization of these codes in an operational environment is still open to question. Investigation of this problem has been a continuing one by IBM Research.

F. Upper Bound Conditions

In many instances, the variables of a linear programming are bounded from above, that is, $x_j \leq u_j$. Again, a slight variation of the basic simplex method can solve the bounded problem without the explicit representation of the upper bound constraints in the simplex tableau. This procedure can be used when all or some of the variables are bounded. The situation when the x_j are bounded below is easily handled by direct substitution, that is, if $d_j \leq x_j$. Let $x_j = d_j + x'_j$ and substitute $d_j + x'_j$ for the corresponding x_j .

G. Sensitivity Analysis

Sensitivity analysis in linear programming deals with the investigation of how the optimum solution varies with changes in the original data. For example, we are interested in how much a particular cost-coefficient can vary before the computed optimum is no longer optimal; or how much a right-hand side coefficient b_i can vary before the solution is no longer feasible; or finally, what is the effect on the optimum solution if a change is made in an

a_{ij} . Techniques are available for performing these analyses and are usually included in linear programming codes. Additional procedures exist for handling the parametric programming cases where each cost coefficient and/or each right-hand side coefficient is a linear function of the same parameter -- for example, $c_j^i = c_j^i + \lambda d_j$. This procedure yields sets of optimum solutions corresponding to ranges of the coefficients.

H. Decomposition Algorithm

Although it is theoretically possible to solve any given linear programming model, the analyst is quickly made aware of certain limitations which restrict his endeavors. Chief among these limitations is the problem of dimensionality. Almost all difficulties that arise in the development of a programming problem can be related to its size. This is certainly true for such restrictive items as the cost of data gathering, matrix preparation, computing costs, reasonableness of the linear model, etc.

For many problems the constraints consist of rather large subsets of equations which are related in that they refer to the same time period or same production facility, and these subsets are tied together by a small set of equations. These "tie-in equations" might represent total demand for a product. In problems of this sort we have, in a sense, a number of separate linear programming problems whose joint solution must satisfy a set of additional restrictions. If we boxed in the sets of constraints and corresponding part of the objective function, the system shown in Figure 1 would result.

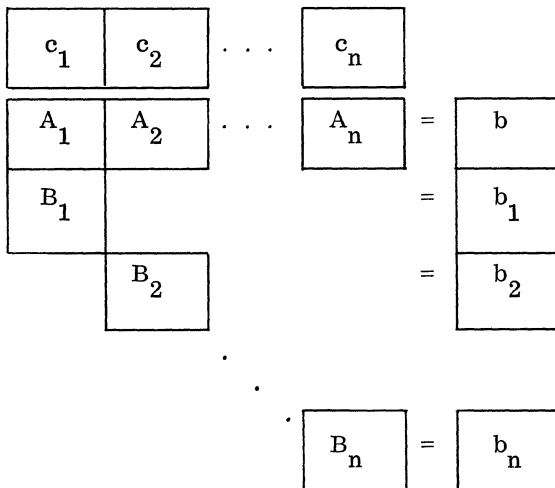


Figure 1. A block angular system

Here we have partitioned the original problem of $Ax = b, x \geq 0, cx$ a minimum into the decomposed program of finding the vectors $x_j \geq 0$ ($j = 1, 2, \dots, n$) such that

$$\sum_j A_j x_j = b \quad (8)$$

$$B_j x_j = b_j \quad (9)$$

$$\sum_j c_j x_j \text{ is a minimum,} \quad (10)$$

where A_j is an $m \times n_j$ matrix, B_j is an $m_j \times n_j$ matrix, c_j is an n_j -vector, b is an m -vector, and b_j is an m_j -vector; x_j is a variable n_j -vector. As written above, this problem has $m + \sum m_j$ constraints and $\sum n_j$ variables.

Assume that for each j , there is available the corresponding convex set S_j of solutions to each subproblem $B_j x_j = b_j, x_j \geq 0$. Then the solution of the original problem could be thought of as the selection of a convex combination of solution points from S_j for each j so as to satisfy the tie-in restrictions $\sum A_j x_j = b$ and make $\sum c_j x_j$ a minimum. Although this bare idea of the decomposition appears fraught with its own difficulties, there are a few saving features which enable us to consider the optimization of an $m + n$ constraint problem subject to the solution of n ($m_j \times n_j$) subproblems instead of one large problem with $m + \sum m_j$ constraints.

The new problem to be considered is called the extremal program and it arises in the following fashion. Let us consider for some j a particular extreme point solution x_{jk} of the convex set of solutions S_j to the subproblem $x_j \geq 0, B_j x_j = b_j$. Define for each such extreme point k

$$P_{jk} = A_j x_{jk}, \quad c_{jk} = c_j x_{jk}.$$

The extremal program is to find the numbers $s_{jk} \geq 0$ satisfying for all (j, k)

$$\sum_j \sum_k P_{jk} s_{jk} = b, \quad (11)$$

$$\sum_k s_{jk} = 1 \quad (\text{all } j) \quad (12)$$

$$\sum_j \sum_k c_{jk} s_{jk} \text{ is a minimum.} \quad (13)$$

As Dantzig and Wolfe (see Bibliography) note, "The relation of the extremal problem to the original problem lies in the fact that any point of

S_j , because it is bounded (assumed) and a convex polyhedral set, may be written as a convex combination of its extreme points, that is, $\sum_k x_{jk} s_{jk}$, where the s_{jk} satisfy (12); and the expressions (11) and (13) are just the expressions (8) and (10) of the decomposed problem rewritten in terms of the s_{jk} .

As mentioned above, this transformation yields a problem with only $m + n$ constraints, the m tie-in constraints (11), and the set of n single constraints (12). However, the number of variables has been increased to the total of all extreme points to the convex polyhedral S_j , an extremely large number. The saving element of the decomposition principle is that we need consider only a small number of this total and we need only the explicit representation of those to be considered, and only when required.

LINEAR PROGRAMMING APPLICATIONS

It should be stressed that the models described below, even though important in their own right, should be looked upon as "starting positions" or basic models in one's approach to problem formulation. The user's problems might resemble a particular model and, with suitable modifications to the basic model, a good first approximation to the real-life situation can be developed. A good example would be a transportation model which must be combined with a production and storage model in order to reflect the true environment. One must guard against forcing a problem to fit a particular model and, at the same time, recognize that a basic model might lend itself to the evolution of an acceptable model.

A. The Basic Transportation Problem

A homogeneous product is to be shipped in the amounts a_1, a_2, \dots, a_m , respectively, from each of m shipping origins and received in amounts b_1, b_2, \dots, b_n , respectively, by each of n shipping destinations. The cost of shipping a unit amount from the i^{th} origin to the j^{th} destination is c_{ij} and is known for all combinations (i, j) . The problem is to determine the amounts x_{ij} to be shipped over all routes (i, j) so as to minimize the total cost of transportation. The a_i are the availabilities and the b_j are the requirements.

To insure the consistency of the mathematical model, we restrict the $\sum_i a_i = \sum_j b_j$ -- that is, the sum of the availabilities equals the sum of the requirements. This is not a restriction which must be adhered to for a particular application in

that if $\sum_i a_i > \sum_j b_j$, a dummy destination (for example, warehouse facility) is added to the problem with a requirement of $\sum_i a_i - \sum_j b_j$. If $\sum_i a_i < \sum_j b_j$ a dummy origin (for example, purchases from a competitor) is added to the problem with an availability of $\sum_j b_j - \sum_i a_i$. Letting $x_{ij} \geq 0$ be the unknown amount to be shipped from origin i to destination j (that is, the set of variables) the linear programming model for $m = 2$ and $n = 3$ is then:

Minimize

$$c_{11}x_{11} + c_{12}x_{12} + c_{13}x_{13} + c_{21}x_{21} + c_{22}x_{22} + c_{23}x_{23}$$

subject to

$$\begin{array}{rccccccc} x_{11} & + & x_{12} & + & x_{13} & & & = & a_1 \\ & & & & & x_{21} & + & x_{22} & + & x_{23} & = & a_2 \\ x_{11} & + & & & & + & x_{21} & & & & = & b_1 \\ & & x_{12} & & & & & + & x_{22} & & = & b_2 \\ & & & & x_{13} & & & & & + & x_{23} & = & b_3 \\ & & & & & & & & & & & & x_{ij} \geq 0 \end{array}$$

As any one of the above equations can be deduced from the remaining $(m + n - 1)$, the rank of the system is $(m + n - 1)$. Any feasible basis is of order $(m + n - 1)$ and the corresponding matrix is triangular.

This model has wide application with the sources and destinations taking on a variety of interpretations -- for example, warehouse, stores, ports, and the cost coefficients representing distances, times, dollars, etc. The model can be modified, for example, to supply information which would enable management to select from a set of possible new origins the one which is the best with respect to transportation costs. Here a set of problems would be solved, one for each possibility, with data reflecting the anticipated costs between the new origin and old destinations and the anticipated availability at the new origin and requirements for the destinations. A similar analysis would be done in order to aid in the selection of a new destination, or in the selection of an origin or destination to be closed down.

B. Transshipment Problem

This is a transportation problem in which the origins and destinations can act as an intermediate shipping point from which the goods are transshipped to their

final destination. This problem can be transformed into a basic transportation problem and can be solved by the procedures applicable to this problem.

C. Capacitated Transportation Problem

This is a basic transportation problem in which the possible shipments are bounded from above -- that is, $x_{ij} \leq u_{ij}$. This problem can be solved with slight modification to the computational procedure for the basic problem. A special primal-dual algorithm is also available.

D. Generalized Transportation Problem and Machine-Assignment Problem

In this problem, the constraint set is of the form:

Minimize

$$\sum_i \sum_j c_{ij} x_{ij}$$

subject to

$$(1) \quad \sum_{j=1}^n a_{ij} x_{ij} = a_i \quad (i=1, \dots, m)$$

$$(2) \quad \sum_{i=1}^m b_{ij} x_{ij} = b_j \quad (j=1, \dots, n)$$

$$x_{ij} \geq 0$$

Problems which fit this model can be found in the transportation field and in problems of machine assignment. For the latter, the $b_{ij} = 1$, equations (1) become inequalities (\leq), the a_{ij} represent the time it takes to process one unit of product j on machine i , x_{ij} the number of units of j produced on machine i , a_i the time available on machine i , b_j the number of units j which must be completed, and c_{ij} the cost of processing one unit of product j on machine i .

E. Multi-Dimensional Transportation Problem

The model of this problem can take on two forms:

a. Minimize

$$\sum_i \sum_j \sum_k c_{ijk} x_{ijk}$$

subject to

$$\sum_i x_{ijk} = a_{jk}$$

$$\sum_i x_{ijk} = b_{ik}$$

$$\sum_k x_{ijk} = c_{ij}$$

$$x_{ijk} \geq 0$$

b. Minimize

$$\sum_i \sum_j \sum_k c_{ijk} x_{ijk}$$

subject to

$$\sum_j \sum_k x_{ijk} = a_i$$

$$\sum_i \sum_k x_{ijk} = b_j$$

$$\sum_i \sum_j x_{ijk} = d_k$$

$$x_{ijk} \geq 0$$

F. Allocation or Assignment Problems

In this problem, we have a number of individuals, machines, etc., to be assigned to perform a set of jobs. Each individual i has a given rating c_{ij} which measures his effectiveness in doing job j . An individual can be assigned to only one job. If x_{ij} represents the assignment of the i^{th} to the j^{th} job, the linear programming formulation is then:

Maximize

$$\sum_{i,j} c_{ij} x_{ij}$$

subject to

$$\sum_j x_{ij} = a_i \quad i = 1, \dots, m$$

$$\sum_i x_{ij} = b_j \quad j = 1, \dots, n$$

$$x_{ij} \geq 0$$

where a_i is the number of persons of type i available and b_j is the number of jobs of type j available. We assume $\sum a_i = \sum b_j$. In many instances all a_i and b_j equal 1 and $m = n$. Mathematically, the assignment problem is a transportation problem.

G. Tanker Routing Problem

Given a number of loading points $i = 1, 2, \dots, m$ at which tankers are to be loaded for delivery to destination points $j = 1, 2, \dots, n$. We know the time at which a tanker is to be loaded at i for making a delivery to j and the time it takes for a tanker to go from any point to any other. (We assume all tankers are identical and delivery is in units of tanker capacity.) Having made a delivery a tanker can travel to any loading point. The problem is to devise routes for each tanker so that the given loading schedule is met with the minimum number of tankers.

H. Network Flow Problems

Given a network (road, railway, pipeline, etc.) consisting of a single source (origin), a single sink (destination), and intermediate nodes (transfer points). Let x_{ij} be the flow between any point i to point j of the network. A point is either the source, sink, or any node. We assume that the network is oriented and there is a capacity restriction $c_{ij} \geq 0$ on each branch (i, j) . We wish to determine the flow through the network which maximizes the flow from source to sink.

With n nodes in the network, with $i = 1$ the source and $i = n$ the sink, we have the following formulation:

Maximize

$$\sum_j x_{ij} = \sum_j x_{jn}$$

subject to

$$(1) \quad \sum_j x_{ji} = \sum_j x_{ij} \quad i = 2, \dots, n-1$$

$$(2) \quad x_{ij} \leq c_{ij}$$

where (1) represents the conservation of flow at the nodes -- that is, what flows in must flow out and the summations are taken over these j which form branches for the given i . The objective function maximizes the flow out of the source or the flow into the sink.

I. Bid Evaluation

Whenever a government agency wishes to procure items from civilian sources, producers of the items must be invited to participate in the bidding of contracts. The individual manufacturer submits bids which reflect his desire for profit, his guess about the other fellow's bid, and his own peculiar limitations. The agency must award contracts in such a way that the total dollar cost to the government is at a minimum. These types of problems can be transformed into a sequence of transportation problems and solved by that algorithm.

J. Activity Analysis Problems

A manufacturer has at his disposal fixed amounts of a number of different resources. These resources, such as raw material, labor, and equipment, can be combined to produce any one of several different commodities or combinations of commodities. The manufacturer knows how much of resource i it takes to produce one unit of commodity j . He also knows how much profit he makes for each unit of commodity j produced. The manufacturer desires to produce that combination of commodities which will maximize the total profit. For this problem, we define the following:

- m = the number of resources
- n = the number of commodities
- a_{ij} = the number of units of resource i required to produce one unit of the commodity j
- b_i = the maximum number of units of resource i available
- c_j = profit per unit of commodity j produced
- x_j = the level of activity (the amount produced) of the j^{th} commodity

The a_{ij} are sometimes called input-output coefficients.

The total amount of the i^{th} resource that is used is given by the linear expression

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n$$

Since this total amount must be less than or equal to the maximum number of units of the i^{th} resource available, we have, for each i , a linear inequality of the form

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n \leq b_i$$

As a negative x_j has no appropriate interpretation, we require that all $x_j \geq 0$. The profit derived from producing x_j units of the j^{th} commodity is

given by $c_j x_j$. This formulation yields the problem of maximizing the profit function

$$c_1 x_1 + c_2 x_2 + \dots + c_n x_n$$

subject to the conditions

$$\begin{aligned} x_1 &\geq 0 \\ x_2 &\geq 0 \\ &\vdots \\ x_n &\geq 0 \end{aligned}$$

and

$$\begin{aligned} a_{11} x_1 + a_{12} x_2 + \dots + a_{1n} x_n &\leq b_1 \\ a_{21} x_1 + a_{22} x_2 + \dots + a_{2n} x_n &\leq b_2 \\ \dots &\dots \\ a_{m1} x_1 + a_{m2} x_2 + \dots + a_{mn} x_n &\leq b_m \end{aligned}$$

These types of problems are used in the field of economics in discussions on theory of the firm and in inter-industry (input-output) analysis.

K. The Diet Problem

Here we are given the nutrient content of a number of different foods. For example, we might know how many milligrams of phosphorus or iron are contained in one ounce of each food being considered. We are also given the minimum daily requirement for each nutrient. Since we know the cost per ounce of food, the problem is to determine the diet that satisfies the minimum daily requirements and is also the minimum cost diet. Define

- m = the number of nutrients
- n = the number of foods
- a_{ij} = the number of milligrams of the i^{th} nutrient in one ounce of the j^{th} food
- b_i = the minimum number of milligrams of the i^{th} nutrient required
- c_j = the cost per ounce of the j^{th} food
- x_j = the number of ounces of the j^{th} food to be purchased ($x_j \geq 0$)

The total amount of the i^{th} nutrient contained in all the purchased food is given by

$$a_{i1} x_1 + a_{i2} x_2 + \dots + a_{in} x_n$$

Since this total amount must be greater than or equal to the minimum daily requirement of the i^{th} nutrient, this linear programming problem involves minimizing the cost function

$$c_1 x_1 + c_2 x_2 + \dots + c_n x_n$$

subject to the conditions

$$\begin{aligned} x_1 &\geq 0 \\ x_2 &\geq 0 \\ &\vdots \\ x_n &\geq 0 \end{aligned}$$

and

$$\begin{aligned} a_{11} x_1 + a_{12} x_2 + \dots + a_{1n} x_n &\geq b_1 \\ a_{21} x_1 + a_{22} x_2 + \dots + a_{2n} x_n &\geq b_2 \\ \vdots &\vdots \\ a_{m1} x_1 + a_{m2} x_2 + \dots + a_{mn} x_n &\geq b_m \end{aligned}$$

Although questionable when applied to diets for human consumption, this formulation has proved highly satisfactory for the evaluation of diets for cattle and chickens.

L. Blending Problem

Blending problems refer to situations where a number of components are mixed together to yield one or more products. There are restrictions on the available quantities of raw materials, restrictions on the quality of the products, and restrictions on the quantities of the products to be produced. There are usually infinitely many different ways in which the raw materials can be blended to form the final products while satisfying the various constraints. It is desired to carry out the blending operation so that the given objective function can be optimized. Blending problems are found in the making of petroleum (blending of gasolines), paint, steel, etc. As a description of the mathematical model would call for too specific a discussion of a particular process, and hence, is beyond the scope of this presentation, we refer the interested reader to the bibliography.

M. Production Scheduling

A manufacturer knows that he has to produce r_i ($i = 1, \dots, n$) items of a certain commodity during the next n months. They can be produced either in regular time, subject to a ceiling of a_i per month, or in overtime, subject to a ceiling of b_i per month. The cost of one item produced in the i^{th} month is c_i on regular time and d_i on overtime. The variation of cost with time and also the capacity restrictions might make it more economical to produce in advance of the period when the items are actually needed. Storage cost is assumed to be s_i per item in each month. We wish to determine the production schedule which minimizes the sum of production and storage costs. Although this problem can be formulated as a standard linear programming model, it can also be formulated in terms of a transportation problem and solved by that algorithm.

N. Smooth Patterns of Production

A manufacturer of an item must determine his monthly production schedule for the next n months. The demand for his product fluctuates, but he must always meet monthly requirements as given to him by his sales forecast or advance orders. He can fulfill the individual demands either by producing the desired amount during the month or by producing part of the desired amount and making up the difference by using the overproduction from previous months.

In general, any such scheduling problem has many different schedules that will satisfy the requirements. For example, the manufacturer could produce each month the exact number of units required by the sales forecast. However, since a fluctuating production schedule is costly to maintain, this type of production schedule is not an efficient one. On the other hand, the manufacturer faced with fluctuating requirements could overproduce in periods of low requirements, store the surplus, and use the excess in periods of high requirements. The production pattern can thus be made quite stable. However, because of the cost of keeping a manufactured item in storage, such a solution may be undesirable if it yields comparatively large monthly surpluses. For this problem, we wish to determine a production schedule that minimizes the sum of the costs due to output fluctuations and to inventories.

Letting

x_t = the production in month t
 r_t = the requirements in month t
 s_t = the storage in month t

The linear programming formulation becomes

$$a \sum y_t + b \sum s_t$$

subject to

$$(1) \quad x_t + s_{t-1} - s_t = r_t$$

$$(2) \quad x_t - x_{t-1} - y_t + z_t = 0$$

where (x_t, s_t, y_t, z_t) ≥ 0 ; (1) states that the amount produced in month t plus the previous month's storage equals the requirements and the storage in month t ; (2) states that the difference between the production in month t and the production in month $(t-1)$ can be represented as the difference between two non-negative numbers with y_t representing an increase in production and z_t a decrease; a is the cost of a unit of increase in production and b is the cost of storing a unit for one month.

O. Caterer's Problem

A caterer knows that, in connection with the meals he has arranged to serve during the next n days, he will need r_j (≥ 0) fresh napkins on the j^{th} day, with $j = 1, 2, \dots, n$. Laundering normally takes p days -- that is, a soiled napkin sent for laundering immediately after use on the j^{th} day is returned in time to be used again on the $(j + p)^{\text{th}}$ day. However, the laundry also has a higher-cost service which returns the napkins in $q < p$ days (p and q integers). Having no usable napkins on hand or in the laundry, the caterer will meet his early needs by purchasing napkins at a cents each. Laundering expense is b and c cents per napkin for the normal and high-cost service, respectively. How does he arrange matters to meet his needs and minimize his outlays for the n days?

In its original formulation, the caterer's problem dealt with aircraft engine overhaul facilities to which the engines can be sent by land (slow service) or air (fast service). This problem can also be formulated as a transportation problem.

P. Warehouse Problem

Given a warehouse with fixed capacity and initial stock of a certain product, which is subject to known seasonal price and cost variations, and given a delay between the purchasing and the receiving of the product, what is the pattern of purchasing, storage, and sales which maximizes profit over a given period of time?

Q. Other Applications

A number of other applications can sometimes be solved by linear programming methods, but are more commonly solved by specialized techniques or algorithms. These include the trim loss problem, traveling salesman problem, critical path scheduling, certain structural design problems and zero-sum two-person games.

BIBLIOGRAPHY

- Charnes, A., and W. W. Cooper, Management Models and Industrial Application of Linear Programming, John Wiley & Sons, Inc., New York, 1960.
- Charnes, A., W. W. Cooper, and A. Henderson, Introduction to Linear Programming, John Wiley & Sons, Inc., New York, 1953.
- Dantzig, G. B., On Integer and Partial Integer Linear Programming Problems, RAND Report P-1410, The RAND Corporation, Santa Monica, Calif., 1958.
- Dantzig, G. B., Linear Programming, Princeton University Press, 1962.
- Dantzig, G. B., and P. A. Wolfe, A Decomposition Principle for Linear Programs, RAND Report P-1544, The RAND Corporation, Santa Monica, Calif., 1959; also in Operations Research, Vol. 8, No. 1, 1959.
- Dennis, J. B., Mathematical Programming and Electrical Networks, John Wiley & Sons, Inc., New York, 1959.
- Dorfman, R., P. A. Samuelson, and R. Solow, Linear Programming and Economic Analysis, McGraw-Hill Book Company, Inc., New York, 1958.
- Eisemann, K., and J. R. Lourie, The Machine Loading Problem, IBM Applications Library, New York, 1959.

Ferguson, Robert O., and L. F. Sargent, Linear Programming, McGraw-Hill Book Company, New York, 1958.

Ford, L. R., Jr., and D. R. Fulkerson, Flows in Networks, Princeton University Press, 1962.

Gale, D., The Theory of Linear Economic Models, McGraw-Hill Book Company, New York, 1960.

Garvin, W. W., Introduction to Linear Programming, McGraw-Hill Book Company, New York, 1960.

Gass, S. I., Linear Programming: Methods and Applications, McGraw-Hill Book Company, 1958.

Gass, S. I., "Recent Developments in Linear Programming," chapter in Advances in Computers, Vol. 2, Academic Press, 1961.

Gomory, R. E., "All-Integer Programming Algorithm," Report RC-189, IBM Research Center, Mohansic, New York, 1960

Gomory, R. E., "An Algorithm for the Mixed Integer Problem," RM-2597, The RAND Corporation, Santa Monica, Calif., 1960.

Hadley, G., Linear Programming, Addison Wesley, Reading, Massachusetts, 1962.

Kelley, J. E., Jr., "Parametric Programming and Primal-Dual Algorithm," Operations Research, Vol. 7, No. 3, 1959.

Koopmans, T. C. (ed.), Activity Analysis of Production and Allocation, Cowles Commission Monograph 13, John Wiley & Sons, Inc., New York, 1951.

Kuhn, H. W., and A. W. Tucker, "Linear Inequalities and Related Systems", Annals of Mathematics Studies 38, Princeton University Press, Princeton, New Jersey, 1956.

Luce, R. D., and H. Raiffa, Games and Decisions, John Wiley & Sons, Inc., New York, 1957.

Meisels, Kurth, A Primer of Linear Programming, New York University Press, 1962.

Orchard-Hays, W., Matrices Elimination & The Simple Method, C-E-I-R, Inc., Revised Edition, October 1961.

Riley, V., and S. I. Gass, Bibliography on Linear Programming and Related Techniques, Johns Hopkins Press, Baltimore, 1958.

Vajda, S., Mathematical Programming, Addison-Wesley, Reading, Massachusetts, 1958.

Section	Area	Page
05	05	14

Vajda, S. , Readings in Linear Programming, John Wiley and Sons, Inc. , New York, 1958.
Wagner, H. M. , "A Practical Guide to the Dual Theorem", Operations Research, Vol. 6, No. 3, 1958.

Wolfe, P. , Linear Programming and Recent Extensions, RAND Symposium on Mathematical Programming, The RAND Corporation, Santa Monica, Calif.

MODULAR PROGRAMMING

Modular programming is used to divide the problem solution into its logical parts or routines so that each routine may be programmed independently. It enables complex problems to be divided into many simple sections. A building block program is thereby created which is controlled by a single routine commonly known as the "main line".

A modular program utilizes the same communication system as established by an organizational chart. Work assignment decisions are made by the main line routine which is not concerned with the functions of the processing routines. If for some reason a routine is revised or eliminated, other processing routines within the program are not affected. However, a segment of the main line might be changed.

There are three primary design criteria of modular programming: ease of understanding, ease of program modification, standardization of program construction.

To prepare and use an operational program effectively and efficiently, the user must be able to understand the content of the program readily. Ease of understanding is provided in the following three ways:

1. Modular flowcharts. -- A modular system flowchart gives an overall picture of the major component elements and structure of the routine; program flowcharts then progress to any desired level of detail, depending upon the complexity of the routine. The program coding is referenced throughout.

2. Detailed narrative of each routine. -- The narrative of each routine states the purpose of the routine, describes the data processed by the routine, and explains each step of the program logic as portrayed by the modular flowchart of the routine.

3. Programming conventions. -- The use of standard labeling conventions, and standard program documentation techniques enables a person unfamiliar with the program to readily understand the program content.

All operational programs require modification and change during their useful life. Ease of program modification is of cardinal importance when the program must be converted to fit a specific new situation. This may be because of changing company policy, varied environmental parameters, or different management objectives. The designer of a program then has the problem of creating a program which can be adjusted to each specific situation. There are two ways of handling the problem.

First, try to anticipate every type of special situation that might be encountered and write a set of routines to handle each situation. This would require a fantastic ability to forecast the future and would lead to slow, cumbersome programs.

The other alternative is to create a program that can be quickly understood and easily modified to reflect changing conditions. Modular programming aims to accomplish the latter alternative.

Once again, the customer may more readily prepare and more quickly implement an operational program if all the runs (programs) within the application adhere to a standardized construction. As indicated above, the logical structure of the program must be such that modifications and additions can be easily made.

Consider the problem of multiple routines -- for instance, three order quantity routines. The normal method of lumping these three routines into a program necessitates setting switches to tell the program which routine to execute at a given time. Any attempt to modify one of the existing routines necessitates trying to extract the routine, patching up the holes in the flow of the program created by the changes, and then fitting the modified routine back in. Anyone who has ever tried to modify a program written by someone else knows how difficult it is to dissect and patch another person's logic if the routines are intertwined.

Using modular programming, each routine is a separate entity. The main line routine provides the master control that ties all the individual processing routines together and coordinates their activity.

Modification of routines is simplified. Furthermore, adding new routines is accomplished by simply expanding the main line routine to transfer control to the new routine in the proper sequence.

MODULAR PROGRAMMING CONVENTIONS

Modularity is accomplished by employing the following conventions:

1. The Main Line
 - a. The main line routine makes all decisions governing the flow of data to the proper processing routines.
 - b. No processing routine can direct data flow to another processing routine.
 - c. Input and output functions which are common to more than one processing routine are controlled by the main line routines.

d. All common areas are defined as part of the main program.

2. Processing Routines

a. A separate processing routine is created for each logical segment of the program. It should accomplish one task in its totality.

b. Each processing routine is complete within itself, with its own defined areas when such areas are for the exclusive use of that routine.

No decision made outside the segment should determine the processing within a segment, and likewise, no decision within a segment should determine the processing outside the segment.

c. Each routine is designed so that it is, in effect, a closed (out-of-line) subroutine. Control is transferred to the processing routine from the main line routine, and when the routine has performed its function, it sends control back to the main line routine.

Entrance to and exit from the routine is never dependent on a particular preceding or trailing segment.

d. A processing routine may transfer control to a multiple-use subroutine. When that routine has performed its function, it transfers control back to the processing routine.

e. Input or output functions that affect only one processing routine may be performed by that routine. All segments contain their own house-keeping to assure noninterference with other segments.

f. A debugging aid which is sometimes useful is the inclusion of program halts at the exit of processing routines. During testing, the halt indicates that a particular processing routine has been executed. After the routine is checked out, the halt is changed to a no-operation or branch. The insertion of halts into the program at strategic points may also be used to bypass the testing of particular routines. (Branches might be preferable.) Action to be taken regarding such program halts must be known and documented before the testing session. This technique tends to use a goodly amount of test time.

3. Multiple-Use Subroutines

- a. If the same sequence of instructions is used by two or more processing routines, these instructions should be established as a multiple-use closed (out-of-line) subroutine.
- b. The proper calling sequence (subroutine linkage) should be established in order that control may be transferred back to the processing routine.
- c. A multiple-use subroutine must be well documented for the purpose of program modification. Comments cards should be used to indicate which processing routines call upon each multiple-use routine and to document the linkage established.

DESIGNING A RUN

To design a modular program, determine the program parameters. List the requirements, elements and functions of the program as they come to mind, giving no attention to logical order.

Once the parameters have been set down, reviewed and revised, determine the logical order of the processing routines, and design the main line of the program. The main line should be so constructed that the largest volume of data is processed by the lowest number of instructions -- that is, in the fastest possible way. A speedy main line contributes greatly to the throughput capabilities of a program.

Once the logic of the main line has been established, the page one overall big-picture system flowchart is drawn. Careful attention should be given to this diagram because it will tend to reveal most errors in logic.

The following components are generally found to be present in the main line of typical programs:

1. Beginning of Item. -- Before obtaining a record, it is often necessary to initialize certain switches, counters, areas, index registers, and instructions. Generally, fewer instructions are required to initialize before entering a routine than after exiting from it, since routines commonly have several exits.
2. Obtain the Item. -- This segment of the run gets the record, sequence-checks the file, and updates the input controls.
3. Process the Item. -- The processing of the record is accomplished. The main line transfers control to the proper processing routines in the proper sequence.

4. End of Item. -- Generally, there are a few instructions to be executed just before disposing of a record. The instructions associated with the clean-up work for the present record should not be confused with initialization for the next record.

5. Dispose of the Item. -- This segment of the run generally puts the record in an output file, updates the output controls and transfers the program to the beginning-of-item routine to start the loop once more.

Use the modular technique within a block whenever it simplifies the logic of the processing routine. Each routine should be as efficient as possible. Look for opportunities to consolidate several in-line routines into one multiple-use subroutine. While sophisticated programming techniques are acceptable, the particular degree of skill and knowledge available to maintain and modify the program should be kept in mind.

The following suggestions may help when programming and documenting:

1. List the functions of the routine.
2. Plan the logic of the routine and sketch a flowchart.

3. Program the routine.

4. Draw the final modular flowcharts of the routine, showing the necessary levels of detail.

5. Create the test data so that every leg of the routine will be thoroughly tested.

6. Write the detailed narrative of the routine.

It is easier to document a routine when the information is fresh in the programmer's mind; furthermore, the documentation thus produced is more meaningful and more comprehensive.

SUMMARY

It has been found that programs written employing the modular programming technique are efficient both from the standpoint of core storage utilization and program execution time. Furthermore, extremely comprehensive and detailed applications, designed and documented with the use of modular techniques, may be readily understood by nonprogram-oriented personnel, ranging from company executives to novice programmers.



COBOL REFERENCE DESCRIPTION

The card below contains the standard IBM COBOL reference information. Individual processor publications will provide the environment information that is not contained on this card.

IBM Reference Card | COBOL

SPECIAL CHARACTERS USED IN THE COBOL SOURCE LANGUAGE		
Name	Character	Card Code
space	(blank)	
plus sign	+	12
minus sign	-	11
hyphen	-	11
multiplication sign	*	11-4-8
check protection symbol	*	11-4-8
division sign	/	0-1
left parenthesis	(0-4-8
right parenthesis)	12-4-8
comma	,	0-3-8
period	.	12-3-8
decimal point	.	12-3-8
dollar sign	\$	11-3-8
equal sign	=	3-8
quotation mark	'	4-8

ARITHMETIC EXPRESSIONS - SEQUENCE OF SYMBOLS					
First Symbol	Second Symbol				
	Variable	* or / or **	+ or -	()
Variable	-	P	P	-	P
* or / or **	P	-	P	P	-
+ or -	P	-	-	P	-
(P	-	P	P	-
)	-	P	P	-	P

CONDITIONAL EXPRESSIONS SEQUENCE OF SYMBOLS					
First Symbol	Second Symbol				
	C	OR	AND	NOT	()
C	-	P	P	-	P
OR	P	-	-	P	P
AND	P	-	-	P	P
NOT	P†	-	-	-	P
(P	-	-	P	P
)	-	P	P	-	P

† Permissible only if the condition itself does not contain a NOT.

ARITHMETIC OPERATORS	RELATIONAL OPERATORS	
OP Meaning	Long Form	Key
+ addition	IS GREATER THAN	GREATER
- subtraction or negation	IS EQUAL TO	EQUAL TO or =
* multiplication	IS LESS THAN	LESS
/ division	IS NOT GREATER THAN	NOT GREATER
** exponentiation	IS NOT EQUAL TO	NOT EQUAL TO or NOT =
	IS NOT LESS THAN	NOT LESS

DATA-ITEM PICTORIAL CHARACTERS		
(n) pictorial character repeated n times		
Numeric		
9 any numeric character		S operational sign
V assumed decimal point		P scale factor
Alphabetic		
A any alphabetic character or space		
Alphanumeric		
X any character		. true decimal point
Z numeric character or space		* numeric character or *
\$ dollar sign †		0 true zero
- minus sign or space †		B true space
+ plus or minus sign †		CR credit symbol
,		DB debit symbol

† These symbols may be either fixed or floating

LIST OF COBOL WORDS		
ACCEPT	Float	PROCEED
ADD	FOR	PROGRAM-ID
ADDRESS	FROM	PROTECT
ADDRESSES		
AFTER	GIVING	QUOTE
ALL	GO	QUOTES
ALPHABETIC	GREATER	
ALPHANUMERIC		READ
ALTER	HIGH-VALUE	RECORD
ALTERNATE	HIGH-VALUES	RECORDING
AN		RECORDS
AND	IDENTIFICATION	REDEFINES
APPLY	IF	REEL
ARE	IN	REELS
AREA	INPUT	REMARKS
AREAS	INPUT-OUTPUT	RENAMING
ASSIGN	INSTALLATION	REPLACING
AT	INTO	RERUN
AUTHOR	IO-CONTROL	RESERVE
AUTHORS	IS	REWIND
		RIGHT
BLANK	JUSTIFIED	ROUNDED
BLOCK		RUN
BLOCKS		
BY	LABEL	
	LEADING	SECTION
CHARACTER	LEAVING	SECURITY
CHARACTERS	LEFT	SELECT
CHECK	LESS	SENTENCE
CLASS	LIBRARY	SIGN
CLOSE	LOCATION	SIGNED
COBOL	LOCK	SIZE
COMPUTATIONAL	LOW-VALUE	SOURCE-COMPUTER
COMPUTE	LOW-VALUES	SPACE
CONFIGURATION		SPACES
CONSTANT	MEMORY	SPECIAL-NAMES
CONTAIN	MODE	STANDARD
CONTAINS	MOVE	STATUS
COPY	MULTIPLE	STOP
CORRESPONDING	MULTIPLY	SUBTRACT
		SUPPRESS
		SYNCHRONIZED
DATA	NEGATIVE	
DATE-COMPILED	NEXT	
DATE-WRITTEN	NO	TALLY
DEPENDENT	NOT	TALLYING
DIGIT	NOTE	THAN
DIGITS	NUMERIC	THEN
DISPLAY		THROUGH
DIVIDE	OBJECT-COMPUTER	THRU
DIVISION	OBJECT-PROGRAM	TIME
DOLLAR	OCCURS	TIMES
	OF	TO
ELSE	OFF	UNTIL
END	OMITTED	UPON
ENTER	ON	USAGE
ENVIRONMENT	OPEN	
EQUAL	OPTIONAL	VALUE
ERROR	OR	VARYING
EVERY	OTHERWISE	
EXAMINE	OUTPUT	WHEN
EXIT	PERFORM	WITH
	PICTURE	WORDS
FD	PLACE	WORKING-STORAGE
FILE	PLACES	WRITE
FILE-CONTROL	POINT	ZERO
FILLER	POSITIVE	ZEROS
FIRST	PROCEDURE	ZEROS

Printed in U.S.A. X28-1520

Underscored words are key words; words not underscored are optional words; brackets, [], enclose words and phrases which may be used or omitted as required; braces, { }, are used to indicate a choice of two or more variant forms.

PAGE SERIAL - A B
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 IDENT.

PROCEDURE DIVISION — BASIC FORMATS

ACCEPT *data-name* [FROM *mnemonic-name*]

ADD $\left\{ \begin{array}{l} \textit{data-name-1} \\ \textit{literal-1} \end{array} \right\}$ [$\left\{ \begin{array}{l} \textit{data-name-2} \\ \textit{literal-2} \end{array} \right\} \dots$] [TO GIVING *data-name-n*]
 [ROUNDED] [ON SIZE ERROR *any imperative statement*]

ALTER *procedure-name-1* TO PROCEED TO *procedure-name-2*
 [*procedure-name-3* TO PROCEED TO *procedure-name-4* . . .]

CLOSE *file-name-1* [REEL] [WITH $\left\{ \begin{array}{l} \textit{LOCK} \\ \textit{NO REWIND} \end{array} \right\}$] [*file-name-2* . . .]

COMPUTE *data-name-1* [ROUNDED] = *arithmetic expression*
 [ON SIZE ERROR *any imperative statement*]

DISPLAY $\left\{ \begin{array}{l} \textit{data-name-1} \\ \textit{literal-1} \end{array} \right\}$ [$\left\{ \begin{array}{l} \textit{data-name-2} \\ \textit{literal-2} \end{array} \right\} \dots$] [UPON *mnemonic-name*]

DIVIDE $\left\{ \begin{array}{l} \textit{data-name-1} \\ \textit{literal-1} \end{array} \right\}$ INTO $\left\{ \begin{array}{l} \textit{data-name-2} \\ \textit{literal-2} \end{array} \right\}$ [GIVING *data-name-3*]
 [ROUNDED] [ON SIZE ERROR *any imperative statement*]

ENTER *language-name*.

EXAMINE *data-name* $\left\{ \begin{array}{l} \textit{TALLYING} \left\{ \begin{array}{l} \textit{ALL} \\ \textit{LEADING} \\ \textit{UNTIL FIRST} \end{array} \right\} \\ \textit{literal-1} \left[\textit{REPLACING BY} \textit{literal-2} \right] \\ \textit{REPLACING} \left\{ \begin{array}{l} \textit{ALL} \\ \textit{LEADING} \\ \textit{UNTIL FIRST} \end{array} \right\} \textit{literal-3} \textit{ BY} \textit{literal-4} \end{array} \right\}$

EXIT.

GO TO [*procedure-name*]

GO TO *procedure-name-1* *procedure-name-2* [*procedure-name-3* . . .]
DEPENDING ON *data-name*

MOVE $\left\{ \begin{array}{l} \textit{data-name-1} \\ \textit{literal} \end{array} \right\}$ TO *data-name-2* [*data-name-3* . . .]

MOVE CORRESPONDING *data-name-1* TO *data-name-2* [*data-name-3* . . .]

MULTIPLY $\left\{ \begin{array}{l} \textit{data-name-1} \\ \textit{literal-1} \end{array} \right\}$ BY $\left\{ \begin{array}{l} \textit{data-name-2} \\ \textit{literal-2} \end{array} \right\}$ [GIVING *data-name-3*]
 [ROUNDED] [ON SIZE ERROR *any imperative statement*]

NOTE *any comment.*

OPEN [INPUT *file-name-1* [*file-name-2* . . .]] [OUTPUT *file-name-3*
 [*file-name-4* . . .]]

PERFORM *procedure-name-1* [THRU *procedure-name-2*]

PERFORM *procedure-name-1* [THRU *procedure-name-2*]
 $\left\{ \begin{array}{l} \textit{integer-1} \\ \textit{data-name-1} \end{array} \right\}$ TIMES

PERFORM *procedure-name-1* [THRU *procedure-name-2*]
UNTIL *condition-1*

PERFORM *procedure-name-1* [THRU *procedure-name-2*]
VARYING *data-name-1* FROM $\left\{ \begin{array}{l} \textit{numeric-literal-1} \\ \textit{data-name-2} \end{array} \right\}$
BY $\left\{ \begin{array}{l} \textit{numeric-literal-2} \\ \textit{data-name-3} \end{array} \right\}$ UNTIL *condition-1*

PERFORM *procedure-name-1* [THRU *procedure-name-2*]
VARYING *subscript-name-1* FROM $\left\{ \begin{array}{l} \textit{integer-1} \\ \textit{data-name-1} \end{array} \right\}$ BY $\left\{ \begin{array}{l} \textit{integer-2} \\ \textit{data-name-2} \end{array} \right\}$
UNTIL *condition-1* [AFTER *subscript-name-2* FROM $\left\{ \begin{array}{l} \textit{integer-3} \\ \textit{data-name-3} \end{array} \right\}$]
BY $\left\{ \begin{array}{l} \textit{integer-4} \\ \textit{data-name-4} \end{array} \right\}$ UNTIL *condition-2* [AFTER *subscript-name-3*]
FROM $\left\{ \begin{array}{l} \textit{integer-5} \\ \textit{data-name-5} \end{array} \right\}$ BY $\left\{ \begin{array}{l} \textit{integer-6} \\ \textit{data-name-6} \end{array} \right\}$ UNTIL *condition-3*]

READ *file-name* RECORD [INTO *area-name*]
 [AT END *any imperative statement*]

STOP $\left\{ \begin{array}{l} \textit{literal} \\ \textit{RUN} \end{array} \right\}$

SUBTRACT $\left\{ \begin{array}{l} \textit{data-name-1} \\ \textit{literal-1} \end{array} \right\}$ [$\left\{ \begin{array}{l} \textit{data-name-2} \\ \textit{literal-2} \end{array} \right\} \dots$]
FROM $\left\{ \begin{array}{l} \textit{data-name-n} \\ \textit{literal-n} \end{array} \right\}$ [GIVING *data-name-m*]
 [ROUNDED] [ON SIZE ERROR *any imperative statement*]

WRITE *record-name* [FROM *area-name*]

IDENTIFICATION DIVISION — BASIC FORMATSPROGRAM-ID. *program-name.*[AUTHOR. *author-name.*][INSTALLATION. *any sentence or group of sentences.*][DATE-WRITTEN. *any sentence or group of sentences.*][DATE-COMPILED. *any sentence or group of sentences.*][SECURITY. *any sentence or group of sentences.*][REMARKS. *any sentence or group of sentences.*]ENVIRONMENT DIVISION — BASIC FORMATS

See individual processor publications.

DATA DIVISION — BASIC FORMATS

File Description

FD *file-name* COPY *library-name.*FD *file-name* [RECORDING MODE IS *mode*][BLOCK CONTAINS *integer-1* {RECORDS|CHARACTER(S)}][RECORD CONTAINS [*integer-2* TO] *integer-3* CHARACTER(S)]

LABEL RECORDS {ARE|IS} {STANDARD|OMITTED}

[VALUE OF *data-name-1* IS *literal* [*data-name-2* IS . . .]]DATA RECORDS {ARE|IS} *data-name-3* [*data-name-4* . . .]

Record Description

level-number data-name-1 [REDEFINES *data-name-2*]COPY *data-name-3* [FROM LIBRARY]*level-number* {FILLER|*data-name-1*} [REDEFINES *data-name-2*][SIZE IS *integer-1* {CHARACTER(S)|DIGIT(S)}][USAGE IS {COMPUTATIONAL|DISPLAY}] [OCCURS *integer-2* TIME(S)]

[SIGNED] [SYNCHRONIZED {LEFT|RIGHT}] [POINT LOCATION

IS {LEFT|RIGHT} *integer-3* PLACE(S)] [CLASS IS {ALPHABETIC|NUMERIC|ALPHANUMERIC|AN}][PICTURE IS *any allowable combination of characters and symbols as described in Chapter 6*]

[JUSTIFIED {LEFT|RIGHT}] {ZERO SUPPRESS|CHECK PROTECT|FLOAT DOLLAR SIGN}

[LEAVING *integer-4* PLACE(S)] [BLANK WHEN ZERO][VALUE IS *literal*]



AVAILABLE FORTRAN STATEMENTS

The following two tables indicate, by means of an X, which of 70 possible FORTRAN statements are valid for systems in IBM's product line.

FORTRAN I was the name given to the original 704 FORTRAN and included the first 33 X's under the column headed "704 FORTRAN II". The name now refers to any FORTRAN system that includes these items.

FORTRAN II is FORTRAN I plus CALL, RETURN, FUNCTION, SUBROUTINE, and COMMON. In 7070, FORTRAN II is referred to as FULL FORTRAN.

FORTRAN III was a modification of 704 FORTRAN II by IBM Programming Research. It includes:

1. Library use at load time
2. Additional format features
3. Boolean expressions.
4. Subroutine and function names as arguments
5. Symbolic language

These items were released by Applied Programming as a modification to 704 FORTRAN II. Items 1 and 5 above were dropped from 709/7090 FORTRAN. Item 3 will be dropped in 7090 FORTRAN and item 5 will be handled by the statement EXTERNAL in 7090 FORTRAN.

FORTRAN IV is the name given to the latest 709/7090 FORTRAN. A slightly smaller subset holds for the 7040/44.

X - Part of FORTRAN on this machine	1620 Basic	1620 FORTRAN II	1401	1410/7010	7010 Monitor	7070/74 Basic	7070/74 Full	705	7080	704 FORTRAN II	709/90 FORTRAN II	709/90 FORTRAN II	650	7040/44
A = B	X	X	X	X	X	X	X	X	X	X	X	X	X	X
GO TO n	X	X	X	X	X	X	X	X	X	X	X	X	X	X
GO TO (n ₁ , n ₂ , n ₃ , . . .), i	X	X	X	X	X	X	X	X	X	X	X	X	X	X
ASSIGN i TO n				X		X	X	X	X	X	X	X		X
GO TO n, (n ₁ , n ₂ , . . .)				X	X	X	X	X	X	X	X	X		X
IF (a) n ₁ , n ₂ , n ₃	X	X	X	X		X	X	X	X	X	X	X	X	X
IF (t) s					X								X	X
SENSE LIGHT i			X	X		X	X	X	X	X	X			
CALL SLITE (i)					X							X		X
IF (SENSE LIGHT i) n ₁ , n ₂			X	X		X	X	X	X	X	X			
CALL SLITET (i, j)					X							X		X
IF (SENSE SWITCH i) n ₁ , n ₂	X	X	X	X		X	X	X	X	X	X			
CALL SSWTCH (i, j)					X							X		X
IF ACCUMULATOR OVERFLOW n ₁ , n ₂				X		X	X	X	X	X	X			
IF QUOTIENT OVERFLOW n ₁ , n ₂				X		X	X	X	X	X	X			
CALL OVERFL (j)					X							X		X
IF DIVIDE CHECK n ₁ , n ₂				X		X	X	X	X	X	X			
CALL DVCHK (j)					X							X		X
PAUSE	X	X	X	X	X	X	X	X	X	X	X	X	X	X
STOP	X	X	X	X	X	X	X	X	X	X	X	X	X	X
DO n i = m ₁ , m ₂ , m ₃	X	X	X	X	X	X	X	X	X	X	X	X	X	X
CONTINUE	X	X	X	X	X	X	X	X	X	X	X	X	X	X
END	X	X	X	X	X	X	X	X	X	X	X	X	X	X
READ n, List	X	X	X	X		X	X	X	X	X	X	X	X	X
READ INPUT TAPE i, n, List			X	X		X	X	X	X	X	X			
READ (i, n) List					X				X			X		X
PUNCH n, List	X	X	X	X		X	X	X	X	X	X	X	X	X
PRINT n, List	X	X	X	X		X	X	X	X	X	X	X		X
WRITE OUTPUT TAPE i, n, List			X	X		X	X	X	X	X	X			
WRITE (i, n) List					X				X			X		X
READ TAPE i, List			X	X		X	X	X	X	X	X			
READ (i) List					X				X			X		X
WRITE TAPE i, List			X	X		X	X	X	X	X	X			
WRITE (i) List					X				X			X		X
READ DRUM i, j, List										X				

	1620 Basic	1620 FORTRAN II	1401	1410/7010	7010 Monitor	7070/74 Basic	7070/74 Full	705	7080	704 FORTRAN II	709/90 FORTRAN II	709/90 FORTRAN IV	650	7040/44
X - Part of FORTRAN on this machine														
1 - Feature of later versions of 7080 FORTRAN														
2 - 1410-7010 20K systems only														
3 - Not in 40K systems														
WRITE DRUM i, j, List										X				
END FILE i			X	X	X	X	X	X	X	X	X	X		X
REWIND i			X	X	X	X	X	X	X	X	X	X		X
BACKSPACE i			X	X	X	X	X	X	X	X	X	X		X
TYPE n, List	X	X		3		X	X		X					
RECORD (i) List				2										
FETCH (i) List				2										
FIND (a)				2										
ACCEPT n, List	X	X												
ACCEPT TAPE n, List	X	X												
PUNCH TAPE n, List	X	X												
FORMAT	X	X	X	X	X	X	X	X	X	X	X	X		X
DIMENSION	X	X	X	X	X	X	X	X	X	X	X	X	X	X
FREQUENCY				X						X	X			
EQUIVALENCE		X	X	X	X	X	X		1	X	X	X		X
CALL		X		X	X		X		1	X	X	X		X
RETURN		X		X	X		X		1	X	X	X		X
DATA												X		
BLOCK DATA												X		
COMMON		X		X	X		X		1	X	X	X		X
EXTERNAL					X							X		X
LOGICAL												X		X
REAL					X							X		X
DEFINE FILE (n ₁ , n ₂)				2										
DOUBLE PRECISION												X		
INTEGER					X							X		X
COMPLEX												X		
SUBROUTINE name (a ₁ , a ₂ , . . .)		X		X	X		X		1	X	X	X		X
FUNCTION name (a ₁ , a ₂ , . . .)		X		X	X		X		1	X	X	X		X
REAL FUNCTION name (a ₁ , a ₂ , . . .)												X		X
INTEGER FUNCTION name (a ₁ , a ₂ , . . .)												X		X
DOUBLE PRECISION FUNCTION name (a ₁ , a ₂ , . . .)												X		
COMPLEX FUNCTION name (a ₁ , a ₂ , . . .)												X		
LOGICAL FUNCTION name (a ₁ , a ₂ , . . .)												X		X
CALL (EXIT)					X							X		X

PROGRAMMING SYSTEMS CONCEPTS

Absolute or machine code. -- Some combination of alphabetic, numerical and special symbols (;, . # / &, etc.) constitute the notation codes for both instructions and storage addresses for communicating with electronic computers. These absolute codes are assigned by engineers as the vocabulary of a designated system. The 1401 system, for example, will accept as operating instructions only the absolute code designed for the 1401.

A single machine instruction designates an operation code and usually one or more addresses where data is to be operated upon. In absolute coding, each instruction is written on a line and the next instruction of a sequence is written on the next line. Such machine-language coding is often referred to as straight-line coding.

Source program. -- A program coded in other than machine language which must be translated into machine language before use.

Object program. -- A program resulting from the translation of a source program by a processor. The object program is ready for use by a computer to process data.

Processor. -- A machine-language program which performs the functions necessary to convert a source program into the desired object program.

Symbolic language. -- Any collection of mnemonic symbols used in programming to represent operation codes, functions and/or addresses, with rules of usage.

Programming system. -- Any method of programming problems, other than machine language, consisting of a language and its associated processor(s).

Assembly program. -- A machine program which substitutes machine-language instructions for symbolic instructions, assigns storage locations and performs other activities necessary to produce the final object program.

Macro instruction. -- A method of describing, with a symbolic language statement, a function to be performed by the object program. This function will normally involve more than one machine-language instruction. A macro instruction does not

indicate how the function is to be performed by the object program; it merely indicates logically what is to be done at this point in the program.

Macro generator. -- The segment of a compiler program which interprets a macro instruction and develops the instruction sequence required to perform the specified function.

Compiler. -- A processor which will accept the macro instructions of a symbolic program and translate them either directly to machine code or into one-for-one symbolic instructions (using one or more of the following methods: inserting subroutines, linking to subroutines, generation) and then pass the entire set of instructions to an assembly program for final translation into machine language.

Generator. -- A machine-language program used during compiling to produce symbolically coded (one-for-one) instructions which will perform the operation called for by the symbolic coding of the source program.

Symbolic programming systems. -- Programming systems allow the programmer to assign his choice of names (with such restrictions as a limitation on the number of characters and a requirement that the first character be alphabetic) to data so that these names represent storage locations. Such assigned names, which are usually mnemonic, together with the defined set of mnemonic operation codes and macro instructions, are symbolic programming systems (SPS).

Routine. -- A sequence of machine instructions which carry out a defined function.

Subroutine. -- A sequence of instructions to perform a specified operation. Some subroutines are parts of a program which are used repetitively within the same program -- discount calculation, for example, for each transaction item in a billing operation. Others, such as end-of-job, may be used only once in a given program but they can be used in more than one application area.

Merge. -- A generalized program to combine records from two or more similarly sequenced files into one sequenced file, including all items

from the original files, according to a key contained in the records and specified by the user. The generalized program includes an assignment phase in which the program is specialized in accordance with specifications supplied by the user on control cards.

Sort. -- A generalized program to sequence records from one or more files according to a key contained in the records and specified by the user. The sequence is usually numerical, alphabetic or alphabetic. The generalized program includes an assignment phase in which the program is specialized in accordance with specifications supplied by the user on control cards.

Autocoder. -- An automatic coding system. The term was applied to early techniques of symbolic coding which have been continuously improved and expanded. Advanced Autocoder systems are available for many IBM data processing systems. Other processors, such as COBOL for the 7070, use Autocoder as an intermediate language. Here the COBOL processor develops a program in Autocoder language, which the Autocoder processor then translates into a 7070 object program.

FORTTRAN (FORmula TRANslation). -- Resembles the language of mathematics. FORTRAN permits people who are not trained programmers to prepare programs for a computer. It is especially useful in scientific and technical fields where most problems are expressed in mathematical form. Processors to convert source programs written in FORTRAN are supplied for all IBM data processing systems.

COBOL (COmmon Business Oriented Language). -- A programming language primarily for commercial data processing. It is the result of work by the Conference on Data Systems Languages (CODASYL), a voluntary cooperative effort of a number of users and manufacturers of data processing equipment. The COBOL system is a result of efforts to produce an English-like language which can be used with many different types of data processing systems. Processors to convert source programs written in COBOL are supplied for all commercial data processing systems manufactured by IBM.

Input/output control systems (IOCS). -- Programs designed to relieve the programmer of the necessity for writing input and output routines by automatically handling the preparation and checking of labels, the blocking and unblocking of data words,

and the overlapping of processing with input and output utilizing external storage such as magnetic tape and disk storage.

Report program generators. -- Systems which generate a program to produce desired reports. Using a printer spacing chart, a pictorial representation is prepared as a guide for the input, data and output descriptions and the calculation definition. Input to these programs consists of specifications of the data from which the report is to be made and specifications of the desired report. The output from these generators is a program, in symbolic language, which, when assembled and executed, will produce the desired report.

Utility programs. -- Designed to do repetitive jobs which are common to many computer installations. Unlike subroutines, which are incorporated as integral parts of a program, utility programs are executed independently of the main program. Examples of utility programs are such conversion routines as card-to-tape, disk-to-tape, disk-to-card, etc.; program printout; and tape correction loader (which provides a convenient method of inserting corrections in a program located in core storage).

Monitors. -- Designed to reduce or eliminate manual setup and loading, operator decisions and interventions when many jobs with relatively short run time are involved. A simple monitor accepts a single input tape containing all programs and associated data. The monitor loads a job program into the computer and gives control to that program. The program reads the data which follows, performs its tasks, and writes its output for printing on a common output tape. When the job is complete, it returns control to the monitor, which loads the next program, handles it, and goes on to others. In this manner a complete "stack" of jobs is processed without stopping the computer. Monitors generally include other features, such as job accounting and tape assignments, to further assist the computer operator.

Supervisory systems. -- Serve to link and control other systems.

Simulator. -- Permits the running of machine-language programs of the simulated machine on the simulator machine.

Section	Area	Page
06	04	03

Conversion programs. -- Facilitate the running of programs written for one type of machine on another type.

Interpretive routine. -- Decodes instructions written as pseudo codes and immediately executes those instructions, as contrasted with a compiler which decodes the pseudo codes and produces a machine-language routine to be executed at a later time. Such routines are often referred to as "load-and-go".

Application programs. -- Perform specific tasks, such as demand deposit, lens design, etc. Application programs are usually classified as industry-oriented or industry-independent.

BIBLIOGRAPHY

Systems Reference Library. -- For each IBM data processing system, a Systems Reference Library includes all the respective literature available. Prompt updating of all publications is accomplished by Technical Newsletters.

Catalog of Programs for IBM Data Processing Systems (C20-8090). -- Indexes and abstracts programs available from IBM under one cover.

Programming Systems Concepts (F20-8149). -- Describes the general concepts behind programming systems. Nonmachine-oriented, it does not explain usage methods.



DATA TABLES AND TABLE SEARCHING

Tables and table search or lookup techniques have wide application in computer programming. They can be used as an aid in storing and retrieving data, particularly certain types of data, such as values of mathematical functions or rate information. They can also be used in programming to control program branching, convert data from one notation to another, sort records, obtain addresses of records stored in random files or handle queues of input or output data.

TABLES

A table is a structured array of related information stored in a computer. It has two purposes:

1. To provide a compact format for information representation.
2. To facilitate the processing of information in it.

Table information is expressed in three media: table arguments, table functions and the position of either of these in the table. A table argument is the complete or partial identification of an entity. A table function is the value of a property that is characteristic of some entity. This value may be null. A table function has a computer location but the function is in itself unidentifiable to the process that is to determine its location or existence. Position in the table may be pertinent; it may give the value of a characteristic of an argument or it may identify the item of which a function is characteristic.

There are three basic table forms:

1. A table may be a list of arguments identifying items of a set.
2. A table may be a list of functions. Each function would be the value of a property characteristic of an item of a set.
3. A table may be made up of an argument and a function for each entity represented in it. An argument identifies an item of a set and the corresponding function gives the value of a property of the item.

These basic table forms may be varied by having multiple arguments and/or multiple functions.

Figure 1 shows a table consisting of man numbers, job codes and department numbers. It could be stored in a data processing system in this form without the headings.

Man Number	Job Code	Dept. Number
102	3	13
206	4	11
257	2	11
132	6	12

Figure 1

There are a variety of ways in which this table could be used:

1. Man number could be the table argument identifying an entity, that is, a particular person. The department numbers and job codes would be functions; they describe or measure particular properties that are characteristics of these entities. Locating a man's job code requires two things: his man number and knowledge of the positions of job code and department number relative to man number in the table.

2. Position can be the argument. It can identify an item, a record, for example, that has a certain position. Man number, job code and department number are then all properties of the item.

3. Position can also be a function. If the argument is 257 and identifies a record, then the number 3 is a function. It describes a property of this record, namely its position. If 257 identifies a person, then the function 3 describes, perhaps, his position with respect to other men in some queue.

4. Job code and department number can both be arguments. The numbers 2 and 11 identify an entity (there could be several) that works in department 11 and does job 2. A property of this entity is its man number which is described by the function 257.

Table Arrangement

Tables are generally arranged so that the elements of a set (i. e., all similar arguments or all similar functions) are separated by a constant address interval. The interval can be different for each set or it can be the same for several sets. Examples of these two arrangements are shown in Figure 2.

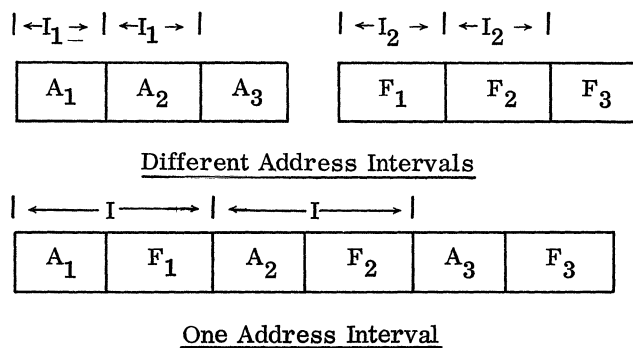


Figure 2

SEARCH ARGUMENT

A search argument is the complete or partial identification of an entity. It is used by a table lookup or search process to determine the table argument or arguments having a specific relationship to it and/or to locate a function associated with it.

Objectives of Table Search Processes

The objectives of searching a list of table arguments with a search argument are generally to:

- Determine the existence or nonexistence of a particular table argument.
- Determine the table argument that is similar, in a certain way or to some degree, to the search argument. The similarity may be measured by the elements or fraction of elements of the search and table arguments that are alike or nearly alike.
- Determine the table argument that has a certain relation to the search argument in terms of collating sequence or numeric value; for example:
 - Equal to
 - Most nearly equal to
 - Less than
 - Greater than
 - Lowest in the table and greater than
 - Highest in the table and less than
 - OR combinations: equal-low, equal-high
- Determine coincidences that exist between the table arguments and the search argument; for example, the number of table arguments that have a certain relation or similarity to the search argument.
- Determine the position of a certain argument in the table.

The objectives of table search processes in tables that include functions may be to locate a function so that it can be retrieved or changed. The location of a function is determined in one of two ways:

1. Directly, with a given or computed function address.
2. Indirectly, by first locating the corresponding table argument and then deriving the function's address from the argument's address.

TABLE SEARCH OR LOOKUP TECHNIQUES

Scanning or Consecutive Search

Scanning is done by beginning at either end of a table and comparing the search argument with consecutive table arguments until the search objective is met or until all arguments have been inspected. This technique, or variations of it, will meet any of the objectives mentioned earlier.

Table arguments are normally ordered in ascending or descending sequence if the search is for an equal-high or equal-low condition. If the table is ordered, the scan should, if practical, be done in the direction that will minimize the number of inspections. This will depend on the distribution of activity in the table.

If there is not a particular reason for ordering a table, it may be arranged so that arguments receiving the greatest use are inspected first.

Assuming that table arguments all have the same frequency of use and that arguments at either end of the table are not exempt from inspections, the following formulas give the number of compares required to scan for an equal condition.

Let M = Maximum inspections required.
 A = Average inspections required.
 N = Number of table arguments.

E1. $M = N$

E2. $A = \frac{N + 1}{2}$

Binary Search (True Midpoint)

A binary search requires that table arguments be ordered in an ascending or a descending sequence and that there is a high, low or equal relation, based on this sequence, between the search argument and each table argument.

Binary searching is done by comparing the search argument with the argument at the midpoint of the table. An unequal compare indicates

whether the segment above or below the table argument tested can be eliminated ("cut") from further search. Next, the midpoint argument of the remaining portion of the table is inspected. An unequal compare indicates another segment of the table that can be excluded. This elimination process continues until a table argument that is equal to the search argument is located or until the table has been segmented to the point where the search argument falls between two specific table arguments.

Figure 3 illustrates a binary search using a search argument of 58. It shows the sequence of the search and the portion of the table that is eliminated by each inspection.

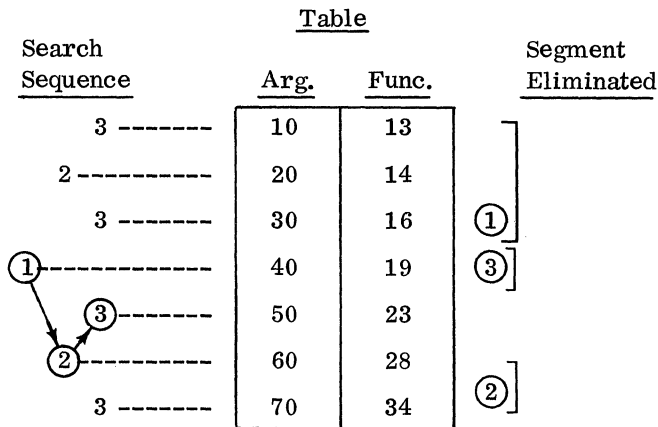


Figure 3

To maintain true midpoint inspections throughout a search the number of table arguments must be such that

E3. $N = 2^n - 1$,
 $N =$ the number of table arguments,
where n is a positive integer.

Values of N for n from 1 to 12 are given in the table in Figure 4.

True midpoint inspection can be maintained when the number of table arguments does not agree with equation E3 by using one of the following schemes:

1. An N that is larger than the number of arguments going into the table is computed from equation E4.

E4. $Z^{n-1} < N < Z^n - 1$ where n is an integer. False arguments with values outside the range of the real arguments are placed at the top and/or bottom of the table to complete it. False arguments that will never be inspected need not appear.

2. An N is obtained as in the last scheme. The addresses of all table arguments are compared with the address limits of the real arguments. An address outside of the limits is not inspected.

3. The table can be segmented into tables that sustain midpoint inspection. For example; tables with $N = 7$ and $N = 3$ could be used to accommodate 10 arguments.

The following applies to binary searches of tables where

$N = 2^n - 1$;

E5. $M = n$.

M inspections will always be made if an equal relation is not found and if the end arguments are not exempt.

When the search will always reveal an equal relation, use

E6. $A = \frac{1}{N} (1 + 2 \times 2^1 + 3 \times 2^2 + \dots + n \times 2^{n-1})$.

Values of A for n up to 12 are given in the table in Figure 4.

R	$N = 2^n - 1$	Average Inspections
1	1	1.00
2	3	1.67
3	7	2.43
4	15	3.27
5	31	4.16
6	63	5.10
7	127	6.06
8	255	7.03
9	511	8.02
10	1023	9.01
11	2047	10.00
12	4095	11.00

Figure 4

Binary Search (Approximate Midpoint)

Another binary search technique is to compare the search argument with the argument that is at the midpoint of the segment of the table still subject to search. If there is not a true midpoint argument, one of the two arguments at the center of the segment is used.

This method negates the requirement that $N = 2^n - 1$. It will, however, be necessary on occasion to adjust addresses by one half of the argument address interval.

The average number of compares to search for an equal relation is:

E7. $A = \frac{[1 + 2 \times 2^1 + 3 \times 2^2 + \dots + (n-1) \times 2^{(n-2)}] + n [N - (2^{n-1} - 1)]}{N}$

E8. $2^{n-1} \leq N \leq 2^n - 1$.

This method usually results in fewer compares than true midpoint search requiring false table arguments or segmentation.

Direct Lookup

Direct lookup does not involve the use of table arguments. It is accomplished by computing the address of the function directly from the search argument. The method is particularly effective when there is a linear relation between the search arguments and the addresses of the functions.

Table Segmenting

Segmenting divides a table into parts. It can be done in several ways. Three of these are interval scan, prescan, and character structure.

Interval scan. Instead of scanning every argument of an ordered list, only arguments of a certain interval are inspected (e. g. , every fourth argument). Interval scan will either locate the desired argument or determine the table segment that contains it.

Prescan. A separate table is made from certain of the arguments (and perhaps their associated functions). This table is scanned first to determine the segment to be searched in the initial table. Prescan can be effective in directing searches to table segments of high activity.

Character structure. Some tables may be segmented on the character structure (i. e. , zone bits). This is an effective way of segmenting tables used in converting characters from one code to another.

Presorted Input

Presorting batched search arguments into the sequence of the scan can improve the efficiency of scanning an ordered table of arguments. The first table argument that needs to be inspected on any scan (except the first) is the argument that was inspected last on the previous scan.

The maximum and average inspections for equal scanning are:

$$E9. M = N + S - 1$$

$$E10. A = \left(\frac{N - S}{S + 1} \right) - \frac{1}{S} + 2$$

where S = the number of search arguments and

N = the number of table arguments.

The number of compares obtained from E9 and E10 are exclusive of the presorting requirements.

Multiple Argument Tables and Searches

Several methods of constructing and searching multiple argument tables are noted below. Figure 5 shows a table that uses two arguments to identify

an entity. For example, Y_3 and X_2 identify item (Y_3X_2) of which the function F_{32} is characteristic. The term "set of arguments" used below means all of the arguments measuring a single dimension or identifier. ($Y_1, Y_2, Y_3 \text{ ---}$) and ($X_1, X_2, X_3 \text{ ---}$) are both sets of arguments.

Y_3	F_{31}	F_{32}	F_{33}
Y_2	F_{21}	F_{22}	F_{23}
Y_1	F_{11}	F_{12}	F_{13}
	X_1	X_2	X_3

Figure 5

Combining Arguments

The arguments of each set are combined with the arguments of every other set to make single arguments. The number of arguments resulting is the product of the numbers of arguments in each set. These arguments (and their functions) can be used in a table as in Figure 6.

Table	
Arg.	Func.
Y_1X_1	F_{11}
Y_1X_2	F_{12}
Y_1X_3	F_{13}
Y_2X_1	F_{21}

Figure 6

Given random input, the average number of compares using this technique is:

$$E11. A = \frac{(N_1 \times N_2 \times \text{---} \times N_k) + 1}{2}$$

where there are k different sets of arguments.

Segmented Tables

A separate table is required for each set of arguments. The position of an argument in one table determines the segment to be searched in another table. Every table has its arguments repeated once for every argument in the preceding table. The number of arguments in the last table is the product of the number in each set. Figure 7 illustrates this method.

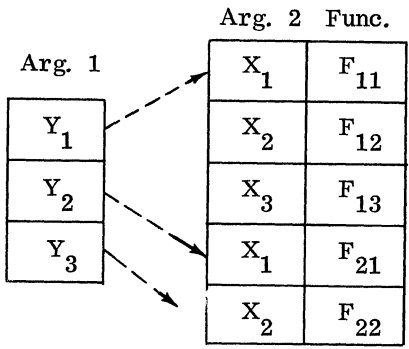


Figure 7

The average number of compares is:

$$E12. A = \frac{N_1 + 1}{2} + \frac{N_2 + 1}{2} + \dots + \frac{N_k + 1}{2}$$

The compares for this method are shorter as well as fewer than for the combining arguments method.

Direct Addressing

A table is made for each set of arguments. Each of these tables is searched so that the actual address of functions can be computed. For example, examining the two argument tables in Figure 8 will determine the column and row of the desired function.

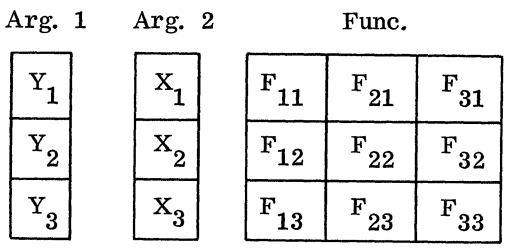


Figure 8

The average number of compares is the same as for the preceding method. This method does not require that the arguments be repeated.

Shortening Compares

The search argument and all of the table arguments with which it is compared will usually be equal in length and contain no unneeded positions. Two ways of shortening the length of the argument compares are mentioned below.

Character or Element Compares

It is not always necessary that the search argument and table arguments be compared over their entire length. A scan of an ordered table can be done by comparing the most significant character position of the search argument with that of the table arguments until an equal is found. The scan continues with the compare on the next-less significant character position. This process continues until the compare is on the least significant character position. The average number of one-character compares is:

$$E13. A = C \frac{(N + 1)}{2} + C - 1$$

where C = the number of characters in the arguments.

The compares can, of course, be on an element consisting of several character positions.

Argument Transformation

Arguments can sometimes be shortened by transforming them from one base to another or by packing numeric information (i. e. , using zone bits). Every search argument that is introduced must undergo the same transformation.

Interpolation

It is frequently necessary to obtain a function for a search argument that is not equal to any table argument but that is within the table range. To do this, the functions of table arguments on either side of the search argument are found, and an arithmetic routine is used to develop an intermediate function.



SYSTEMS EFFICIENCY

Systems efficiency requires minimum job times, the full use of the system's capacity, and the economical investment of programming effort. An analysis of installed and projected systems will provide frequency-time information by program, such as "daily -- three hours". This procedure highlights the time-consuming jobs where program changes may result in a significant reduction in running time.

The following approaches can be considered on the system level.

Short-run, high-frequency programs. Investigate rescheduling these programs for longer, less frequent runs, or for combining them with related or unrelated jobs. Are successive programs linked together?

Long-run programs. Consider exception reporting, cycle reporting, reassigning processing to other programs and rearranging tape or disk files.

Administrative control. Review the time required for operator intervention, paperwork and decisions during program runs. How closely is machine utilization controlled for testing, compiling and assembling programs?

At the individual program level, the three major factors to review are the processing requirements, the input/output requirements, and the relationship to other programs. Generally, the input/output time should be balanced with the process time, taking full advantage of available channels, buffers or overlap time.

Prior to changing an existing operational program, the anticipated increase in efficiency should be weighed against the cost of programming, testing, recompiling and/or reassembling the program.

PROCESS-LIMITED PROGRAMS

The following alternatives can be considered for programs with extensive processing, where reductions in process time will result in a direct reduction of run time.

1. Shift part of the processing to preceding or following programs. The part shifted may include preliminary calculations, table lookup, summarizing, validity checking, and editing.

2. Review moving and clearing operations for efficiency. Is there a better way?

3. Concentrate on main-line or high-usage sections of the program where most processing time is required.

4. Evaluate the effect of file organization and coding structure on process time. Does 80% of the activity involve only 20% of the master file?

5. Will an expanded tape record eliminate processing time? For example, if control field XXYYYYZ is needed on the master file for processing but must be edited to Z-YYY-XX for printing, both formats can be carried to avoid editing.

6. Will additional input/output activity such as printing, punching or preparing a summary tape result in an advantageous net change?

7. Are input, output and work areas being cleared unnecessarily?

8. Consider the time required for index register control of routines as opposed to the advantages of separate routines. Audit control loop logic for brevity. Consider the two methods of loop control shown below.

<u>Method 1</u>	<u>Method 2</u>
Load counter	Load counter
Add 1	Subtract 1
Store counter	Store counter
Compare to limit	Branch if not zero
Branch if low	

Method 2 is at least 25% faster than method 1. In some systems, the use of negative counters is faster due to the time requirements of recompiling.

9. Review the mathematical approach used along with possible alternatives and compare time requirements. For example, the expression $A^2 - B^2 = (A + B)(A - B)$ requires two multiplications and one subtraction on the left, while the right requires one multiplication, one addition and one subtraction. The right half of the expression uses a relatively faster add instruction in place of a multiply operation.

10. Evaluate different table search (i. e., table lookup) techniques such as direct address and binary search. High-activity items could be concentrated on a separate table. Can many tables be combined into one? Can the results of one table search be used to shorten the lookup time in another? A tendency for successive lookups to return to the same table location may suggest retaining the last table area used to eliminate a full search each time. Could any of this be done in a preceding program?

11. Moving records to different areas, changing record formats, and using multiple input/output areas may have a net effect of increasing overall job time.

12. The use of common subroutines for tape reading and writing, developing percentages, and other functions tends to save storage and cost time because of linkage, initialization and modification requirements. If storage capacity is available, separate routines may be faster, increasing throughput.

13. Review available special features and RPQs to improve program run times.

INPUT/OUTPUT-LIMITED PROGRAMS

The following alternatives can be considered for programs with input/output requirements extensive enough to delay processing. The general approaches are to include additional processing, reduce the input/output time, and redesign the files.

1. Review the timing and instructions for the appropriate I/O device. What maximum process time is available with each I/O instruction? Which tests interlock the equipment? What buffers, special features and RPQs are available?

2. Review the files used and generated. Can the records and/or blocking factor be changed to reduce time requirements? Can two or more reports or

documents be printed at the same time, side by side?

3. Chart the time relationship of input and output devices. Can input be condensed? Can input and output be alternated to provide a gain? Will multiple input and output areas help? What alternative media are available?

4. Consider including processing currently accomplished in a preceding or following program that is process-limited, thereby achieving a better balance in both programs for a net gain.

5. Evaluate modifying the run design. Will a segmented file provide a faster search? Determine the advantages of using a "change tape", "alternating file", "split file", or extracted file in maintenance runs.

6. Balance potential gains against the investment of systems and programming effort, test time, and operator training required for implementation.

7. Many code and status indicators can be represented by the presence or absence of bits within a single character location. Review magnetic tape records and calculate the possible decrease in input/output time if the existing codes and indicators were condensed. Consider the effect on the collating sequence in sorting the file, and the possibility of generating invalid characters.

FIXED AND FLOATING POINT NOTATION

Three general requirements associated with performing calculations on a data processing system are:

1. Arithmetic operations must take into account the point (decimal or binary) of the numbers. Points of results must be properly located.
2. Numbers that vary greatly in magnitude must sometimes be manipulated.
3. Sufficient significant digits must be maintained in the numbers to provide accurate results.

Computers usually do not recognize the point of a number. Also, it is often impractical or impossible to use a word size large enough to allow all numbers to be treated as integers. Therefore, some method of number notation is required. Two such methods are described.

FIXED POINT NOTATION - SCALE FACTORS

Fixed point notation using scale factors is a method of indicating to the programmer the positions of points. It facilitates programming computational procedures.

A fixed point number is one possible machine representation of a number. It has an understood (or assumed) point at a definite position. When scale factors are used, all fixed point numbers are understood to have points in the same position, usually at the extreme left of the word; thus all numbers are regarded as less than 1. Using this convention, a fixed point number (N) will be in the interval

$$-1 < (N) < 1$$

A scale factor indicates the relation of the actual number N to the fixed point number (N) that represents it. The scale factor value is also the number of digits that separate the positions of the actual point and the understood point. If it is positive, the actual point is to the right of the understood point. If it is negative, the actual point is to the left. In the decimal number system the scale factor d is such that

$$N = (N) \times 10^d$$

In the binary number system the scale factor b is such that

$$N = (N) \times 2^b \text{ (all binary)}$$

Scale factors are determined for the input, intermediate results and final results by the programmer. A number's scale factor will depend on the bounds of the magnitude of the number and on the significance desired in the fixed point word. When bounds and significance present no problems, scale factors are selected that will minimize program instructions.

Figure 1 is a table listing the bounds, scale factor, an actual value and the fixed point representation in a five-digit word for three variables.

Variable	Bounds	Scale Factor	N	(N)
X	.xxxx	0	.9973	Λ 99730
Y	.ooyy	-2	.0068	Λ 68000
Z	z.zzzz	1	1.0041	Λ 10041

Figure 1

Once determined, the scale factors provide a guide for optimal programming of arithmetic operations and for determining the points and scale factors of results. The following are general rules for their use:

1. Shifting. -- An n-position right shift adds n to the scale factor. n is subtracted for left shifts.
2. Addition/subtraction. -- The scale factors of numbers added or subtracted must be equal.
3. Multiplication. -- The resulting scale factor is obtained from the following:

$$N_3 = N_1 \times N_2$$

$$= \left[(N_1) \times (N_2) \right] \times 10^{(\alpha_1 + \alpha_2)}$$

4. Division

$$N_3 = \frac{N_1}{N_2}$$

$$= \left[\frac{(N_1)}{(N_2)} \right] \times 10^{(d_1 - d_2)}$$

Both $[(N_1) \times (N_2)]$ and $[(N_1)/(N_2)]$ may require adjustment to make the magnitude of (N_3) less than 1 and also to retain desired significant digits. Any such adjustment will be reflected in the scale factor of (N_3) in using the first rule.

An example of the procedure is to compute $X + Y = Z$ using the scale factors in Figure 1. The scale factor must be adjusted so that X, Y and Z are the same. This can be done by shifting X one position to the right, adding 1 to its scale value and shifting Y three places to the right -- making its scale factor also equal 1, as shown below:

<u>Variable</u>	<u>Adjusted Scale Factor</u>	<u>N</u>	<u>(N)</u>
X	1	.99730	\wedge_{099730}
Y	1	.00680	\wedge_{000680}

The addition can now be performed and the Z sum will have a scale factor of 1.

It should be noted that an initial scale value of 1 assigned to X and Y would have avoided the shifts in programming.

Determining bounds of numbers and manipulating points can be a complicated procedure where the bounds of number magnitudes vary greatly; otherwise fixed point processing is much faster than floating point processing.

Another method of fixed point notation allows all fixed point numbers to be treated as integers. The least significant digit position that will occur in any number used is assigned the rightmost position in the machine word. Other numbers are located in words so that all digits are properly related to the least significant digit position already established. This technique is widely used in simple calculations but does not provide for numbers with magnitudes exceeding the word length.

FLOATING POINT NOTATION

Floating point notation is a method whereby an extra digit or digits are appended to a fixed point number to indicate where the point should be. This allows the actual point of a number to be determined during processing.

A floating point number consists of two parts:

The mantissa is the machine representation of a number. It is always understood to have a point in a particular position, usually at the extreme left. This convention means that the magnitude of a mantissa is always less than 1.

The characteristic is a number associated with the mantissa to indicate its relation to the actual number. The characteristic will be used during processing to determine the position of the actual point.

If (N) is the mantissa (fixed point number) that represents the actual number N, the characteristic is D in the decimal number system,

where $D = d +$ a decimal constant

such that $N = (N) \times 10^d$

In the binary system B is the characteristic,

where $B = b +$ a binary constant

such that $N = (N) \times 2^b$ (all in binary)

d and b have precisely the same mathematical meaning and function here as they did in the discussion of scale factors. They therefore indicate the number of digits separating the positions of the actual point and the understood point.

The constants are used with fixed word processors so that the characteristics are always unsigned positive numbers. This leaves the sign position of the word available to sign the mantissa.

For example, to represent a number N in the IBM 7070 where the constant is 50:

If $N = -.0005862$
 $d = -3$
 $(N) = -.5862000$
 $D = 47$

the floating point number is -4758620000.

The constant can be zero in a variable word length processor where an adjacent field can be used to store and sign the characteristic.

The number -.0005862 appears in the IBM 1620 as

58620-----003

The mantissa length may be variable in a variable word length processor.

Floating point subroutines or machine instructions perform arithmetic operations on floating point numbers after analyzing the characteristics. Processing is similar to the rules mentioned under scale factors and produces results in floating point numbers.

A floating point number is said to be normalized when the most significant digit of the mantissa is nonzero. Most floating point operations use normalized numbers.

The primary advantages of floating point notation are:

1. It minimizes the scaling and programming required to manipulate numbers with points. Standard floating point machine operations or sub-routines can generally be used for any number that is expressed in floating point form.
2. Even if floating point numbers vary in magnitude, significance will still be maintained.

The disadvantages of floating point processing are:

1. Effective word size is reduced because of the characteristic. This can be overcome by using multiple-word or larger variable field mantissas.
2. Processing time may be considerably more than that required by fixed point processing.



IBM COMMUNICATIONS PRODUCTS

This paper provides a brief summary of announced IBM communications products. Further information is available in the IBM Sales Manual and in the references listed at the end.

Special communications devices (RPQ items) are not included. Requirements which cannot be met by standard products should be referred to the Special Equipment Department.

65 AND 66 DATA TRANSCEIVERS

The data transceiver, used in conjunction with a 67 Telegraph Signal Unit or 68 Telephone Signal Unit, provides low-speed, half-duplex, point-to-point transmission of punched card data at five to eleven cards per minute. Transmission may be over leased lines or, with a Digital Subset Feature, over dial-up telephone facilities. Transmission is completely checked to insure accuracy, and, with the Successive Card Checking Feature, error cards may be offset-stacked to avoid transmission interruptions. Transceiver terminals may be integrated into the 7740 Communications Control System or other centralized communications systems by provision of appropriate adapters on central equipment.

References

Reference Manual A24-0512
Installation Manual C24-1037

357 DATA COLLECTION SYSTEM

The 357 Data Collection System consists of a number of input stations connected through a control unit to an output station. The 357 Input Stations, in various models, provide for punched cards, badge and manual data inputs. Up to 20 stations may be connected to one output station, a modified card punch. Transactions may be formatted singly or in multiples into output cards. A clock is available for automatic insertion of time of day in hours or hundredths.

References

Machine Bulletin 624-0518
Planning Chart X24-6862
Physical Planning Manual A24-1032
Reference Manual A24-1027
Controlled Reset Feature 624-1070
373 Punch Switch 624-3046

1408 Transmission Unit G24-3078
Programming the Output Station J24-1076

1001 DATA TRANSMISSION SYSTEM

The 1001 provides a low-cost terminal for the centralized collection of data from low-volume locations. Dial-up connection is made to a central punching station through the public telephone network. Either numeric or alphabetic punched cards and numeric manual data can be transmitted for punching into an output card. The system is simplex, providing for transmission only, and not receiving at the remote locations.

References

Reference Manual A24-1029
OEM Information A24-1049
Alphabetic Feature 624-1084
OEMI Alpha Feature A24-1085

1009 DATA TRANSMISSION UNIT

The 1009 provides a half-duplex communications terminal for an IBM system (except 7072). Transmission is over leased or public telephone lines at 150, 250 or 300 cps. Transmission is fully checked, and errors are corrected automatically by retransmission. The 1009 uses a 4-out-of-8 transmission code, and is compatible with the IBM 1013 and 7702 terminals as well as other 1009s on remote processors, the 7750 Programmed Transmission Control and the 7740 Communications Control System.

References

GI Manual A24-1039
OEMI Manual A24-1065
1009 Data Transmission Unit for IBM 1440, A24-3027

1013 CARD TRANSMISSION TERMINAL

The 1013 transmits and receives punched card data at 150, 250 or 300 cps on leased or public telephone circuits with another 1013, or with an IBM 1009, 7702, 7750 or 7740. Operation is controlled by a stored internal program. The effective character speed depends (1) on the number of characters per card and the number of cards per record, (2) on whether the 1013 is reading or punching, and (3) on the type of terminal with which it is communicating.

Data is buffered in core storage, and errors are corrected by automatic retransmission. The terminal is half-duplex.

References

Reference Bulletin A21-1068
Installation Bulletin J24-1069

1014 REMOTE INQUIRY UNIT

The 1014 provides remote keyboard input to, and hard copy output from, a 1410 or any 7000 system except the 7072. Connection is via an appropriate 1414 Synchronizer. Transmission is at 15.5 cps, in half-duplex mode. Circuits are limited to eight wire-miles. Up to ten 1014 units may be attached to one adapter on the 1414.

References

Reference Manual G24-1444
Installation Bulletin J24-1447

1030 DATA COLLECTION SYSTEM

The 1030 system provides combined two-wire and multiconductor data collection from punched cards, badges, data cartridges and manual levers. Output at remote locations is provided by printer stations. The system operates online to a central data processor, or offline to a receiving card punch. Existing 357 Data Collection Systems can be placed online with a 1030 system by use of the 1408 Transmission Unit.

References

1408 Machine Bulletin G24-3078
1030 SRL A24-3018
1030 Installation A24-3021
1030 Configurator A24-3045

1050 DATA COMMUNICATION SYSTEM

The 1050 system provides various combinations of keyboard, card, and paper tape inputs, and printer, card, and paper tape outputs at remote terminal locations. Line operation is at 14.8 cps, in a half-duplex mode. Two models are available, one with all components online, the other component assignment to either home loop or online channels which can operate simultaneously. Operation may be point-to-point between terminals, with a central processor utilizing an IBM 1448, or 7750, or with

a 7740 Communication Control System. Transmission is checked, and automatic error correction features are available. Up to 26 units may be connected to one circuit and addressed individually.

References

GI Manual A24-3020
Installation Planning A24-3022
1050/1448/1440 with 1410/7010, A22-0537
(See also manuals listed under 1060)

1060 DATA COMMUNICATION SYSTEM

The 1060 is an online data recording and transmission system for banking applications. Transmission is half-duplex, at 14.8 cps. Job-oriented teller terminals are available in online only or online/offline models. A 1061 control unit connects either one or two terminals to an adapter on the 1448 Transmission Control Unit.

References

Reference Manuals	A21-9001 1060 Configurator A24-3034 (1060) A24-3010 (1448) A24-3019 (1440-1401) A24-3049 (1440/1448 - 1410/7010) A22-0537 (1410/7010 - 1440/1442) A22-0538
Installation	A24-3014 (1448) A24-3026 (1447-4)

1414 I/O SYNCHRONIZER

Models 4, 5 and 6 of the 1414 Synchronizer provide communications capabilities for the IBM 1410 and any 7000 series processor except the 7072. Online, buffered operation of a 1009 Data Transmission Unit, 1014 Remote Inquiry units, telegraphic channels and the 1011 Paper Tape Reader is possible, in various combinations.

References

SRL	A22-0523 (1410) A22-6720 (7010) A22-6288 (7040/7044)
Reference Manuals	G22-6653 (7070/7074) G22-6624 (7080) G22-6625 (7090)
Installation	J22-6626 (Models 5, 6)

1447 CONSOLE

Model 2 provides attachment of a local 1051 terminal unit to a 1440 or 1460 system. The local terminal allows attachment of a single communications line with up to 24 remote 1050 terminal systems attached.

Model 3 provides similar terminal attachment capabilities for a 1401 system.

Model 4 provides attachment of a 1448 Transmission Control Unit to a 1440 or 1460 system. It also provides direct attachment of a local 1051 controlling up to 24 remote 1050 systems.

By addition of Buffer Feature 1490, operation of the console and terminal system is completely buffered.

References

Reference Manual A24-3031
Installation A24-3064

1448 TRANSMISSION CONTROL UNIT

The 1448 provides for connection of up to 40 half-duplex circuits (1030 Data Collection and 1050 and 1060 Data Communication Systems) to either a 1440 or 1460 system. Line control functions and character buffering are accomplished by the 1448, under program control of 1440 or 1460.

References

Reference Manual A24-3030
A24-3010 (1440)
A24-3050 (1460)
Installation Manual A24-3014

7701-7702 MAGNETIC TAPE TRANSMISSION TERMINAL

The 7701 and 7702 provide transmission and reception of magnetic tape data over voice-grade circuits--the 7701 at 75 or 150 cps, and the 7702 at 150, 250 or 300 cps. The units are compatible with each other and with the IBM 1009, 1013, 7710, 7750 and 7740 when operated at the same line speed. Transmission accuracy is checked, and error records are automatically retransmitted.

References

Reference Manual A22-6702 (7702)
A22-6527 (7701)
G22-6596 (7702)

Installation Manual J22-6606 (7702)

7710 DATA COMMUNICATION UNIT

The 7710 is a transmission control unit operating at 150, 250, 300 or 5,100 cps (depending on the communications channel used) on an IBM 1401 system. It is compatible with the IBM 1009, 1013, 7701 and 7702, as well as other 7710s, when operated at the same data speed. Error detection is automatic, and retransmission is accomplished under program control.

References

Machine Bulletin A22-6707
Installation Bulletin A22-6709

7740 COMMUNICATION CONTROL SYSTEM

The 7741, with a 1050 terminal system console and a 1311 disk drive channel, controls a communication network of up to 84 half-duplex, low-speed lines. Up to four half-duplex, high-speed lines may also be attached, depending on system configuration. The 7741 is a binary stored-program processor, with 4, 8, or 16K 32-bit words. It is capable of performing complete message routing, logging, error control and traffic accounting functions for store-forward communications systems. The system may also be used as a communications pre-processor, through attachment to a 1410 or 7000 series system. Various equipments, including telegraphic terminals, data transceivers, the IBM 1009, 1013, 7701, 7702 and 1050, can be accommodated by the 7740.

References

Principles of Operation A22-6753
Physical Planning C22-6754

7750 PROGRAMMED TRANSMISSION CONTROL

The 7750 controls data transfers between a communication network and a 1410 or any 7000 system except 7072. It contains its own stored program, and can accommodate up to 112 half-duplex, low-speed lines, 16 half-duplex, high-speed lines or combinations of high- and low-speed lines. Various equipments, including telegraphic terminals, data transceivers, IBM 1009, 1013, 7701, 7702 and 1050, can be accommodated by the 7750.

References

Reference Manual	A22-6679
GI Manual	D22-6627
Installation Manual	J22-6659
	G22-6700 (1410)
	G22-6739 (1050)

7900 INQUIRY STATION

The 7900 provides remote inquiry for a 7070 or 7074 system. Up to ten stations can be attached, five on each of two Inquiry Control Synchronizers.

References

Reference Manual	A22-7003
General Information	D22-7004
Installation	C22-6603
Bulletin	G22-6588 (7070)
Bulletin	G22-6545, G22-6583 (7074)

Reference Manuals

SRL Bibliography A24-3089
SRL Data Communications Concepts and Communications Facilities E20-8158

GLOSSARY

This glossary includes many terms which are used in other fields besides communications. Such terms are defined here as they are used in communications only.

For the sake of conciseness, some of the definitions make use of terms which are, in turn, defined elsewhere in the glossary. Consequently, several references in the glossary will sometimes have to be consulted in order to gain understanding of a term being looked up.

Address. — The coded representation of a message's destination.

Addressee. — The intended recipient of a message.

ACU (Automatic Calling Unit). — Unit capable of establishing a circuit connection upon input of a digital calling sequence from an attached processor.

Alternate routing. — A secondary or backup circuit path; used if the normal routing is not possible.

Amplitude modulation. — Variation of a carrier signal's strength (voltage and/or current), as a function of an information signal.

Analog signal. — A nondiscrete or continuous signal, the variations of which represent meaningful information.

Answerback. — The response of a data set or terminal to remote control signals.

ASR (Automatic Send-Receive). — A telegraphic terminal incorporating keyboard, page printer, paper tape reader and punch, and a stunt box capable of automatic control functions.

Asynchronous transmission (also called start/stop). — Transmission in which each intelligent character or byte is individually synchronized with a start signal.

Attenuation. — Reduction in the strength, or power, of a transmitted signal, through circuit losses.

Audio frequency. — A frequency within the range of human hearing (approximately 15 to 20,000 cps).

Automatic switching. — Switching which directs the transfer of traffic from incoming to outgoing circuits without manual intervention.

Bandpass. — The range of usable frequencies between a high- and low-frequency cutoff point; it determines the bandwidth of a channel.

Bandwidth. — The difference, in cycles per second, between high- and low-frequency cutoff points (see "Bandpass").

Baud. — Unit of transmission signaling speed, equivalent to "bits per second" (bps).

Baudot code. — Five-bit telegraphic signaling code, incorporating two shifts (figures/letters) which provide 64 combinations: 52 data characters and 6 control characters, which are identical in each shift. When transmitted, 1-bit start and 1.42-bit stop intervals are added, giving 7.42-bit effective character length.

Bid. — An attempt to occupy a line in order to pass traffic.

Bit. — Contraction of "binary digit", the smallest unit of information, which is dual-state (one or zero, on or off, mark or space).

Bit rate. — Transmission speed, usually synonymous with "bits per second" and "baud".

Broadband circuit or channel. — High-bandwidth channel, using microwave or coaxial transmission facilities.

Broadcast. — Transmission to a number of receiving locations simultaneously.

Buffer. — Temporary storage; usual applications are for speed conversion or parallel/serial conversion between asynchronous units.

Byte. — A group of bits handled as a unit; may be a character, a part of a word, or a word.

Cable. — One or more conductors, insulated within a protective covering.

Calling sequence. — A unique series of digits which establishes a circuit connection to one station.

Camp-on. — Continuous bid for a line that is busy, in order to seize it when it becomes idle.

Carrier. — A continuous frequency, capable of being modulated, or impressed with a second (information-carrying) signal.

Carrier System. — A system of combining more than one carrier frequency for simultaneous occupancy of a channel, and subsequently separating the component signals.

Central (exchange). — Switching facilities to allow interconnection of circuits from various routes.

Chad tape. — Paper tape in which the holes are completely punched out.

Chadless tape. — Paper tape in which the holes are partially punched, leaving a flap or "chad" of paper attached.

Channel. — An electrical circuit.

Character. — A digit, letter or special symbol (data); or unique function (control).

Circuit (or channel). — A continuous path for electrical transmission between two or more terminal points; may be a physical (wire) link, a radio link, or a combination.

Circuit switching. — The process of establishing an electrical connection between two channels to allow transmission.

Coaxial cable. — Two-conductor set, in which one conductor encircles and shields the other; provides better signal-to-noise and bandwidth characteristics than parallel-wire or twisted pairs.

Code. — A set of unique bit combinations assigned to represent characters.

Common carrier. — A company with a vested interest in providing communications services to the public.

Communication. — The transfer of information between separate points.

Conductor. — A metallic signal path (usually wire).

Contention. — Unregulated bidding for a line by multiple users.

Cross-office. — Telegraphic terminals for a central point at which traffic may be interchanged between separate circuits.

Cross-talk. — Interference between adjacent communications channels.

Cutoff. — The point of degradation, due to attenuation and/or distortion, at which a signal becomes usable.

Cycle. — One complete oscillation (360 phase rotation) of a signal.

Data. — Information in the form of a code.

Data-Phone. — A registered trademark of the Bell System applicable to data sets suitable for use on voice-grade data signals.

Data set (subset, modem). — A device used to convert data signals to and from other signals which can be transmitted.

DDD (Direct Distance Dialing). — Dial-up connection, without operator assistance, to a point outside the user's local service area.

Demodulation. — The process of recovering intelligence (data) from a carrier signal; the reverse of modulation.

Dial-up. — Establishment of a line connection by dialing a sequence of digits.

Dibit. — A group of two binary bits; in four-phase modulation, each possible dibit is encoded as one of four unique carrier phase shifts.

Digital signal. — A discrete, or discontinuous signal; one whose various states are discrete intervals apart.

Distortion. — Nonlinear variation of the different components of a signal.

Duplex. — Capable of simultaneous transmission in both directions.

EOM (End of Message) — A unique character, or sequence of characters, always used to terminate a message.

Exchange. — Centralized circuit-switching equipment.

Exchange service. — Interconnection, through switching, of two or more customers' communications equipment.

Extended area service. — An exchange service which covers more than the local service area.

FAX. — Facsimile; transmission of images by a process of scanning and reconstruction.

FCC (Federal Communications Commission). — U.S. agency responsible for regulation of interstate communications services.

FIGS. — "Figures" shift, in a dual-case code.

Four-wire circuit or channel. — Two-way circuit, with separate pairs for each direction of transmission.

Foreign exchange service. — Connection of a telephone to an exchange not normally serving the customer's location.

Frequency. — Rate of signal oscillation, in cycles per second.

Frequency modulation. — Variation of a carrier signal's frequency, as a function of an information signal.

Group. — A word; in telegraphic usage, arbitrarily six characters in length.

Half-duplex. — Capable of transmission in two directions, but not simultaneously.

Header. — The first part of a message, containing all necessary routing information.

Home loop. — Data path, allowing offline use of terminal components.

In-line. — Processing in order of receipt, without prior grouping or sorting of transactions.

In-plant system. — A system whose parts, including remote terminals, are all situated in one building or localized area.

Interface. — The common boundary, or physical connection, between two devices or systems.

Kilocycle. — One thousand cycles (per second).

KSR (Keyboard Send-Receive). — Telegraphic terminal with keyboard and page printer.

Leased line. — A circuit leased for sole usage, full time or during specific periods, by one customer.

Line. — See Circuit.

Line loop. — Data path, in a terminal, connected to transmission facilities.

Line switching. — See "Circuit switching".

Local loop. — The circuit connecting a subscriber to a central exchange.

Long line. — AT&T term for circuits between exchanges.

Loop. — A physical circuit.

LTRS. — "Letters" shift, in a dual-case code.

Mark (as opposed to Space). — Telegraphic term for one of two possible signals on a circuit; normal, or closed-circuit, condition is "marking".

Master station. — A terminal having selection control of all other terminals on a multipoint circuit.

Megacycle. — One million cycles (per second).

Message. — A sequence of words or symbols which is complete in itself; typically consists of header, text and EOM.

Message routing. — The process of selecting the correct circuit path for a message.

Message switching. — The process of receiving and forwarding a message, by a central location.

Microwave. — Superhigh radio frequency; nominally, 1,000 to 300,000 megacycles per second.

Mod/demod, modem. — Abbreviations for modulator/demodulator; see "Data set".

Modulation. — A process for varying one or more characteristics (frequency, amplitude, phase) of a carrier signal with an information signal.

Multidrop. — See "Multipoint".

Multiple-address message. — A message to be delivered to more than one location.

Multiplexing. — The process of sharing one facility among several users by time and/or space division.

Multipoint, multistation. — A circuit having multiple stations connected.

Network. — A system of connected points -- for example, terminals connected by communications channels and some form of exchange.

Offline. — Not connected to the transmission facility (line).

Online. — Directly connected to a central processor.

Pair. — Two wires, forming a communications circuit.

Parallel transmission. — Simultaneous transmission of the bits making up a character or byte, either over separate channels or on different carrier frequencies on one channel.

PABX (Private Automatic Branch Exchange). — An exchange which allows calls to and from the public telephone system.

PAX (Private Automatic Exchange). — An exchange not connected to the public telephone system.

PBX (Private Branch Exchange). — Manual or dial exchange connected to public telephone system.

Perforator. — A manually operated or directed paper tape punch.

Phase modulation. — Control of the phase, or timing, of a carrier signal by an information signal.

Poll, polling. — Orderly selection, one at a time, of multiple terminals to allow transmission of traffic.

Priority indicators. — Groups of characters which indicate the relative urgency of a message.

Private wire. — A circuit whose sole usage is leased or owned by one customer.

Private wire network. — A series of points connected by private wire circuit(s).

Propagation delay. — The time necessary for a signal to travel from one point on a circuit to another.

Pulse modulation. — Transmission of information by modulation of a pulsed, or intermittent, carrier; pulse width, count, phase and/or amplitude may be the varied characteristic.

Pushbutton switching. — See "Semiautomatic switching".

Rate. — The charge for a particular service, based on character or message count, number of connections, or time the service is used or available.

Record. — A group of characters treated as a unit; often, in communications, a message.

Relay center. — A message exchange point between circuits.

Repeater. — A device used to regenerate signals on a communications circuit.

Reperforator. — A line-operated paper tape punch.

RO (Receive Only). — A telegraphic terminal with receiving circuits and a page printer.

ROTR (Receive Only Typing Reperforator). — A telegraph terminal with a receiving tape punch which also prints characters directly on the tape.

Routing. — Assignment of a message to the proper circuit to reach an addressee.

Routing indicator. — An address in a message header, showing the destination circuit or terminal.

RT (Reperforating Transmitter). — Telegraphic terminal incorporating a paper tape punch (receive) and reader (transmit).

Selection. — Addressing terminal and/or components on a selective calling circuit.

Selective calling. — The ability to address one or a selected group of multiple terminals on a circuit.

Semiautomatic switching. — Transfer of traffic from incoming to outgoing circuits under operator control (usually pushbutton).

Serial transmission. — Transmission of the bits making up a character or byte, one behind the other, over a single channel.

Section	Area	Page
07	02	05

Side band. — In a modulated carrier, those frequency components above and below the carrier frequency, which contain the information being transmitted.

Signal. — Information in the form of electrical power.

Signal-to-noise ratio. — Relative power of signal to the noise in a channel.

Simplex circuit or channel. — One-way transmission, not capable of being reversed because of the design of the terminals.

Single-address message. — A message to be delivered to only one location.

Single sideband. — Transmission technique in which the carrier and one sideband are suppressed; used to conserve power, since one sideband contains all the information which was impressed on the carrier originally.

Space (as opposed to Mark). — Telegraphic term for one of two possible signals on a circuit; transferred, or open-circuit, condition is "spacing".

Speed. — Rate of transmission: words per minute; characters, or bits, per second.

Start-stop. — Asynchronous transmission.

Store and forward. — See "Message switching".

STR (Synchronous Transmitter-Receiver). — Transmission control unit for IBM's medium-speed synchronous terminal family (1009, 1013, 7702).

Stunt Box. — Telegraphic term for a device capable of performing certain functions automatically upon recognition of special control characters or sequences.

Subset. — See "Data set".

Sub-voice circuit or channel. — A circuit with narrow bandwidth, usually 200 baud or less; may be a subdivision of a voice circuit.

Switching center. — A location which terminates multiple circuits and is capable of transferring traffic between them; may be automatic, semi-automatic, or torn-tape.

Synchronous transmission. — Continuous bit-stream transmission, with no start-of-character identification.

Tariff. — A published rate for an approved commercial service offering of a common carrier.

Teletypewriter. — Western Union trade name for telegraphic terminal equipment.

Teletype. — Teletype Corporation trademark for its system of telegraphic message transmission (keyboard and paper tape input, printed and paper tape output).

Teletypewriter. — AT&T trade name for telegraphic terminal equipment.

Telex (TEX). — Western Union's domestic Teletype Exchange service.

TELPAC. — An AT&T tariff offering four grades of broadband channels.

Terminal. — Any device capable of sending and/or receiving information over a communication channel.

TEX. — See "Telex".

Text. — The information portion of a message.

Tie-line. — A leased channel, usually voice-grade.

Toll. — A charge levied for communication connection beyond a local service area.

Torn-tape switching. — Manual transfer of punched paper tape from incoming circuits to outgoing circuits.

Traffic. — Data or messages in a communications system.

Transmission. — The electrical transfer of information between two points.

Trunk. — A channel between switching centers or exchanges.

TTY. — Abbreviation for Teletype, often used to mean telegraphic.

Turnaround time. — The time necessary to reverse the direction of transmission in a half-duplex circuit.

Two-wire circuit or channel. — Single pair transmission path, capable of operation in one direction at a time; may be simplex or half-duplex.

TWX (Teletypewriter Exchange Service). — AT&T's switched network for interconnecting teletypewriter subscribers, in U. S. and Canada.

UHF (Ultra-High Frequency). — A radio signal in the range of 300 to 3,000 megacycles per second.

Voice circuit or channel. — A circuit with sufficient bandwidth to permit transmission of intelligible speech as an analog signal.

WATS (Wide Area Telephone Service). — A special telephone circuit tariff, allowing calls to any point within specified zone(s) without extra charge.

Way station. — One of the stations on a multipoint circuit.

Word. — In telegraphic usage, a group of six characters (from five characters plus one space, the average word length).

REAL-TIME SYSTEMS DESIGN

In recent years there has been an increasing and substantial demand for information systems which respond to and service customer requirements within extremely short time periods — from minutes and seconds to milliseconds. These real-time systems are differentiated from traditional batch-type systems principally by their ability to respond rapidly and process transactions with little or no delay, despite variations in input volume and timing.

Airlines reservation and stock brokerage inquiry and order-processing systems are typical, well-publicized examples of real-time operations, but an even broader market potential exists in other industries: manufacturing, retailing, transportation, etc. A large, decentralized company with multiple-point product distribution centers and warehouses, for example, can realize significant reductions in finished stock inventory while simultaneously improving customer service and cash flow by consolidating inventory control under a real-time system. With this approach, shipments can be made from dispersed warehouses the same day an order is received, and the invoice can accompany the shipment.

REAL-TIME VS. BATCH SYSTEMS

Apart from time dependency, there are other characteristics in which real-time are distinguished from batch systems: acceptance of inputs, processing of transactions, emphasis on study and evaluation techniques, etc. The concept of real-time system design is similar in many respects to the approach followed in batch applications, but with marked differences in emphasis. Simulation and queuing theory would be useful in both, for example, but these time-oriented techniques are needed and stressed much more heavily in real-time system design. Before turning to the design process, general differences and similarities between these two approaches will be examined further.

Timing and Sequence of Input

Although response speed is a principal characteristic of real-time systems, it does not always represent a clear demarcation between real-time and batch systems. Either type of system may be operated online and fully integrated with the mainstream of business. The essential distinction lies in how well inputs can be buffered. If inputs can be accumulated and sequenced in some designated order within a period of elapsed time (say, several

hours to a day), the system is fundamentally batch in nature; if inputs must be accepted and processed as they are received with no time delay, the system is operating in real time. In addition to responsiveness, then, another feature of a real-time system is the unregulated order of the input.

When the time and sequence of inputs cannot be scheduled in a system, and must be accepted randomly, their occurrence is described in terms of probabilities. In an economically feasible banking system, for example, where tellers must share a single, central file of all customer accounts, whenever several customers are demanding simultaneous service from the common facility, queues will probably form, and some individuals will have to wait before receiving service. It can be demonstrated mathematically that the amount of waiting time for a facility depends on several factors: arrival time and service rate, and the distribution of arrival and service times.

Process Overlap

Since messages (inputs) in real-time processing arrive at random times and vary in length and content, prediction of the sequence of operations to be performed is also a probability function. Generally, a new message arrives for processing before its predecessor has been completely serviced, and the system must be able to handle several inputs simultaneously. To promote optimum equipment utilization, requests are overlapped to other devices (such as disk files), which are slower than the central processor. During a disk file delay, then, another message may be initiated for processing, or a partially processed message may be continued rather than allow some equipment unit to be idle. Further, each demand on the system may require different programs within the central processing unit, a situation the system must be readily able to handle.

For these reasons, different programming methods are required in a real-time application. The ability to handle several messages simultaneously, and to efficiently allocate equipment time, core storage, etc., must be built into the programming. This clearly is an additional burden not imposed on batch-type systems.

Techniques of Design and Evaluation

A broad spectrum of tools and techniques is used in the analysis and design of information systems,

some simple, some complex. While complex techniques (e. g. , probability theory, queuing theory, matrix algebra, linear programming, etc.) can be applied in batch system design for more intricate problems, they are a decided necessity in real-time systems.

The application of queuing theory illustrates this point. Queuing delays experienced in real-time systems are not usually negligible, and to ensure that the system will meet established response time and throughput criteria, a designer must determine how long transactions wait in queue as well as queue length at any given time. Such analytic techniques are more difficult to develop and require a higher level of skill to apply than in batch systems.

THE DESIGN PROCESS

Differences in the concept and methodology of design between real-time and batch systems can be recognized mainly in the magnitude of the tasks, rather than in the type of work to be performed (Figure 1). Data collection and reduction, for example, tend to penetrate deeper at some points and continue longer in real-time systems than they do in batch systems; detailed system design frequently requires several more iterations in real-time to arrive at a satisfactory solution, and nearly always involves the application of simulation. Likenesses and dissimilarities will become more apparent in the ensuing description of the real-time design process.

Problem Definition

Initially, the problem a system is to solve is defined through an examination of present operating methods and practices. Out of this review, present and future functional performance requirements are determined, along with pertinent constraints (such as cost), and are stated as system objectives. This typifies the work performed in the first two phases when the Study Organization Plan is being followed.

General Systems Solution

System objectives and required output are then used to formulate a generalized solution to the problem in terms of a class of systems. Actual equipment is not specified at this point, but a preliminary estimate is prepared on the type of system that will best fulfill defined objectives. This might range from unit record equipment to a duplexed IBM 7094 configuration.

Once a preliminary solution has been developed,

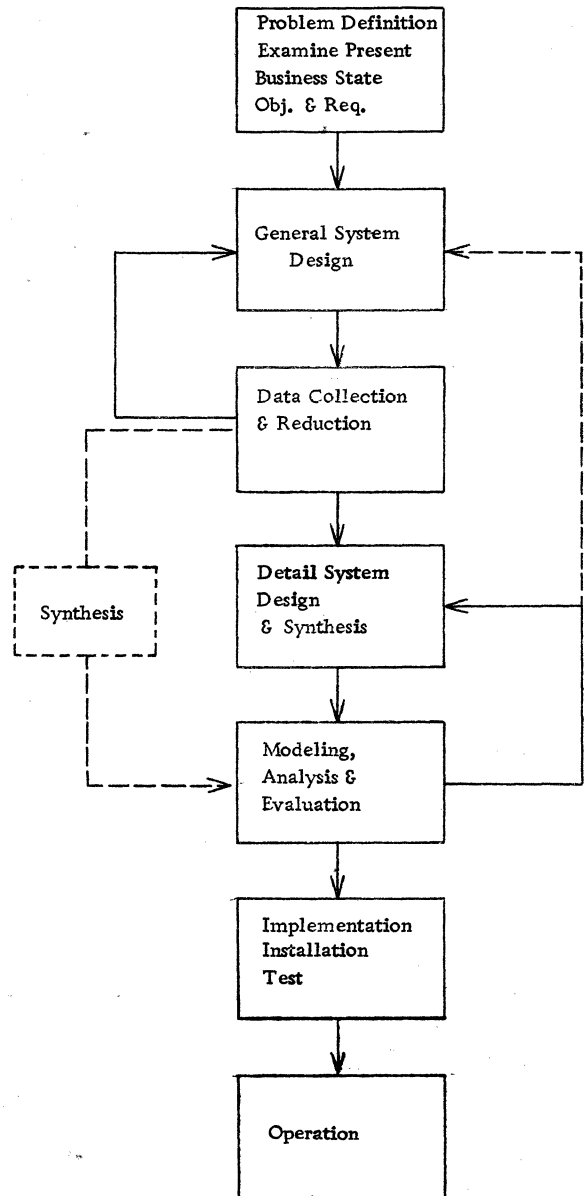


Figure 1 Real-time design process schematic

it is necessary to specify the magnitude of the job the system is to perform within an exact operating environment by:

1. Quantifying the items of work associated with proposed system functions.
 2. Defining methods and procedures to be employed in handling incoming customer work units (transactions). A real-time transaction may be a telephone call, inquiry, or any other unit of demand requiring service by the system.
- Together, these two factors are referred to as the "functional environment".

A systems engineer determines the average number of arrivals for each type of transaction over a given time period, and also the distribution of arrival times (i. e. , do transactions arrive every 5 seconds, or in groups of ten followed by time intervals when nothing transpires?). At terminal locations, transactions are converted to one or more demands on the system through defined operational procedures. These demands are referred to as "actions" (e. g. , a "sell" or "display record" operation), and are entered into the system on a specified format comprising one or more messages (a group of characters headed by an address or terminal, and confined to a stated length). Consequently, if a terminal is buffered, and a single action generates two bufferloads of input, two input messages are produced.

Data Collection and Reduction

To properly evaluate the functional environment of a business and to confirm the justification of the preliminary design decision, a considerable amount of data must be collected, reduced, and projected to reflect future conditions. Data collection has to be objective in outlook and systematically conducted to ensure that the information is a valid representation of the environment in which a system operates. Normally, a data format is set up in advance, and statistics are gathered from a field experiment in accordance with this specification. Collected data is then reduced to statistics that are meaningful in terms of operational procedures. Experimental design techniques and sampling theory are often employed in dealing with various problems of data collection and statistical inference.

Data gathered on current operations, however, provides little insight into the future, and must be projected for advance periods to anticipate conditions the system will encounter during its expected life cycle. In compiling these forecasts, care must be taken to account for variations in the traffic pattern induced by the presence of the new system; for example, a highly effective system might attract traffic not anticipated from growth patterns based on present operating practices and systems. Also, proposed operating procedures at this point in the design process should provide for unusual as well as normal conditions. Switchover might be a case of normal operation, whereas equipment failure would be an unusual condition.

When evaluation of the functional environment has been completed, a specific equipment configuration is selected within the general class of systems previously designated to best satisfy the

general system solution. The tasks involved in these two steps (i. e. , data collection and general systems solution) are closely related to the work carried out in Phase III of Study Organization Plan for batch systems.

System Design and Synthesis

Since no precise mathematical formula or analytical technique exists for synthesizing a system design and arriving at a total design solution which satisfies stated requirements and is capable of operating efficiently in a predicted environment, the process of detailed system design is essentially one of iteration.

During this iteration or "cut-and-try" procedure, the system is designed in detail. Performance is then predicted and evaluated against established design criteria, using selected analytic techniques. If the system meets all constraints imposed on it (i. e. , cost and projected performance satisfy system objectives), the design process is completed, and implementation may begin. However, if the design is considered inadequate in any respect, the system must be modified or redesigned. After another design is produced from subsequent iterations, the evaluation is repeated once again to ascertain whether it fulfills all cost and performance requirements. Sometimes a new definition of system objectives is needed as a result of redesign, and this, in turn, may necessitate a re-evaluation of the general solution.

Modeling and Analysis

Two general categories of techniques are employed in systems design and evaluation: analytic techniques (e. g. , linear programming) and simulation (e. g. , Monte Carlo techniques). The former are generally used to describe a system's behavior under expected conditions through mathematical equations. Three factors influence the applicability of the particular technique or discipline:

1. Nature of service demands is random probabilistic.
2. Sharing of equipment causes queues which have to be measured.
3. Time required to service an action is variable, depending on the state of the system — e. g. , seek time for a disk file is related to the position of the arm.

When application of mathematical techniques becomes impractical due to system complexity, Monte Carlo simulation is frequently used. Under this approach, a simplified model of the system

(simplified in the sense that nonpertinent aspects of the problem are excluded) is constructed and operated under various conditions. On the basis of the model's observed performance, predictions are made on the parameters of the system. Simulation does not obviate the need for analytic techniques, but does tend to lessen the depth of knowledge required for their application.

For convenience in design, equipment is subdivided into five general categories:

1. Terminals — input/output devices may be remotely located.
2. Communication facilities — include transmission lines, microwave links, etc.
3. Exchanges — may be programmed to control terminals and communication facilities.
4. Processors — programmed to perform major logic and control functions within the system.
5. Auxiliary storage — contains information not stored in core memory.

Design Criteria for System Evaluation

Once an acceptable design has been developed, it is evaluated to see how well it meets two criteria: capability and reliability. Capability includes an appraisal of performance in regard to response time, peak traffic conditions, etc. Reliability is concerned with the continuity of system operation and availability, especially when there is no alternate or backup equipment. If a customer faces the possibility that a major part of his operating facilities might be unavailable for any reason, he will usually impose stringent requirements on system reliability.

Mathematically, the problems of capability and reliability are quite similar. The same techniques are applied, although the parameters of the equations will be significantly different. These sets of parameters are discussed in detail under "Real-Time Systems Design Criteria", Section 07, Area 04.

Implementation

When a detailed system design has been formulated to meet all established criteria, results are documented in the form of specifications (system, environment, control and operation programs, etc.), which provide the necessary information for system implementation. Implementation tasks are much the same as in batch systems, including coding and debugging of programs, selection and training of personnel, program and system testing, physical installation of equipment, and conversion to full operation. Planning and execution of the several implementation tasks are substantial and involved due to the complex nature of real-time systems.

Operation

After the system is converted to online operation from a parallel or pilot basis, it is monitored for a time and modified as necessary to improve overall operating efficiency.

SUMMARY

There are numerous parallels in the concept and methodology of real-time and batch system design, while at the same time there are marked differences. Dependency on time and the ability to respond to demands with relatively little delay are the significant features which distinguish real-time systems from others. By their nature, real-time systems are more complex than batch systems, and present many difficult problems for the designer. They are also uniquely characterized by the way inputs are accepted and transactions processed, and they increase the need for applying sophisticated scientific techniques during system design. Iteration and synthesis are featured in the design process to a greater extent than in other types of systems. Although initial applications have been concentrated in the airline, stock brokerage, and banking fields, real-time systems have a potentially broad utility in many areas and industries.

REAL-TIME SYSTEMS DESIGN CRITERIA

DESIGN CRITERIA

Throughout the design process of a real-time system, a systems engineer must continuously bear in mind, two distinct, but related facets of performance: (1) the ability of the system to carry out its intended function (capability), and (2) the percentage of time the system will be operable (reliability). To appraise the design in terms of capability and reliability, the designer evaluates performance of the actual system or some representative model against specified design criteria. First, system parameters have to be defined qualitatively; then they are translated into numeric values that can be measured quantitatively. From these parameters the performance of two or more systems can be described in technically accurate language, and compared with each other. The following discussion of capability and reliability parameters is reasonably complete and indicative of the requirements of real-time systems, and provides a minimum base for establishing design criteria.

CAPABILITY PARAMETERS

An appraisal of capability includes three elements: transit time, stability, and expandability.

Transit time spans the interval between the instant an action enters the system, and the instant processing of the action is completed or the required output response is received at the proper location. This definition is refined further in terms of individual situations and the location at which transit time is applicable — i. e., at a remote terminal or at the data processing center. Two classes of systems have to be considered:

1. Inquiry — each input is acknowledged at the point of input.
 2. Store and forward — each input will cause an output at some other terminal.
- Transit time in an inquiry-type system is referred to as response time.

Stability is a measure of the peak load safety margin built into a real-time system. In order to provide a useful numeric value to stability, it is necessary to define the response and traffic level at which a system's stability is to be measured or predicted. Stability (70%, 100%) is a measure of the system's sensitivity to increasing traffic loads when operating at 100% of the peak expected traffic

load with the time to 70% response being the critical time interval. To attain the desired measure, the following definition is made:

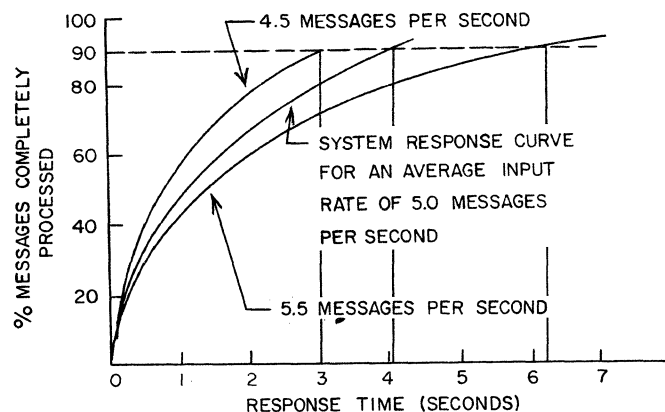
$$\text{Stability (Y\%, W\%)} = \frac{\text{Time to Y\% Response at W\% Expected Input Rate}}{\text{Time to Y\% Response at (W + 10)\% Expected Input Rate}}$$

For example, if the 90% response point is critical to a system, stability at the expected peak input rate might be defined as:

$$\frac{\text{Time to 90\% Response at 100\% Expected Input Rate}}{\text{Time to 90\% Response at 110\% Expected Input Rate}}$$

A low stability ratio is an indication that the system is sensitive to changes in its input parameters (e. g., traffic rate) and consequently unstable.

Suppose, in the above example, the system had a predicted input rate of five messages per second with the following response time characteristics:



Then, two stability factors are of interest:

$$(90\%, 100\%) = \frac{90\% \text{ Response at } 100\% \text{ Predicted Input}}{90\% \text{ Response at } 110\% \text{ Predicted Input}}$$

$$(90\%, 90\%) = \frac{90\% \text{ Response at } 90\% \text{ Predicted Input}}{90\% \text{ Response at } 100\% \text{ Predicted Input}}$$

From the graph, the first factor is equal to $\frac{4}{6}$ or .667 and the second is $\frac{3}{4}$ or .75.

Generally, if a stability factor is lower than .667, this is a warning that further and more rapid response time deterioration will occur as the input rate is incremented. When this happens, accurate knowledge of system input rate and distribution will determine the extent of exposure.

For a given system, it can be expected that stability (X%, 100%) will be slightly greater than stability (Y%, 100%), if X is less than Y. Also, stability (90%, R%) will be greater than stability (90%, S%), if R is less than S. The validity of the first comparison, of course, depends on the shape of the transit time curves for various input loads. The second comparison should hold for almost all cases; i. e., as traffic level rises, the system's sensitivity to a further increase in traffic should also rise.

Expandability is the potential of a system to be augmented or modified to increase its performance capacity, while still maintaining a basic equipment configuration and programming philosophy over a reasonable time period, with minimal cost and inconvenience to the customer. For example: "This system will handle X times the projected traffic while still meeting stated transit time and stability criteria with the following modifications, etc."

RELIABILITY PARAMETERS

Evaluation of when and how long a system will be operable is included under the category of reliability. An important factor here is switchover delay, or the total length of time it takes to effect a switchover. Where certain operations have to be performed during switchover (e. g., diagnostics or file updating) before the system again becomes

operational, this time is included under switchover delay. Other criteria are also measured under reliability, but discussion of these matters is beyond the scope of this paper.

ADDITIONAL PARAMETERS

The foregoing criteria are used where they are appropriate, ignored where they are not applicable. While capability and reliability parameters are fundamental in major or minor degree to every real-time system evaluation, others may also be employed to augment the appraisal when they are necessary and relevant.

Each new parameter developed and introduced in real-time system evaluation must, however, meet the test of being meaningful, measurable and predictable. System requirements are determined early in the design process and are specified in terms of the job to be performed; system parameters, too, should be job-oriented and interpret system operation meaningfully in accordance with the tasks to be performed. In like manner, there can be no assurance a system meets criteria established in the design process if the performance parameter is not measurable. Every parameter purporting to describe system performance should be measurable in the field and predictable in the laboratory, using mathematical and simulation techniques.

REAL-TIME SYSTEMS REQUIREMENTS CHECKLIST

The following checklist is designed to assist the reviewer in his evaluation of capability models. The points outlined here represent the characteristics and significant factors which have appeared in several systems studied to date. These systems were of maximum size and complexity and the points reflect this. For systems of less complexity, some of the points will not be applicable. Nonetheless, it is believed that these points will serve as a means of suggesting questions and areas of exploration in the review of such models. Examples or explanatory comments have been attached to some of the points to aid the reader. It is impossible to make any a priori statements concerning the importance of the individual points for particular systems. Each point must be evaluated in terms of the system under consideration. Finally, the points and their order of appearance are not intended to be followed as a rigid procedure.

DETAILED CHECKLIST

I. Message Handling

A. Input.

1. Distribution of message types.
2. Message types which are subdivided into programming action blocks. (These blocks may be processed independently of the total message.)
3. The sources of each message type. Type of terminal. Line type, e.g., private, dial-up, high-speed, low-speed. (The coding and decoding time of teletype messages and those on low-speed lines differ.)
4. Message types.
 - a. Size in characters.
 - b. Data (central processing unit — CPU — or exchange core) per message.
 - c. Messages requiring more than one block of (transmitted) data prior to processing. (Information items in transmitted data blocks may be dependent on each other, and thus require processing as a group.)
5. Subclasses of message types. (A message may be of a specific class, but because of the data which it contains, it does not require the same amount of processing as another message in the identical class.)
6. Priority by message type.
 - a. Real-time peak messages (messages which require immediate processing).
 - b. Real-time nonpeak messages (messages

having a low processing priority).

- c. Nonreal-time messages or internal activities. (A class of nonreal-time messages would be batch-type messages or special programming requests. Internal activities would be the completion of a series of events, time-initiated programs, etc.)
7. Method of polling.
 - a. Hub go-ahead.
 - b. Roll call.
 - c. Contention.
- B. Output.
 1. Programs which generate output messages.
 2. Number of multiple output transmissions. (Multiple transmissions may be from a common block of core or from individual blocks of core set up for each message.)
 3. Length of each message type.
 4. Priority of output messages. (See A-6.)
 5. Method of storing low-priority and nonreal-time messages.
 6. Method of handling long-length messages.
 - a. Segmented vs. continuous.
 - b. Location of segments.
 - c. Method of accessing segments.
- II. Core Allocation and Release
 - A. Generally falls into three categories.
 1. Direct request — Operational programs may call on core to perform such functions as shifting of data to output area, data storage, and format editing.
 2. Indirect request — Operational programs may call on a control program containing implied request for core (e.g., auxiliary storage read).
 3. Conditional request — Operational programs may request data or additional program segments which, if not already in core, will automatically be called from auxiliary storage and assigned core. The parameters which determine whether or not the requested data is in core must also be specified in these cases.
 - B. Permanent storage. The amount of CPU core storage permanently assigned should be specified. Although permanent storage does not directly affect the simulation, the reviewer must ensure that enough core is available to store the necessary items. This is allocated among three general categories.

1. Control programs — Those programs associated with the overall control and scheduling of the events occurring within the CPU, exchange and auxiliary storage.
 2. Stored operational programs — The programs permanently in core due to high usage or relatively small size.
 3. Permanent data and reference tables — These include such items as:
 - a. Current line and terminal configuration (for error and failure correction and detection).
 - b. Records of storage allocation (i. e., addresses of available records).
 - c. Current equipment configuration (for fallback).
 - d. Storage for intermediate results.
 - e. Conversion tables (BCD - binary, TTY - BCD).
- C. Data storage. Data storage consists of segments or blocks of core which are temporarily allocated to actions which are being processed. Such blocks are used for storing data records, as work areas, and for storing (action) messages as they arrive. Actions are generally assigned a segment of core (called an entry block) as they enter the CPU or exchange. When processing is complete and the entry block is no longer required, it is released.
1. The following items pertaining to entry blocks should be reviewed:
 - a. Method of assigning entry blocks to actions.
 - (1) Priority by which entry blocks are assigned to messages.
 - (2) Amount of processing required prior to assigning block.
 - (3) Amount of processing required to allocate block.
 - b. The size, in units of core storage, of entry blocks.
 - (1) The parameters which determine length, and their value by action type.
 - (2) Maximum number of entry blocks which may exist within CPU simultaneously.
 - c. Restrictions on multiple use of entry blocks. (May not accept only alpha or numeric data. Processing time required to handle alphameric or numeric data may vary.)
 - d. Location of entry blocks (fixed vs. relocatable).
 - e. Method and time of releasing entry blocks. (May not be permitted until message has been transmitted to terminal and response received.)
 2. In addition to the entry blocks, operational programs may obtain additional core storage (record blocks) under supervision of the control program. The following points pertaining to this must be accounted for:
 - a. The method and time of each core request by an operational program.
 - b. The amount of core requested.
 - (1) Parameters determining amount and their value by function, if necessary.
 - (2) Limitations on the amount which can be requested by individual programs or actions.
 - c. The method and time by which core is released.
- D. Program storage. Certain operational programs, or segments thereof, in addition to those permanently in core, may be required. These may be read into an area of core specifically reserved for this purpose or they may be read into data record blocks. Larger programs are generally broken up into smaller pieces called "segments" to provide more efficient utilization of core. The following points pertaining to the allocation and use of this storage should be accounted for:
1. Size of (operational) program segments.
 - a. If variable, the parameters which determine size.
 - b. Maximum number of segments which may exist within CPU at any given time.
 2. Method of transferring among segments.
 - a. Transfer of processing.
 - b. Disposition of old segment — May be destroyed or held until needed again by same message.
 3. Multiple use of segments.
 - a. Number of input messages able to use same segment.
 - b. Conditions governing multiple use.
 4. Method of requesting segments.
 - a. Conditional — Must ascertain location, e.g., core, disk, drum, tape, etc.
 - b. Indirect (performed by control program).
 5. Core location of segments.
 - a. Fixed core locations — Additional processing required to transfer segments. Execution delayed until space available.
 - b. Relocatable — Requires additional input CPU or channel processing due to necessity of partial reassembly.
 6. Method and time of release of segments.
 - a. Criteria for conditional replacement by other segments — size, frequency of use, time in core priority.
 - b. Unconditional replacement.

c. Direct or indirect may call upon control program to perform.

The amount of CPU processing time for each of the above functions should also be included in the model.

III. Overflow Control

- A. Indicators used to determine overflow conditions.
 1. "Peril points".
 2. Threshold values.
- B. Methods used to inhibit overflow.
 1. Reduce polling rate.
 2. Notify personnel (at remote locations or DPC).
 3. Defer certain action types.
 4. Inhibit auxiliary storage read commands.
- C. System operation under overflow conditions.
 1. Messages routed to auxiliary storage.
 2. Terminals disconnected or not polled.
 3. (Processing) limited or deferred.

IV. Multiprocessing

- A. Restrictions on number or type of actions concurrently processed.
 1. Availability of equipment — Actions requiring a large amount of core or long processing periods due to multiple auxiliary storage accesses may be deferred.
 2. Availability of data — Certain actions may "hold" data required by others. This will preclude processing until data is available.
 3. Priority — Actions of different priority may not be processed concurrently.
- B. Priority of processing.
 1. By action type.
 2. By time of day.
 3. By equipment status. (During periods of high equipment utilization, low-priority actions may be delayed.)
- C. Several types of queues may be maintained by the programs.
 1. Lists containing newly arrived actions ready to commence processing.
 2. Lists of partially processed actions waiting for data from auxiliary storage.
 3. Lists of deferred or nonreal-time actions like file purging, report generating, checking, etc.
- D. Method of queuing actions for processing.
 1. By time of arrival or priority.
 2. By "minimum delay" criteria.
 3. Queue reorganization, e. g., the CPU may maintain a record of the cumulative access times for each disk access mechanism. The access mechanism having the least time

(either the original or copy) will be the one used.

- E. Method of servicing queues.
 1. Priority or first-come, first-served.
 2. Conditional — Messages in queues for files are usually assigned by order of arrival. By reorganizing the requests by access arm, the total time for all requests may be reduced.
 3. A number of programs may be queued up in chronological order waiting for a common table or program to be entered into core. When this table or program arrives in core, another program, in processing, may also require the same record. This program may either continue, taking precedence over the programs in the list, or it may be placed at the end of the list.
- F. Maximum number of message segments per action in processing at one time.
 1. Restrictions — Segments may be processed independently but refer to common data. If this data is altered by a segment, its status must then be known, and will impose restrictions.
 2. Sequence of processing — May be a function of status of data or status of equipment.

V. Auxiliary Storage Allocation

- A. Method of creating new records.
 1. Obtaining location of spare record.
 2. Making record of the assignment.
 3. Number of copies created.
 4. Processing time required for the above.
- B. Method of assigning (record) addresses.
 1. By time of arrival (chronological arrival).
 2. By event.
 3. By location of customer.
 4. By distributing new entries across all files. (The demands on files are often associated. Thus, records may be distributed across multiple auxiliary storage units to distribute the peak load activity.)

VI. Auxiliary Storage Release

- A. Method and time of release of auxiliary storage records.
 1. Disposition.
 - a. Complete purge.
 - b. "Downgraded" to slower auxiliary storage unit. (After a shipment is delivered, the customer's record is removed from a disk file and is placed on a tape file. The file record is greatly abbreviated and data

which was necessary to handle the shipment is deleted altogether.)

2. Purging technique used to remove unused records.
 - a. Real time.
 - b. Deferred. (The control program may purge on an hourly or daily basis rather than on a real-time basis, thereby saving processing time during a peak period.)

B. Processing times for the above functions.

VII. Referencing Existing Auxiliary Storage Files

A. Read operations.

1. Method of obtaining auxiliary storage records.
 - a. Multiple copies — Either the original or a copy of a record might be read depending

upon which can be obtained most rapidly.

- b. Single copies.
2. Restrictions.
 - a. "Hold" conditions on equipment or records.
 - b. Unavailability of CPU core.
- B. Write operations.
 1. Method of updating auxiliary storage records.
 - a. Operations involving original and a copy.
 - b. Method of generating copy. (This can be complex if the original is in core or drum file and the copy is on a "lower level" of storage: drum file or disk file.)
 2. Restrictions.
 - a. Records "held" for future reference.
 - b. Equipment "held" by operational programs.
 - c. Core areas from which record is written held for "readback" check.

SELECTED REFERENCE BOOKS

It is not possible for a systems engineer to be an expert on all or even a major share of the subject areas covered by systems engineering. Therefore, as he encounters new and different problem situations, or becomes involved in specialized fields of work, he must know where to turn for additional knowledge and, more important, how to select from the large volume of source material available to him. Of course, a systems engineer interested in self-development does not wait until a need arises to expand his knowledge. He plans a comprehensive reading program and follows it to keep current on data processing and application technology. For these reasons, certain selected readings are included in this part of the handbook.

The following list of reference books incorporates a variety of technical and nontechnical subjects from the principal disciplines employed in systems work: computational methods and control, operations research, probability and statistics, systems methodology and procedures, application of computers, etc. Material, content and coverage of these texts amplifies the subject matter of this handbook and provides a far greater depth of detail than is possible in the individual areas.

While the list cannot be considered an authoritative and definitive bibliography (every text has its strengths and weaknesses), it does represent a diversified selection from the current literature as indicated by favorable reception in the field or by recommendations from various readers involved in systems work.

Books are listed alphabetically by author; entries include title, publisher, number of pages, price and a brief description of content.

For further reference material see Data Processing Bibliography (3rd revision), IBM Corp., 1962, 23 pp. (J20-8014-3).

Batchelor, J. H., Operations Research -- An Annotated Bibliography, St. Louis University Press, 380 pp., \$12.50. This volume provides in conveniently accessible form a much needed digest and index of the rapidly growing world literature of operations research for the year 1960 with additions for preceding years not included in the volume issued in 1962.

Brown, R. G., Statistical Forecasting for Inventory Control, McGraw-Hill, 232 pp., \$7.75. A definitive discussion on inventory control,

emphasizing the development of a practical system for routinely estimating maximum reasonable demand during a lead time. A number of decision rules are analyzed and graphically illustrated.

Brownlee, Industrial Experimentation, Chemical Publishing Co. An excellent statistical "cookbook" for quick reference, especially in experimental design.

Burington, R. S., Handbook of Mathematical Tables and Formulas, 3rd edition, Handbook Publishers, Inc., 275 pp. Contains basic formulas and theorems from elementary mathematics along with extensive tables.

Canning, R. G., Installing Electronic Data Processing Systems, John Wiley & Sons, 193 pp., \$6. A discussion of the problems encountered and planning necessary for the installation of data processing equipment. Uses a hypothetical manufacturing company as the basis for discussion, hence is limited to one set of conditions and one set of solutions.

Chorafas, D. N., Operations Research for Industrial Management, Reinhold Publishing Corp., 303 pp., \$8.75. Demonstrates the broad scope of operations research in business and industry by describing its use in a wide range of activities. Although the emphasis is on simulation studies, there is also discussion of game theory, linear programming and matrix algebra.

Churchman, C. W., et al., Introduction to Operations Research, John Wiley & Sons, 645 pp., \$12. A very complete text on various techniques of operations research. Includes information on establishing and implementing an operations research staff.

Dorfman, R., and P. Samuelson, and R. Solow, Linear Programming and Economic Analysis, McGraw-Hill, 525 pp. A general exposition of the relationship of linear programming to standard economic analysis, stressing economic and practical aspects of the problem.

Faddeeva, V. N., Computational Methods of Linear Algebra, Dover Publications, Inc., 252 pp., \$1.95. This book contains numerical methods for solving systems of linear equations and for the computation of proper numbers of a matrix.

Feller, W., An Introduction to Probability Theory and its Applications, John Wiley & Sons. An excellent mathematical treatment of probability theory with examples. Queuing theory is in the latter part of the book.

Flagle, C. D., and W. H. Huggins, and R. H. Roy, Operations Research and Systems Engineering, Johns Hopkins Press, 889 pp., \$14.50. This book is a compendium on the meaning, use and importance of these fields. The objectives and tools are defined; methodologies are explained, discussed and applied to actual case studies.

Gantmacher, F. R., Applications of the Theory of Matrices, Interscience Publishers, 317 pp., \$9. A mathematical treatment slanted toward numerical analysis and engineering uses.

Gass, S. I., Linear Programming: Methods and Applications, McGraw-Hill, 223 pp., \$6.75. A basic presentation of the theoretical, computational and applied areas for the student, research analyst and other technical personnel.

Grabbe, E. M., and S. Ramo, and D. E. Wooldridge, Handbook of Automation, Computation and Control, 3 vols., John Wiley & Sons, Approx. \$55. A reference book for technical personnel in systems engineering, covering (volume 1) control fundamentals, (volume 2) computers and data processing, and (volume 3) control system design and applications.

Hall, Arthur D., A Methodology for Systems Engineering, D. Van Nostrand Co., 478 pp., \$12. Increases awareness and understanding of systems engineering as a process and sharpens approaches to the main recurring problems of the process-problem definition, goal setting, systems synthesis, systems analysis, choosing among alternative systems, and planning for action.

Hildebrand, F. B., Introduction to Numerical Analysis, McGraw-Hill, 511 pp. A good mathematical treatment of error analysis, differential equations, and approximations.

James, G., and R. James, Mathematics Dictionary, D. Van Nostrand, 546 pp. Contains definitions of symbols, terms, and commonly used functions in mathematics.

Lazzaro, V., Systems and Procedures, Prentice-Hall, 464 pp., \$10. Covers major techniques of systems development from work simplification to operations research.

Leeds, H. D., and G. M. Weinberg, Computer Programming Fundamentals, McGraw-Hill, 368 pp., \$6. This textbook covers the machine-level fundamental principles of computer programming, using the 7090 as its model. Programming concepts are introduced gradually and clearly with examples. The more advanced subjects include trapping, indirect addressing, overlapping operations, symbolic debugging, and supervisory programs.

Ledley, R. S., Programming and Utilizing Digital Computers, McGraw-Hill, 550 pp. A complete and up-to-date text providing an introduction to the many aspects of the digital computer programming field, including automatic programming languages as well as machine languages; the programming of special-purpose computers as well as general-purpose computers; mathematically oriented as well as business-oriented applications; and the modern areas of heuristic programming, as well as mathematical decision programming.

Magee, J. F., Production Planning and Inventory Control, McGraw-Hill, 233 pp., \$7.50. Introduces the concepts and methods which have been developed, the functions which inventories serve, and the pressures and costs bearing on inventory and planning decisions. It points out the useful application of concepts and techniques in the practical design of production planning and inventory control systems.

Mayer, H. A., Symposium on Monte Carlo Methods, John Wiley & Sons, 382 pp., \$7.50. A collection of papers presented at the University of Florida, March 1954.

McCracken, D. D., and H. Weiss, and Tsai-Hwa Lee, Programming Business Computers, John Wiley & Sons, 510 pp., \$10.25. An excellent text which goes beyond the subject of coding to discuss basic concepts of file organization, flowcharting, scheduling, etc., for the business user. Includes numerous helpful examples.

Mosteller, Frederick, and Robert E. K. Rourke, and George B. Thomas, Jr., Probability and Statistics, Addison Wesley, (used with course from Education R 20.4019). Acquaints the reader with the theory of probability -- the mathematics of uncertainty -- and illustrates applications of probability to statistical theory.

Optner, S., Systems Analysis for Business Management, Prentice-Hall, 276 pp., \$10. A documentation of what is a system and what is the systems approach. A framework is provided for understanding what makes a company operate, and how its operations are or are not contributing to its objectives.

Randall, C. B., and S. W. Weimer, and M. S. Greenfield, Systems and Procedures for Automated Accounting, South-Western Publishing. This book examines and discusses data processing from very basic integrated hand systems through complex punched card systems, emphasizing the skills needed by a good systems man.

Richards, R. K., Arithmetic Operations in Digital Computers, D. Van Nostrand Co., Inc.,

Section	Area	Page
08	01	03

397 pp., \$7.50. Carefully written book for the computer designer. Concerned with the logic of digital computer circuits.

Roy, Robert H., The Administrative Process, Johns Hopkins Press, 236 pp., \$5. This book presents a business philosophy which can be considered as a problem-solving guide for modern management.

Sawyer, W. W., Mathematician's Delight, Penguin Books, 233 pp., 95¢. Describes mathematics in language easily understood by the non-mathematician.

Titchmarsh, E. C., Mathematics for the General Reader, Doubleday Anchor Book, 197 pp., 95¢. Used at IBM Systems Research Institute.

Von Neumann, John, The Computer and the Brain, Yale University Press. Notes representing Von Neumann's views on a comparison between the computer and the brain.



SELECTED PERIODICALS

The problem of choosing periodicals for current technical reading is equally as difficult as reference book selection. A large variety of periodicals is available to the systems engineer on computer subjects and related fields. Topic coverage ranges from general data processing information to specialized information, such as information retrieval, programmed learning, decision theory, computational algorithms, etc.

Most often, a systems engineer makes a choice among periodicals by reading or scanning one or two issues of a publication and progressively eliminating those that have low value. He should also check with other systems personnel to determine their reading preferences and reasons for them.

Since it is not economically practical to subscribe to a large number of periodicals, the systems engineer should search out local sources of desired periodicals. It is also desirable to read abstracts and reviews to pick out key articles.

This list of selected periodicals covers most of the important publications (domestic and foreign) concentrating on data processing and its application. The systems engineer can use this list to screen out publications with no immediate interest and find those that are related to his present work.

The list is arranged alphabetically by publication, and the details for the various entries are given in this order:

Name of publication	Content (code)
Publisher	
Address	
Frequency -- copy cost -- annual subscription cost	

The content code indicates the subject of major emphasis. Content codes and their explanation are as follows:

Content Code	Explanation
Gen Inf	<u>General Information</u> - Industry news, and broad, nontechnical surveys of topics listed in other column headings.
Bus App	<u>Business Applications</u> - Data processing applications of the "traditional" type.
Mgt Sci	<u>Management Science</u> - The application of scientific methods and techniques to the operating problems of business.

Sci App	<u>Scientific Applications</u> - Applications from science and engineering in which mathematical computing predominates.
Theo Alg	<u>Theory and Algorithms</u> - Primarily abstract analytical treatment of principles for obtaining desired results in numerical problems, logical and language-handling problems.
Prog Sys	<u>Programming Systems and Languages</u> - Self-explanatory.
Inf Ret	<u>Information Retrieval</u> - Self-explanatory.
Data Tr	<u>Data Transmission</u> - Primarily the facilities, techniques and services of the communications companies, rather than those of the computer manufacturers.
DP Desc	<u>DP Equipment; Descriptive</u> - Covering products of the DP industry in a functional manner aimed at users, rather than designers.
DP Log	<u>DP Equipment; Logical Design</u> - Delves analytically into the circuits in the black boxes. Most useful to those systems engineers with knowledge of electrical engineering principles.
Rel Eqt	<u>Related Equipment</u> - The variety of equipment usable in conjunction with data processing systems: plotters, special visual displays, analog computers, direct data input and output devices, sensing instruments, controllers. Specific coverage within this area can be judged from the periodical titles themselves.

Notes (a) to (p) referenced in the listing are explained at the end of this area.

Information on periodicals in all fields (not limited to systems engineering interests) is included in the following publications, commonly available in libraries:

Ayer, N. W. & Son's, Directory of Newspapers and Periodicals, published annually by N. W. Ayer & Son, Inc., West Washington Square, Philadelphia 6, Pennsylvania.

Ulrich's Periodical Directory, published by R. R. Bowker Company, 62 West 45th Street, New York 36, New York.

LIST OF PERIODICALS*

AID: Auto-instructional devices for education and training. (a)
 INRAD Educational and Training Methods Division
 P.O. Box 4456, Lubbock, Texas
 Monthly - \$0.50 - \$5.

American Documentation Inf Ret
 American Documentation Institute
 1728 N Street, N. W.
 Washington 6, D. C.
 Quarterly - \$3.50 - \$12.50

ACM Communications Gen Inf,
 Association for Computing Theo Alg,
 Machinery Prog Sys
 14 East 69th Street
 New York 21, New York
 Monthly - \$2. (\$1. to members) - \$10. (\$5. to members).

ACM Journal Theo Alg,
 (Same as ACM Communications, Prog Sys
 except single copies are \$4. to non-members, \$2. to members.)

ATE Journal (British) Data Tr
 Editor, A. T. E. Journal
 Strowger Works, Liverpool 7, England
 Quarterly - "Free on a controlled basis".

Automatic Control Rel Eqt
 Reinhold Publishing Corporation
 430 Park Avenue
 New York 22, New York
 Monthly - \$1. - \$10. (Free to control equipment professionals).

Automatic Data Processing (British) Gen Inf
 Business Publications, Ltd.
 Mercury House
 109-119 Waterloo Road
 London, S. E. 1, England
 Monthly - 5s. - \$9.50

Automation Rel Eqt,
 The Penton Publishing Co. (b)
 Penton Bldg., Cleveland 13, Ohio
 Monthly - \$1. - \$10. (Free to automation management and engineers).

Bell Laboratories Record Data Tr
 Bell Telephone Laboratories, Incorporated
 463 West Street
 New York 14, New York
 Monthly - No single copy price - \$2.

Bell System Technical Journal Data Tr
 American Telephone and Telegraph Company
 195 Broadway
 New York 7, New York
 6 times a year - \$1.25 - \$5.

BIT: See Nordisk Tidskrift for Informations-Behandling

British Communications and Electronics (British) Data Tr,
 Heywood and Co., Ltd. Rel Eqt,
 527 Madison Avenue (c)
 New York 22, New York
 Monthly - 2s. 6d. - \$6.

Business Automation Gen Inf,
 Business Publications, Inc. Bus App,
 288 Park Avenue West DP Desc,
 Elmhurst, Illinois (d)
 Monthly - \$0.75 - \$5.

Business Week Gen Inf,
 McGraw-Hill Publishing Company, (e)
 Inc.
 330 West 42nd Street
 New York 36, New York
 Weekly - \$0.50 - \$7.

Chiffres (French) Sci App,
 Association Francaise de Calcul Theo Alg,
 98 bis, boulevard Arago Prog Sys,
 Paris (XIV^e), France Inf Ret,
 Quarterly - No price given. DP Log

Communication and Electronics (Part I, AIEE Transactions) Data Tr,
 American Institute of Electrical DP Log,
 Engineers Rel Eqt
 345 East 47th Street
 New York 17, New York
 Bimonthly - \$1.50 - \$8.
 (\$5. to members).

* Subject to change.

<u>Computer Abstracts (British)</u> Technical Information Company, Ltd. Chancery House, Chancery Lane London W. C. 2, England Monthly - No single copy price - £25.4.0	(f)	<u>Data Processing (British)</u> Iliffe Production Publications, Ltd. Dorset House, Stamford Street London S. E. 1, England Quarterly - 25s. - \$12.	Bus App, DP Desc
<u>Computer Bulletin (British)</u> British Computer Society, Ltd. Finsbury Court, Finsbury Pavement London E. C. 2, England Quarterly - No single copy price - £0.15.0	Gen Inf	<u>Data Processing</u> American Data Processing, Inc. 22nd Floor, Book Tower Detroit 26, Michigan Monthly - \$0.75 - \$7.50	Gen Inf, Bus App
<u>Computer Journal (British)</u> The British Computer Society Finsbury Court, Finsbury Pavement London E. C. 2, England Quarterly - 15s. Od. - \$7. (may be ordered in U. S. from ACM; see ACM Communications for address).	Bus App, Sci App, Prog Sys, DP Desc	<u>Data Processing Digest</u> Data Processing Digest, Inc. 1140 South Robertson Boulevard Los Angeles 35, California Monthly - \$3. - \$24.	Gen Inf, (j)
<u>Computer News (British)</u> Technical Information Company, Ltd. Chancery House, Chancery Lane London W. C. 2, England Monthly - No price given.	Gen Inf	<u>Data Processing Information for the Consulting and Accounting Professions</u> IBM - Data Processing Division 112 East Post Road White Plains, New York Approx. 6 times per year - Free to consultants and accountants.	Gen Inf, Bus App, DP Desc
<u>Computers and Automation</u> Berkeley Enterprises, Inc. 815 Washington Street Newtonville 60, Massachusetts Monthly - No single copy price - \$15.	Gen Inf, (g)	<u>Data Processing Yearbook</u> (Formerly: Data Processing Annual). American Data Processing, Inc. 22nd Floor, Book Tower Detroit 26, Michigan Yearly - \$15. per copy.	Gen Inf, Bus App, DP Desc, (k)
<u>Computing News</u> Computing News Box 261, Thousand Oaks, California Semimonthly - \$0.35 - \$6.	Gen Inf, DP Desc	<u>Data Processor</u> IBM - Data Processing Division 112 East Post Road White Plains, New York Approx. 6 times per year - Free to customer management.	Bus App, DP Desc
<u>Computing Reviews</u> Publisher and address same as ACM Communications Bimonthly - \$2. (\$1.50 to members) - \$10. (free to ACM members).	(h)	<u>Datamation</u> F. D. Thompson Publications, Inc. 141 East 44th Street New York 17, New York Monthly - \$1. - Free to data proc- essing professionals; \$10. to others.	Gen Inf, Prog Sys, DP Desc
<u>Control Engineering</u> McGraw-Hill Publishing Company, Inc. 330 West 42nd Street New York 36, New York Monthly - \$0.75 - \$3. to control engineers and management.	Rel Eq, (i)	<u>EDP Weekly</u> Industry Reports, Inc. 1327 F Street, N. W. Washington 4, D. C. Weekly - No single copy price - \$75.	Gen Inf

<p><u>Electrical Communication</u> International Telephone and Telegraph Corporation 320 Park Avenue New York 22, New York Quarterly - \$0.50 - \$2.</p>	Data Tr	<p><u>IRE Transactions on Electronic Computers</u> The Institute of Radio Engineers, Inc. 1 East 79th Street New York 21, New York Bimonthly - \$4.50 (\$2.25 to members) - \$17. (\$4. to members).</p>	DP Log Rel Eq
<p><u>Electronic News</u> Fairchild Publications, Inc. 7 East 12th Street New York 3, New York Weekly - \$0.10 - \$3.</p>	Gen Inf	<p><u>Journal of Chemical Documentation</u> American Chemical Society 1155 Sixteenth Street, N.W. Washington 6, D. C. Quarterly - No single copy price - \$10. (\$7. to members).</p>	Sci App, Inf Ret
<p><u>Electronics</u> McGraw-Hill Publishing Company, Inc. 330 West 42nd Street New York 36, New York Weekly - \$0.75 - \$6. to electronics professionals.</p>	Rel Eq	<p><u>Journal of Machine Accounting</u> <u>Journal of Machine Accounting, Inc.</u> 1750 West Central Road Mount Prospect, Illinois Monthly - No single copy price - \$5.</p>	Bus App, Mgt Sci
<p><u>Harvard Business Review</u> Harvard Business Review Boston 63, Massachusetts Bimonthly - \$2. - \$10.</p>	Mgt Sci	<p><u>Management Science</u> The Institute of Management Sciences P.O. Box 626, Ann Arbor, Michigan Quarterly - \$2.50 - \$10. (\$8. to members).</p>	Mgt Sci, Theo Alg
<p><u>IBM Journal of Research and Development</u> International Business Machines Corporation 590 Madison Avenue New York 22, New York Quarterly - \$2. - \$5.</p>	(l)	<p><u>Mathematics of Computation</u> American Mathematical Society 190 Hope Street Providence 6, Rhode Island Quarterly - \$2.50 - \$8.</p>	Theo Alg
<p><u>The IBM Supplier</u> IBM - Data Processing Division 112 East Post Road White Plains, New York Approx. 6 times per year - Free to users of IBM Equipment.</p>	(m)	<p><u>Modern Office Procedures</u> The Industrial Publishing Corporation 812 Huron Road Cleveland 15, Ohio Monthly - \$0.75 - \$7.</p>	DP Desc
<p><u>Information and Control</u> Academic Press, Inc. 111 Fifth Avenue New York 3, New York Quarterly - No single copy price - \$13.50.</p>	(n)	<p><u>N.A.A. Bulletin</u> National Association of Accountants 505 Park Avenue New York 22, New York Monthly - No single copy price - \$10. for members only.</p>	Bus App
<p><u>Instruments and Control Systems</u> The Instruments Publishing Company, Inc. 1600 North Main Street Pontiac, Illinois Monthly - No single copy price - \$4. to instrumentation and control systems professionals.</p>	Rel Eq	<p><u>Nordisk Tidsskrift for Informations- Behandling</u>, known also as BIT (Danish) Regnecentralen Gamle Carlsbergvej 2 Copenhagen Valby, Denmark Quarterly - \$1.50 - \$5.</p>	Gen Inf

<p><u>Numerische Mathematik</u> (German) Springer-Verlag Heidelberger Platz 3 Berlin W. 35, Germany Irregular - No single copy price - DM 96 per volume of five numbers.</p>	Theo Alg	<p><u>Systems and Procedures Journal</u> Systems and Procedures Association 7th Floor, 145 Hudson Street New York 13, New York Bimonthly - \$1.50 - \$7.50.</p>	Gen Inf, Bus App, Mgt Sci
<p><u>Office Automation</u> (a loose-leaf DP handbook with monthly updatings). Automation Consultants, Inc. 155 Fifth Avenue New York 10, New York Cost of Handbook: \$37.50 - Updating for one year: \$37.50.</p>	Gen Inf, DP Desc	<p><u>Systems Management</u> Data Processing Publishing Corp. 200 Madison Avenue New York 16, New York Bimonthly - No single copy price - \$3.</p>	Gen Inf, Bus App, Inf Ref
<p><u>Office Equipment and Methods</u> (Canadian) Maclean - Hunter Publishing Co., Ltd. 481 University Avenue Toronto 2, Ontario, Canada Monthly - No single copy price - \$5.</p>	Gen Inf, Bus App	<p><u>Telecommunications, English</u> Edition of Elektrosvyaz (Russian) American Institute of Electrical Engineers Special Subscription Dept. 41 East 28th Street New York 16, New York Monthly - No single copy price - \$14.25.</p>	Data Tr
<p><u>Operational Research Quarterly</u> (British) Pergamon Press, Ltd. Headington Hill Hall Oxford, England Quarterly - \$2.80 - \$8.40.</p>	Mgt Sci, Theo Alg	<p><u>The Office</u> Office Publications, Inc. 232 Madison Avenue New York 16, New York Monthly - \$0.50 - \$5.</p>	(p)
<p><u>Operations Research</u> Operations Research Society of America Mount Royal and Guilford Avenues Baltimore 2, Maryland Bimonthly - \$2. - \$10.</p>	Mgt Sci, Theo Alg	<p><u>Zeitschrift fur Angewante Mathematik und Mechanik</u> (German) Akademie-Verlag, G. m. b. H. Mohrenstrasse 39, Berlin W. 8 Germany Monthly - No single copy price - DM 15.</p>	Theo Alg
<p><u>RCA Review</u> RCA Laboratories, Radio Corporation of America Princeton, New Jersey Quarterly - \$0.75 - \$2.</p>	DP Log, (o)	<u>Notes</u>	
<p><u>Society for Industrial and Applied Mathematics Journal</u> Society for Industrial and Applied Mathematics SIAM Publications, Box 7541 Philadelphia 1, Pennsylvania Quarterly - \$14. for back volumes - \$10.</p>	Theo Alg	<p>(a) Programmed instruction and teaching machines. (b) All aspects of automation and control in manufacturing. (c) General coverage of electronics technology. (d) Annual reference issue serves as directory of products and organizations. (e) Occasional DP industry items are timely. (f) Abstracts of computer and DP articles, books and patents. (g) June issue is "Computer Directory and Buyer's Guide". (h) Reviews all current computer publications. (i) Broad control systems coverage.</p>	
<p><u>Society for Industrial and Applied Mathematics (SIAM) Review</u> (Same as the Journal.)</p>			

Section	Area	Page
---------	------	------

08	02	06
----	----	----

- (j) Reviews selected DP books and articles.
- (k) Contains bibliographical index to articles in periodicals by title, author and subject.
- (l) Technical coverage of basic research.
- (m) Suggestions on design and use of cards, tapes, ribbons, forms, etc.
- (n) Information and control theory.

(o) Technical coverage of research in radio and electronics.

(p) Computer physical planning and accessories.

Proceedings of joint computer and other conferences are not shown on this list, but are often good sources of information on computer design, description and use from a technical viewpoint.

SELECTED PROFESSIONAL SOCIETIES

Many societies and organizations have been founded and exist for the purpose of advancing the theory, principles, practices and techniques of a particular subject area, maintaining professional standards and ethics, exchanging ideas and information, and acting as official spokesmen for their members.

Practitioners within given fields belong to societies for a number of reasons:

- To learn from the experience of others.
- To contribute toward fulfilling the society's goals and objectives.
- To participate in society activities and arrange for or make presentations to symposiums and meetings.

Participation in one or more societies of his choice increases a systems engineer's technical competence and helps him achieve higher-quality performance in his work.

The following list comprises professional organizations heavily involved in the data processing field, or in functional fields which use computer equipment extensively. It represents only societies of nationwide scope. The systems engineer should also investigate local or regional groups reflecting his area of interest.

The list is not an exhaustive one, but includes most of the large societies with direct appeal to the systems engineer in his various spheres of concentration. For example, the Association of Iron and Steel Engineers (AISE), American Nuclear Society (ANS), American Mathematical Society (AMS), and American Chemical Society (ACS) have not been listed, for they would be of value to relatively few systems engineers.

Several of these professional organizations (AICHE, AIIE, ASCE, ASME, IEEE) are located in the new Engineering Societies Building at 345 East 47th Street, New York City, near the United Nations Building. Jointly, they maintain a 180,000-volume technical library for reading and research. Special services such as literature searches are provided on a fee basis.

Certain other societies are of interest to the systems engineer, although membership is open on a company basis only:

1. National Association of Manufacturers (NAM) represents industry views on national and international subjects to government, acts as a clearinghouse for information, and conducts economic and business studies, among other activities. It sponsors the Institute of Industrial Relations and publishes NAM News weekly.

2. Electronic Industries Association (EIA) publishes a Weekly Report, International News, biweekly; Industrial Relations Department Digest, quarterly; Industry Fact Book, annually; and Production Source Code, annually.

Business Equipment Manufacturers Association (BEMA) publishes a Weekly News Bulletin.

For more complete coverage of professional organizations refer to:

1. Scientific and Technical Societies of the United States and Canada, National Research Council Publication No. 900, National Academy of Sciences, Washington, D. C., 1961.

2. National Organizations of the U. S., Volume I of the Encyclopedia of Associations, Gale Research Company, Detroit, Michigan, 1961.

The societies are initially shown in alphabetic order. Information is arranged as follows:

- Society name and abbreviation
- Address
- One paragraph statement covering:
 1. Purpose and objectives.
 2. Publications, frequency of publication, and annual cost.
 3. Membership qualifications or types of individuals and groups belonging to the organization.

American Documentation Institute (ADI)
c/o Library of Congress
Washington 25, D. C.

Advances principles and techniques of communicating recorded knowledge. Publishes American Documentation, quarterly, \$6.50. Members include individuals, societies, associations, government agencies, educational institutions, industrial and business organizations interested in promoting documentation techniques.

American Institute of Chemical Engineers (AICHE)
345 East 47th Street
New York 17, New York

Advances theory and practice of chemical engineering and maintains high professional standards among members. Publishes Chemical Engineering Progress, monthly, \$6; AICHE Journal, quarterly, \$6; Symposium Series, irregular. Members must be proficient in chemical engineering and actively engaged in the practice of the profession.

American Institute of Industrial Engineers (AIIE)
345 East 47th Street
New York 17, New York

Maintains practices of the profession on a high level; fosters a high degree of integrity among members; encourages and assists education and research in the profession; and aids in the identification of qualified industrial engineers. Publishes Journal of Industrial Engineering, bimonthly, \$10. Membership is at two levels: senior and associate. Graduates and nongraduates in engineering and associated fields may qualify if they have equivalent industrial experience ranging from one to thirteen years.

American Management Association (AMA)
1515 Broadway
New York 36, New York

Provides training, research, publications and information services for business managers. Organizes and encourages exchange of thinking and experience within the management profession. Publishes Manager's Letter, monthly; Management News, monthly; Management Review, monthly; Supervisory Management, monthly; Personnel, bimonthly; Management Reports, periodically; and Research Studies, periodically.

American Production and Inventory Control Society (APICS)
330 South Wells Street
Chicago 6, Illinois

Promotes interest and advancement in production and inventory control techniques, practices and procedures, and in the development of materials personnel and materials planning and control systems. Publishes APICS News, monthly, and APICS Quarterly Bulletin. Membership includes individuals active in production and inventory control management, operations, education, and techniques development.

American Society for Quality Control (ASQC)
161 West Wisconsin Avenue
Milwaukee 3, Wisconsin

Creates, promotes, and stimulates interest in the advancement and diffusion of quality control knowledge and its application to industrial processes. Publishes Industrial Quality Control, monthly, \$10. Free to members. Membership is at three levels: associates, fellows and members. Includes graduates of recognized universities, and others with four years of experience in quality, inspection, or statistics fields.

American Society of Civil Engineers (ASCE)
345 East 47th Street
New York 17, N. Y.

Advances the science and profession of civil engineering. Publishes Civil Engineering, monthly, \$5; Proceedings, monthly, \$40; Transactions, annually, \$12; Directory, annual, \$2; and others. Membership is at several levels, each with specialized requirements.

American Society of Mechanical Engineers (ASME)
345 East 47th Street
New York 17, N. Y.

Conducts research, develops boiler, pressure vessel, and power test codes, and serves as sponsor for the American Standards Association in developing safety codes and standards for equipment. Publishes Mechanical Engineering, monthly, \$7 (\$3.50 to members); Transactions, quarterly; and Applied Mechanics Reviews, monthly, \$25. Membership is at various levels, each with specialized requirements (including honorary members).

American Society of Tool and Manufacturing Engineers (ASTME)
10700 Puritan Avenue
Detroit 38, Michigan

Further research in tool engineering and advances the scientific and educational progress of creative manufacturing. Publishes The Tool Engineer, monthly.

Armed Forces Communications and Electronics Association (AFCEA)
1624 Eye Street, N. W.
Washington 6, D. C.

Maintains and improves cooperation between the armed forces and industry in communications, design, production, maintenance and operation of communication, electronic and photographic equipment, and fosters the development of scientists and engineers in the USA. Publishes Signal, monthly, \$7 (\$5 to members). Membership is open to industrial organizations and interested individuals in continental United States and abroad.

Association for Computing Machinery (ACM)
211 East 43rd Street
New York 17, N. Y.

Advances the science, design, development, construction and application of modern machinery for performing operations in mathematics, logic, statistics and kindred fields. Publishes Communications of the Association for Computing Machin-

Section	Area	Page
08	03	03

ery, monthly, \$10; Journal of the ACM, quarterly, \$10; and Computing Reviews, \$10. Publications are free to members, who may be any persons interested in the field.

Data Processing Management Association (DPMA)
Park Ridge, Illinois

Aids members (data processing management, operating and systems personnel) in developing their professional capabilities, improves public understanding of data processing and its benefits, and assists in establishing suitable industry standards. Publishes Journal of Data Management, monthly. Membership includes tabulating supervisors, programmers, systems engineers, data processing managers and others associated with the computer data processing field.

Econometric Society (ES)
Box 1264, Yale Station
New Haven, Connecticut

Advances economic theory in relation to statistics and mathematics. Publishes Econometrica, quarterly. Membership includes economists, statisticians and mathematicians.

Institute of Electrical and Electronic Engineers (IEEE)*
345 East 47th Street
New York 17, N. Y.

Advances the theory and practice of electrical engineering, electronics, radio, allied branches of engineering, through meetings, papers, and publications; and related arts and sciences. Maintains high professional standing among members. Publishes Electrical Engineering, monthly, \$12 (\$6 to members); Proceedings of IRE, monthly, \$18 (free to members); Communications and Electronics, bimonthly; Applications and Industry, bimonthly; IRE Student, quarterly; Transactions of Professional Groups, irregularly; and numerous other publications. Membership is maintained at various levels, with specialized requirements for each one.

Instrument Society of America (ISA)
313 Sixth Avenue
Pittsburgh 22, Pennsylvania

Advances the arts and sciences related to the theory, design, manufacture and use of instruments and controls in the various technologies; disseminates information; stimulates educational

activities and the development of standards within the instrumentation technology. Publishes ISA Journal, monthly; Transactions, annually; ISA Proceedings, irregularly; Recommended Practices, irregularly; Electrical Safety Abstracts; Measurement Techniques, monthly; Instruments and Experimental Techniques, bimonthly; Automation and Remote Control, monthly; and Industrial Laboratory, monthly.

National Association of Accountants (NAA)
505 Park Avenue
New York 22, N. Y.

Conducts research on accounting methods and procedures; maintains library and employment services. Publishes NAA Bulletin, monthly, and Accounting Practice Reports, irregularly. Membership includes nonprofessional and professional accountants in industry, public accounting, government and teaching, and other personnel interested in internal and management uses of accounting.

National Office Management Association (NOMA)
1927 Old York Road
Willow Grove, Pennsylvania

Promotes application of scientific methods to commerce and industry for the purpose of increasing productivity, lowering costs and improving quality; assists institutions in the educational field in developing training programs and courses of study; encourages and participates in research; promotes sound employee and employer relationships; audits hundreds of corporate businesses annually and classifies them in order of excellence; maintains library. Publishes Office Executive, monthly; Proceedings, annually; and Survey Summaries, irregularly. Membership includes office managers, supervisors, personnel men, educators, equipment manufacturers, and utility company executives.

Operations Research Society of America (ORSA)
Mt. Royal and Guilford Avenues
Baltimore 2, Maryland

Advances the science of operations research through exchange of information, establishment and maintenance of professional standards of competence, improvement of methods and techniques, and encouragement and development of students. Publishes Operations Research, bimonthly, \$10, and Bulletin of the ORSA, semi-annually. To qualify for membership, an individual should have at least two years' experience performing effective and original operations research work, or management responsibility for operations research for at least one year.

*This society is newly merged from Institute of Radio Engineers (IRE) and American Institute of Electrical Engineers (AIEE). Publications continue as shown for 1963, but will be changed sometime in 1964.

Society for Advancement of Management (SAM)
74 Fifth Avenue
New York 11, New York

Sponsors numerous conferences, study groups and seminars; provides advisory service to local governments, civic and nonprofit institutions, including projects for hospitals; develops standards and examinations for professional industrial engineers; furnishes data to state registration boards in establishing engineering license requirements; grants fellowships and awards for various aspects of management; maintains consulting and information services, reference library and placement program. Publishes SAM Newsletter, weekly, and Advanced Management, monthly. Membership includes management executives in industry, commerce, government and education.

Society for Industrial and Applied Mathematics (SIAM)

P. O. Box 7541

Philadelphia 1, Pennsylvania

Furtheres the application of mathematics to industry and science; promotes basic research in mathematics, leading to new methods and techniques useful to industry and science; and provides media for the exchange of information and ideas between mathematicians and other technical

and scientific personnel. Publishes Journal, quarterly, \$10; SIAM Review, quarterly, \$10; and Theory of Probability and Its Application, quarterly, \$18. Membership is open to individuals and institutions qualified to further the objectives of the society.

Systems and Procedures Association (SPA)
4463 Penobscot Building
Detroit 26, Michigan

Promotes standardization and simplification of office accounting equipment, forms, systems and procedures, in business, education, government, and the military services. Publishes Systems and Procedures Magazine, Ideas for Management and International Newsletter. Membership includes specialists and supervisors in office management.

The Institute of Management Sciences (TIMS)

P. O. Box 273

Pleasantville, N. Y.

Identifies, extends, and unifies scientific knowledge that contributes to understanding and practice of management. Publishes Management Science, quarterly, \$10, and Management Technology. Membership is open to persons interested in promoting the practice and growth of management sciences.

NUMBER BASE CONVERSION

When working with machines which represent values by means of nondecimal number systems (principally binary), the decimal input which is to be operated upon has to be converted arithmetically to the machine representation, and vice versa for output. Automatic conversion is not generally built into machines, owing to the problem of scaling -- that is, the effect of varying positions of the radix point (decimal point, binary point, etc., depending upon the number system involved). However, independent conversion subroutines, as well as conversion facilities within comprehensive programming systems, are provided for nondecimal machines. These programs accomplish conversion by many variations on the basic principles which follow. The nature of the specific features and instructions (like indexing or table lookup) of the machines for which the programs have been written dictates the variations.

The usual way of writing a number does not explicitly use the base or radix, but only the digits D_i , and a radix point, as

$$D_n D_{n-1} \dots D_1 D_0 \cdot D_{-1} D_{-2} \dots D_{-m}$$

↑
(Radix Point)

Fundamental to nearly all number base conversion is the explicit representation in terms of both the base R , and the digits, as

$$D_n R^n + D_{n-1} R^{n-1} + \dots + D_0 R^0$$

$$+ D_{-1} R^{-1} + \dots + D_{-m} R^{-m}$$

(Recall that $R^0 = 1$)

Examples (bases and exponents always expressed in the decimal system):

1. Decimal ($R = 10$):

$$637.25 = 6 \times 10^2 + 3 \times 10^1 + 7 \times 10^0$$

$$+ 2 \times 10^{-1} + 5 \times 10^{-2}$$

2. Binary ($R = 2$):

$$110101. = 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3$$

$$+ 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$

The principles of number base conversion are grouped into four logical cases in the following tabulation. In addition to displaying the logical foundations of conversion programs, these principles suggest paper-and-pencil procedures for the systems engineer who encounters numerical data in nondecimal form during debugging, etc. To apply this information properly, note that

1. Mixed numbers are converted by applying, respectively, the appropriate techniques to the integral and fractional parts.

2. To preserve significance in conversions, one decimal digit approximately equals 3.3 binary digits in information content.

3. Common notation represents the base subscripted following the number and a right parenthesis -- for example, $275)_{10}$; $101)_{2}$; $275)_{8}$.

OTHER BASE TO DECIMAL

(Binary used for "other base" in all examples)

Type I: Whole Numbers

Given $D_n D_{n-1} \dots D_1 D_0$, substitute the values of the D_i as given, and the decimal value of R into the expression

$$D_n R^n + D_{n-1} R^{n-1} + \dots + D_1 R^1 + D_0 R^0$$

Example: $101011)_{2} = ?)_{10}$

$$101011)_{2} = 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$$

$$= 32 + 8 + 2 + 1 = 43)_{10}$$

A short cut for paper-and-pencil use:

Add vertical columns and multiply by the base in the direction of the arrows:

1	0	1	0	1	1
1	2	5	10	20	42
	2	5	10	21	43

Type II: Fractions

Given $.D_{-1} D_{-2} \dots D_{-m}$, substitute the values of the D_i as given, and the decimal value of R into the expression

$$\frac{D_{-1} R^{m-1} + D_{-2} R^{m-2} + \dots + D_{-m} R^0}{R^m}$$

Example: $.110111)_2 = (?)_{10}$

$$.110111)_2 = \frac{1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0}{2^6}$$

$$= \frac{32 + 16 + 4 + 2 + 1}{64} = \frac{55}{64}$$

= approximately $.86)_{10}$ and is accurate to an equivalent number of places.

To simplify finding the numerator, when using paper and pencil, use the short cut shown for converting whole numbers to decimals.

DECIMAL TO OTHER BASE

Type III: Whole Numbers

Given decimal integer N, write

$$N = D_n R^n + D_{n-1} R^{n-1} + \dots + D_1 R^1 + D_0 R^0$$

Divide by R:

$$N \div R = (D_n R^{n-1} + \dots + D_1 R^0) + \frac{D_0}{R}, \text{ so low -}$$

order digit D_0 is obtained as remainder. Similarly, D_1 appears after a division of the quantity in parentheses by R, and so on.

Example: $43)_{10} = (?)_2$

2	43	Remainders	} Collect remainders in reverse order: $43)_{10} = 101011)_2$
2	21	1	
2	10	1	
2	5	0	
2	2	1	
2	1	0	
	0	1	

Type IV: Fractions

Given decimal fraction F, write

$$F = D_{-1} R^{-1} + D_{-2} R^{-2} + \dots + D_{-m} R^{-m}$$

Multiply by R: $F \times R = D_{-1} + (D_{-2} R^{-1} + \dots$

$+ D_{-m} R^{-m+1})$, so high-order digit D_{-1} is obtained as integer portion of the product. Similarly, D_{-2} appears after a multiplication of the quantity in parentheses (fractional part of previous product) by R, and so on.

Example: $.37)_{10} = (?)_2$

$\begin{array}{r} .37 \\ 2 \\ \hline 0 \end{array} \overline{) .74}$	$\begin{array}{r} .96 \\ 2 \\ \hline 1 \end{array} \overline{) .92}$	$\begin{array}{r} .68 \\ 2 \\ \hline 1 \end{array} \overline{) .36}$
$\begin{array}{r} 2 \\ \hline 1 \end{array} \overline{) .48}$	$\begin{array}{r} 2 \\ \hline 1 \end{array} \overline{) .84}$	$\begin{array}{r} 2 \\ \hline 0 \end{array} \overline{) .72}$ is rounded to 1 since the remainder exceeds $1/2$.
$\begin{array}{r} 2 \\ \hline 0 \end{array} \overline{) .96}$	$\begin{array}{r} 2 \\ \hline 1 \end{array} \overline{) .68}$	

Collect integer parts of products:

$.37)_{10} =$ approximately $.0101111)_2$ and is accurate to the equivalent number of places.

BIBLIOGRAPHY

Crowder, Norman A., The Arithmetic of Computers - A Tutortext, Doubleday, 1960. This is a programmed text of the branching type (i.e., separate reading paths for right and wrong answers).

McCracken, D. D., Digital Computer Programming, John Wiley and Sons, 1957, Chapter 3.

Grabbe, E. M., and Simon Ramo, and D. E. Wooldridge, eds., Handbook of Automation, Computation, and Control, Volume 2, John Wiley and Sons, 1959, Chapters 2, 12.

Culbertson, James T., Mathematics and Logic for Digital Devices, D. Van Nostrand, 1958, Chapter 4.

IDENTIFICATION OF MATHEMATICAL ERRORS

Precision.-- As a computer concept, precision generally refers to the number of places to which it is possible to carry out an arithmetic operation. The longer the word length of a computer, the greater the precision with which numbers can be represented.

Double-precision arithmetic.-- In fixed word-length computers, the number of places in arithmetic operations can be extended by using two machine words.

Significance.-- The digits in a numeric quantity which are meaningful are referred to as significant digits. Zeros, for example, may be used to indicate only the position of the decimal point and as such are not significant. Each of the following numbers contains four significant digits: 3.142, .003076, 65.00.

Any number derived by measuring (as contrasted with counting) is an approximation of the exact or true value. This approximation must be accurate enough so that when the number is carried out to one more decimal position, the accuracy of the added digit is ± 5 ; otherwise not all digits of the original number are significant. The number 49.4, for example, is between 49.35 and 49.44. If the true measurement is within this range, the number 49.4 has three significant digits. On the other hand, if the true measurement were 49.49, then the first digit to the right of the decimal point would not be significant.

Absolute error.-- The absolute error in an approximate number is the absolute value of the difference between the approximate number and the true value of the number being approximated. If, for example, a value $A=5.73568$ is approximated by (or rounded to) $A=5.7$, then the absolute error in the rounded number is .03568.

Relative error.-- The relative error in an approximate number is the absolute error divided by the absolute value of the number being approximated.

If, for example, $A=5.73568$ and this value is approximated by the number 5.7, then the relative error in the approximated number is $.03568/5.73568=.0062$.

Percent error.-- The quotient of the error divided by the correct result multiplied by 100% is the percent error. Relative error is commonly converted to percent error. In the example above, the percent error is 0.62%.

Initial error.-- The difference between the true value and the approximation before any computation is the initial error.

Rounding error.-- This type of error results when the less significant digits of a quantity are deleted and a rule of correction is applied to the remaining part. For example, pi, 3.14159265. . . rounded to four decimals is 3.1416.

Truncation error.-- Simply deleting digits to the right of the decimal point without applying a rule of correction is truncation. Expressing pi, for example, as 3.1415, will result in a truncation error. When an infinite mathematical process is approximated by a finite number of calculations, the results will contain a truncation error. For example, cutting the series for e^x at

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!}$$

will produce a truncation error (sometimes called residual error for series approximations).

Propagated error.-- When an initial error is increased by computation, the resulting error is called a propagated error.

Errors in the model.-- A collection of equations that simulate mathematically all that goes on in a process is called a mathematical model. Errors may arise from inadequate knowledge or erroneous assumptions about the situation being represented mathematically.



BASIC MATHEMATICAL TERMS

Array is a series of numbers or terms called elements. One- and two- dimensional arrays are the most common.

Examples:

One-dimensional array or sequence

$$3, 7, 2, 5 \quad \text{or} \quad x_1, x_2, x_3$$

Two-dimensional array or matrix

$$\begin{pmatrix} 1 & 3 & 9 & 4 \\ 2 & 7 & 10 & 5 \\ 4 & 16 & 3 & 12 \end{pmatrix} \quad \text{or} \quad \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

Subscripts (as n in x_n) are used to identify specific elements in an array. The number of subscripts is the same as the number of dimensions of the array.

Example of double subscripting in a matrix:

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$

In this example the general term can be represented as a_{ij} , where i indicates row number and j represents column number.

A function is a relationship associating some value of a dependent variable or argument with admissible values of the independent variable(s) or argument(s).

Notation: $y = f(x)$

where:

- y is the dependent variable
- f is the function
- x is the independent variable

Examples:

$$\text{theta} = \sin f(\text{beta}); \text{weight} = f(\text{age, height})$$

The absolute value of a number x is the value of x without regard to its sign.

Notation: $|x|$

Examples: $|-17| = 17; |17| = 17$

An arithmetic progression is a sequence of terms each of which is equal to the preceding plus a fixed constant.

Example:

$$a, a + d, a + 2d, a + 3d, \dots$$

$$\text{The } n^{\text{th}} \text{ term} = a + (n - 1)d$$

$$\begin{aligned} \text{The sum of the first } n \text{ terms} &= \frac{n}{2} (\text{first term} \\ &+ \text{last term}) \\ \text{or} \\ &= \frac{n}{2} (2a + (n - 1)d) \end{aligned}$$

A geometric progression is a sequence of terms such that the ratio of each term to the immediately preceding one is the same throughout the sequence.

Example:

$$a, ar, ar^2, ar^3, \dots, ar^{n-1}$$

where:

- a is the first term
- r the common ratio
- ar^{n-1} the last or n^{th} term

$$\text{The sum of the first } n \text{ terms} = \frac{a(1 - r^n)}{1 - r}$$

The sum of an infinite number of terms of a geometric progression is finite only if $|r| < 1$. In this case, the sum is:

$$\frac{a}{1 - r}$$

A series is a sum of elements of a sequence. Standard notation for such a sum utilizes the sigma symbol with the subscripts of the elements summarized below and above this symbol:

$$\sum_{i=1}^n x_i = x_1 + x_2 + \dots + x_n$$

Here, the left-hand side of the equality is read, "The sum of the x sub i for i ranging from 1 to n", with i being called the index of summation.

The greatest integer contained in a number x is the largest integer (i. e., whole number) less than or equal to x .

Notation: $[x]$

Examples:

$$[5.7] = 5; [-3.8] = -4$$

The factorial of a number n (n must be a positive integer) is the product of all positive integers less than or equal to the integer n .

Notation:

$$n! \text{ or } [n = n(n-1)(n-2)(n-3) \dots (3)(2)(1)]$$

Example:

$$5! = (5)(4)(3)(2)(1) = 120$$

$$[3] = (3)(2)(1) = 6$$

An exponent (as n in a^n) may be any positive or negative number. If an exponent is a positive whole number (i. e., integer), it denotes the number of times that the number to which it is affixed (i. e., base) is to be taken as a factor in multiplication.

Example: $4^3 = (4)(4)(4) = 64$

The laws of exponents define the meaning of zero, negative and fractional exponents: For all values of a , and all positive integers m, n ,

$$a^0 = 1; \quad a^{-n} = \frac{1}{a^n};$$

$$a^m \cdot a^n = a^{m+n}; \quad (a^m)/(a^n) = a^{m-n};$$

$$(a^m)^n = a^{mn}; \quad \sqrt[n]{a^m} = a^{m/n} = \left(\sqrt[n]{a}\right)^m$$

The logarithm of a number, N , with respect to the base, b , is the power (or exponent), x , to which the base must be raised to give the number.

Notation: $x = \log_b N$. This means $b^x = N$.

Examples:

$$\log_{10}(1000) = 3; \log_{10}(0.1) = -1; \log_2 32 = 5$$

Common (Napierian) logarithms have base 10.

Notation: $\log x$

Natural (Napierian) logarithms have base $e = 2.71828182 \dots$

Notation: $\log_e x$ or $\ln x$

Laws of logarithms: For all positive values of b, M, N ,

$$\log_b 1 = 0$$

$$\log_b (MN) = \log_b M + \log_b N$$

$$\log_b (M/N) = \log_b M - \log_b N$$

$$\log_b M^N = N (\log_b M)$$

DATA REPRESENTATION CODES

The following chart of character codes, used in various IBM machines and various standard media, illustrates several points:

1. Not all machines are designed to handle the maximum number of characters that are theoretically possible within the code structure of the machine.

2. Different machines may have different-size character sets and may have different BCD code structures for a given character (for example, IBM 1401 vs. IBM 1620).

3. The codes for some machines carry redundant data to permit automatic error detection (for example, IBM 1401).

4. Most codes preserve the sequences 0-9 and A-Z. It should be noted that data processing machines will sort records of information by subtraction or comparison of the field controlling the

sequence. In either case the result will depend on a previously established relative sequence (that is, collating sequence) of the character code within the machine. In the IBM 1401, for instance, the ascending sequence of characters for collating is:

BLANK	#
.	@
∩	?
≠	A THROUGH I
&	!
\$	J THROUGH R
*	≠
-	S THROUGH Z
/	O THROUGH 9
,	
%	

This sequence will vary from machine to machine.

Graphic Character ¹	Notes	1620/1710 ²		1401/1410/1420 1440/1460/7010		7070/7072/7074 ²		705/7080	704/709/7040 7044/7090/7094	IBM Standard BCD Interchange	IBM Preferred Printing Arrangement	IBM Card Code Zone Number		Eight-Track Paper Tape	Teletype ³ Baudot	Fielddata ⁴
		CF8421	CF8421	CBA8421	01236	01236	CBA8421	BA8421	BA8421	A	H	12-11-0	1-9	XOC8421	L/F 12345	543210
A		4	1	BA 1	0 6 01	CBA 1	A 1	BA 1	A A	12	1	XO 1	L 12	21		
B		4	2	BA 2	0 6 0 2	CBA 2	A 2	BA 2	B B	12	2	XO 2	L 1 45	210		
C		4	C 21	CBA 21	0 6 0 3	BA 21	A 21	BA 21	C C	12	3	XOC 21	L 234	3		
D		4	4	BA 4	0 6 1 3	CBA 4	A 4	BA 4	D D	12	4	XO 4	L 1 4	3 0		
E		4	C 4 1	CBA 4 1	0 6 2 3	BA 4 1	A 4 1	BA 4 1	E E	12	5	XOC 4 1	L 1	3 1		
F		4	C 42	CBA 42	0 6 0 6	BA 42	A 42	BA 42	F F	12	6	XOC 42	L 1 34	3 10		
G		4	421	BA 421	0 6 1 6	CBA 421	A 421	BA 421	G G	12	7	XO 421	L 2 45	32		
H		4	8	BA8	0 6 2 6	CBA8	A8	BA8	H H	12	8	XO 8	L 3 5	32 0		
I		4	C 8 1	CBA8 1	0 6 3 6	BA8 1	A8 1	BA8 1	I I	12	9	XOC8 1	L 23	321		
J		C 4 1	1	CB 1	1 6 01	B 1	B 1	B 1	J J	11	1	X C 1	L 12 4	3210		
K		C 4 1	2	CB 2	1 6 0 2	B 2	B 2	B 2	K K	11	2	X C 2	L 1234	4		
L		C 4 1 C	21	B 21	1 6 0 3	CB 21	B 21	B 21	L L	11	3	X 21	L 2 5 4	0		
M		C 4 1	4	CB 4	1 6 1 3	B 4	B 4	B 4	M M	11	4	X C 4	L 345	4 1		
N		C 4 1 C	4 1	B 4 1	1 6 2 3	CB 4 1	B 4 1	B 4 1	N N	11	5	X 4 1	L 34	4 10		
O		C 4 1 C	42	B 42	1 6 0 6	CB 42	B 42	B 42	O O	11	6	X 42	L 45	4 2		
P		C 4 1	421	CB 421	1 6 1 6	B 421	B 421	B 421	P P	11	7	X C 421	L 23 5	4 2 0		
Q		C 4 1	8	CB 8	1 6 2 6	B 8	B 8	B 8	Q Q	11	8	X C8	L 123 5	4 21		
R		C 4 1 C	8 1	B 8 1	1 6 3 6	CB 8 1	B 8 1	B 8 1	R R	11	9	X 8 1	L 2 4	4 210		
S		C 42	2	C A 2	2 6 0 2	A 2	BA 2	A 2	S S	0	2	OC 2	L 1 3	43		
T		C 42 C	21	A 21	2 6 0 3	C A 21	BA 21	A 21	T T	0	3	O 21	L 5	43 0		
U		C 42	4	C A 4	2 6 1 3	A 4	BA 4	A 4	U U	0	4	OC 4	L 123	43 1		
V		C 42 C	4 1	A 4 1	2 6 2 3	C A 4 1	BA 4 1	A 4 1	v v	0	5	O 4 1	L 2345	43 10		
W		C 42 C	42	A 42	2 6 0 6	C A 42	BA 42	A 42	w w	0	6	O 42	L 12 5	432		
X		C 42	421	C A 421	2 6 1 6	A 421	BA 421	A 421	x x	0	7	OC 421	L 1 345	432 0		
Y		C 42	8	C A8	2 6 2 6	A8	BA8	A8	Y Y	0	8	OC8	L 1 3 5	4321		
Z		C 42 C	8 1	A8 1	2 6 3 6	C A8 1	BA8 1	A8 1	Z Z	0	9	O 8 1	L 1 5	43210		
0	Zero	421 C		C 8 2	36 12	8 2		8 2	0 0	0		O	F 23 5	54		
1		421	1	1	36 01	C 1	1	1	1 1	1	1	1	F 123 5	54 0		
2		421	2	2	36 0 2	C 2	2	2	2 2	2	2	2	F 12 5	54 1		
3		421 C	21	C 21	36 0 3	21	21	21	3 3	3	3	C 21	F 1	54 10		
4		421	4	4	36 1 3	C 4	4	4	4 4	4	4	4	F 2 4	54 2		
5		421 C	4 1	C 4 1	36 2 3	4 1	4 1	4 1	5 5	5	5	C 4 1	F 5	54 2 0		
6		421 C	42	C 42	36 0 6	42	42	42	6 6	6	6	C 42	F 1 3 5	54 21		
7		421	421	421	36 1 6	C 421	421	421	7 7	7	7	421	F 123	54 210		
8		421	8	8	36 2 6	C 8	8	8	8 8	8	8	8	F 23	543		
9		421 C	8 1	C 8 1	36 3 6	8 1	8 1	8 1	9 9	9	9	C8 1	F 45	543 0		

¹Graphic printer output depends upon the characters available in the typeset of an individual machine.

²Single-digit numerical representation is normally used except when alphabetic or special characters are present. Single-digit representation is the right-hand digit in the double-digit code. The "F" bit in the 1620 serves as a minus sign for numerical data.

³The teletype code is named after its French inventor, Baudot. It is not checked. To double the 32-character capacity of 5-track teletype tape, a shift character is punched in the tape to describe all the characters that follow it. An "F" indicates a "Figures" shift character preceding the code shown. An "L" indicates a preceding "Letters" shift. No indication means that either shift gives the same results.

⁴Fielddata code is standard for data processing and communications in the U.S. Army.

Graphic Character ¹	Notes	1620/1710 ²		1401/1410/1420 1440/1460/7010	7070/7072/7074		705/7080	704/709/7040 7044/7090/7094	IBM Standard BCD Interchange	IBM Preferred Printing Arrangement		IBM Card Code Zone Number		Eight-Track Paper Tape	Teletype ³ Baudot	Fielddata ⁴
		CF 8 4 2 1	CF 8 4 2 1	CBA 8 4 2 1	0 1 2 3 6	0 1 2 3 6	CBA 8 4 2 1	BA 8 4 2 1	BA 8 4 2 1	A	H	12-11-0	1-9	XOC 8 4 2 1	L/F 1 2 3 4 5	5 4 3 2 1 0
Blank	Nonprinting blank, Space	C	C	C	12	12	C A	BA					No punches	C	3	2 0
# =	Number sign; Equal sign	C	2 1 C 2 1	8 2 1	1 3	2 3	C 8 2 1	8 2 1	8 2 1	# =		3-8	8 2 1	F 3 5	5 2	
@	"At" sign; Apostrophe	C	2 1 4	C 8 2	1 3	0 6	8 4	8 4	8 4	@		4-8	C 8 4	F 1 2 4	5 4 3 1	
:	Colon			8 4 1	1 3	1 6	C 8 4 1	8 4 1	8 4 1			5-8		F 2 3 4	5 3 1 0	
>	"Greater than" symbol			8 4 2	1 3	2 6	C 8 4 2	8 4 2	8 4 2			6-8			5 2 0	
√	Tape mark, Line feed			C 8 4 2 1	1 3	3 6	8 4 2 1	8 4 2 1	8 4 2 1			7-8		2	1 0	
␣	Printing blank			A			8 2	A	A			2-8				
/	Slash, Division symbol	2	1	C A 1	0 3	0 1	A 1	BA 1	A 1	/ /		0 1	OC 1	F 1 3 4 5	5 4 3 2	
␣	Record mark, Carriage return	C	C 8 2	A 8 2	2 6	1 2	C A 8 2	BA 8 2	A 8 2	␣ ␣		0 2-8	O 8 2	4	2	
,	Comma	2	C 2 1	C A 8 2 1	0 3	2 3	A 8 2 1	BA 8 2 1	A 8 2 1	,		0 3-8	OC 8 2 1	F 3 4	5 3 2 1	
% (Percent; Left parenthesis	2	4	A 8 4	0 3	0 6	C A 8 4	BA 8 4	A 8 4	% (0 4-8	O 8 4	F 1 2 3 4	5 3 0	
γ	Word separator, "Figures" shift			C A 8 4 1	0 3	1 6	A 8 4 1	BA 8 4 1	A 8 4 1			0 5-8		1 2 4 5	0	
\	Reverse slash, Break			C A 8 4 2	0 3	2 6	A 8 4 2	BA 8 4 2	A 8 4 2			0 6-8			5 2 1	
++ "	Tape segment mark; Quotes			A 8 4 2 1	0 3	3 6	C A 8 4 2 1	BA 8 4 2 1	A 8 4 2 1			0 7-8		F 1 5	5 3 1	
-	Minus, Dash, Hyphen	2	C	B	0 3	1 2	CB	B	B	- -		1 1	X	F 1 2	5 0	
0 !	Minus zero; Exclamation point	C	4 1 C	B 8 2	1 6	1 2	CB 8 2	B 8 2	B 8 2			1 1-0		F 1 3 4	5 3 2 0	
\$	Dollar sign		1 C 2 1	CB 8 2 1	0 2	2 3	B 8 2 1	B 8 2 1	B 8 2 1	\$ \$		1 1 3-8	X C 8 2 1	F 1 4	5 2 1 0	
*	Asterisk, "Bell"	1	4	B 8 4	0 2	0 6	CB 8 4	B 8 4	B 8 4	* *		1 1 4-8	X 8 4	F 1 3	5 3	
]	Right bracket, Special			CB 8 4 1	0 2	1 6	B 8 4 1	B 8 4 1	B 8 4 1			1 1 5-8			5 4 3 2 1	
;	Semicolon			CB 8 4 2	0 2	2 6	B 8 4 2	B 8 4 2	B 8 4 2			1 1 6-8		F 2 3 4 5	5 4 3 1 0	
Δ	Delta (mode change), Stop			B 8 4 2 1			CB 8 4 2 1	B 8 4 2 1	B 8 4 2 1			1 1 7-8			5 3 2 1 0	
a +	Ampersand; Plus	1	C	CBA	0 2	1 2	BA	A	BA	a +		1 2	XOC	F 2 4 5	5 1	
? 0	Question mark; Positive zero			CBA 8 2	0 6	1 2	BA 8 2	A 8 2	BA 8 2			1 2-0		F 1 4 5	5 3 2	
.	Period, Decimal point	C	C 2 1	BA 8 2 1	0 1	2 3	CBA 8 2 1	A 8 2 1	BA 8 2 1			1 2 3-8	XO 8 2 1	F 3 4 5	5 4 3 2 0	
⌋)	Lozenge; Right parenthesis	C	4	CBA 8 4	0 1	0 6	BA 8 4	A 8 4	BA 8 4	⌋)		1 2 4-8	XOC 8 4	F 2 5	5	
[Left bracket, Idle			BA 8 4 1	0 1	1 6	CBA 8 4 1	A 8 4 1	BA 8 4 1			1 2 5-8			5 4 3 2 1 0	
<	"Less than" symbol			BA 8 4 2	0 1	2 6	CBA 8 4 2	A 8 4 2	BA 8 4 2			1 2 6-8			5 1 0	
␣	Group mark, "Letters" shift	C	C 8 4 2 1	CBA 8 4 2 1	0 1	3 6	BA 8 4 2 1	A 8 4 2 1	BA 8 4 2 1			1 2 7-8		1 2 3 4 5	1	

¹Graphic printer output depends upon the characters available in the typeset of an individual machine.

²Single-digit numerical representation is normally used except when alphabetic or special characters are present. Single-digit representation is the right-hand digit in the double-digit code. The "F" bit in the 1620 serves as a minus sign for numerical data.

³The teletype code is named after its French inventor, Baudot. It is not checked. To double the 32-character capacity of 5-track teletype tape, a shift character is punched in the tape to describe all the characters that follow it. An "F" indicates a "Figures" shift character preceding the code shown. An "L" indicates a preceding "Letters" shift. No indication means that either shift gives the same results.

⁴Fielddata code is standard for data processing and communications in the U. S. Army.



BOOLEAN ALGEBRA

George Boole (1815-1864), an English mathematician, constructed a logical algebra based upon his investigation of the laws of thought. C. E. Shannon, a research assistant at the Massachusetts Institute of Technology, was the first to use this logical algebra in the design of relay switching circuits. This logical algebra has become known as Boolean algebra.

Boolean variables can assume only two values. These may be thought of as "true" and "false" or "on" and "off". Frequently 1 is used to represent "true" or "on" or "up" and 0 to represent "false" or "off" or "down".

Symbolic Logic	Boolean Algebra	Elementary Electrical	Some Condition
TRUE	1	ON	Exists
FALSE	0	OFF	Does not exist

FUNDAMENTAL OPERATIONS ON LOGICAL (OR BOOLEAN) VARIABLES

Let A and B be such variables. Then

Name	Common Notations	Definition
NOT A	$\sim A, \bar{A}, A', -A$	Has the value 0 if A = 1; has the value 1 if A = 0.
A AND B	$A \wedge B, A \cdot B, AB, A \cap B$	Has the value 1 only if both A and B are 1; otherwise 0.
A OR B (inclusive)	$A + B, A \vee B, A \cup B$	Has the value 1 if A = 1 or B = 1 or A = B = 1. 0 only if A = B = 0.
A OR B (exclusive)	$A \oplus B, A \nabla B, A \vee\vee B, A \neq B$	Has the value 1 only if A = 1 or B = 1, but not both.

Truth tables are used either to define logical operations, or to verify that two logical operations are equivalent. For example, one large truth table could define all of the above operations. The five right-hand columns are regarded as dependent variables, whose values in each row are associated with the given values of A and B in the two left-hand columns. A and B are regarded as independent variables.

A	B	\bar{A}	\bar{B}	AB	A+B	$A \oplus B$
0	0	1	1	0	0	0
0	1	1	0	0	1	1
1	0	0	1	0	1	1
1	1	0	0	1	1	0

BASIC RELATIONS OF BOOLEAN ALGEBRA WHICH CAN BE VERIFIED BY MEANS OF A TRUTH TABLE

- $0 + A = A$
- $1 + A = 1$
- $A + \bar{A} = 1$
- $A + \bar{A} = 1$
- $0 \cdot A = 0$
- $1 \cdot A = A$
- $A \cdot A = A$
- $\underline{A} \cdot \bar{A} = 0$
- $\bar{\bar{A}} = A$
- $A + B = B + A$ (commutative law of addition)
- $A \cdot B = B \cdot A$ (commutative law of multiplication)
- $A + (B+C) = (A+B) + C$ (associative law of addition)
- $A(BC) = (AB)C$ (associative law of multiplication)
- $A(B+C) = AB + AC$ (distributive law)
- $A + BC = (A + B)(A + C)$
- $A + AC = A$
- $A(A+B) = A$
- $(A+B)(A+C) = A + BC$
- $A + \bar{A}B = A + B$

DEMORGAN'S THEOREMS

A basic duality which underlies all of Boolean algebra is that for every proposition about sums there is an analogous one about products. If there is a term which, when multiplied by any other term, leaves that term unaffected, then there is also one which may be added to any term without altering it, and vice-versa. If products are associative, so are sums; if sums are commutative, so are products, etc.

Any expression that can be formed with OR can also be expressed with AND and negation. Any expression that can be formed with AND can also be expressed with OR and negation. These relationships are stated in two theorems known as DeMorgan's theorems.

- $\overline{A + B} = \bar{A} \cdot \bar{B}$ which also may be stated $A + B = \overline{\bar{A} \cdot \bar{B}}$
- $\overline{A \cdot B} = \bar{A} + \bar{B}$ also stated $A \cdot B = \overline{\bar{A} + \bar{B}}$

The rules for finding the dual or equivalent expression of a function are:

1. Replace OR symbols with AND symbols and replace AND symbols with OR symbols.
2. Complement each of the variables (terms) in the expression.
3. Recomplement the remaining expression.

CANONICAL FORMS

The Boolean expression or truth statement derived from truth tables is called a canonical expansion.

Two methods of developing the canonical form are:

1. Sum of products or disjunctive normal form.

Consider the problem:

Inputs: A, B

Output: C

The relationships between the inputs and outputs are

INPUTS		OUTPUTS
A	B	C
0	0	1
0	1	0
1	0	1
1	1	1

Create a new column in the truth table for the "product terms". The column will contain the product of the input variables with the "letter" representing the respective input variable when its value is 1 and the "complement of the letter" when its value is 0.

INPUT		OUTPUT	PRODUCT
A	B	C	TERMS
0	0	1	$\bar{A}\bar{B}$
0	1	0	$\bar{A}B$
1	0	1	$A\bar{B}$
1	1	1	AB

Multiply the product terms by their respective values of C and sum the products.

$$(\bar{A}\bar{B} \cdot 1) + (\bar{A}B \cdot 0) + (A\bar{B} \cdot 1) + (AB \cdot 1)$$

$$\bar{A}\bar{B} + A\bar{B} + AB = \text{sum of products or canonical form}$$

This is the method most often used.

2. Product of sums or conjunctive normal form. Consider the problem:

Inputs = A, B

Output = C

The relationship between the inputs and output are:

INPUT		OUTPUT
A	B	C
0	0	0
0	1	1
1	0	1
1	1	0

Create a new column in the truth table for the "sum terms". The column will contain the sum of the input variables with the "complement of the letter" representing the respective input variable when its value is 1 and the "letter" representing the input variable when its value is 0.

INPUT		OUTPUT	SUM
A	B	C	TERMS
0	0	0	$A + B$
0	1	1	$A + \bar{B}$
1	0	1	$\bar{A} + B$
1	1	0	$\bar{A} + \bar{B}$

Form the product of sums by adding the sum terms to their respective values of C, and multiplying the sums.

$$(A + B + 0) \cdot (A + \bar{B} + 1) \cdot (\bar{A} + B + 1) \cdot (\bar{A} + \bar{B} + 0)$$

$$(\bar{A} + B) \cdot 1 \cdot 1 \cdot (\bar{A} + \bar{B}) = (A + B) \cdot (\bar{A} + \bar{B})$$

STATISTICAL TABLES

ORDINATES OF THE NORMAL DENSITY FUNCTION

This table gives values of

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$$

for values of x between zero and four at intervals of 0.01. Of course one uses the fact that $f(-x) = f(x)$ for negative values of x.

x	.00	.01	.02	.03	.04	.05	.06	.07	.08	.09
.0	.3989	.3989	.3989	.3988	.3986	.3984	.3982	.3980	.3977	.3973
.1	.3970	.3965	.3961	.3956	.3951	.3945	.3939	.3932	.3925	.3918
.2	.3910	.3902	.3894	.3885	.3876	.3867	.3857	.3847	.3836	.3825
.3	.3814	.3802	.3790	.3778	.3765	.3752	.3739	.3725	.3712	.3697
.4	.3683	.3668	.3653	.3637	.3621	.3605	.3589	.3572	.3555	.3538
.5	.3521	.3503	.3485	.3467	.3448	.3429	.3410	.3391	.3372	.3352
.6	.3332	.3312	.3292	.3271	.3251	.3230	.3209	.3187	.3166	.3144
.7	.3123	.3101	.3079	.3056	.3034	.3011	.2989	.2966	.2943	.2920
.8	.2897	.2874	.2850	.2827	.2803	.2780	.2756	.2732	.2709	.2685
.9	.2661	.2637	.2613	.2589	.2565	.2541	.2516	.2492	.2468	.2444
1.0	.2420	.2396	.2371	.2347	.2323	.2299	.2275	.2251	.2227	.2203
1.1	.2179	.2155	.2131	.2107	.2083	.2059	.2036	.2012	.1989	.1965
1.2	.1942	.1919	.1895	.1872	.1849	.1826	.1804	.1781	.1758	.1736
1.3	.1714	.1691	.1669	.1647	.1626	.1604	.1582	.1561	.1539	.1518
1.4	.1497	.1476	.1456	.1435	.1415	.1394	.1374	.1354	.1334	.1315
1.5	.1295	.1276	.1257	.1238	.1219	.1200	.1182	.1163	.1145	.1127
1.6	.1109	.1092	.1074	.1057	.1040	.1023	.1006	.9899	.9973	.9957
1.7	.9940	.9925	.9909	.9893	.9878	.9863	.9848	.9833	.9818	.9804
1.8	.9790	.9775	.9761	.9748	.9734	.9721	.9707	.9694	.9681	.9669
1.9	.9656	.9644	.9632	.9620	.9608	.9596	.9584	.9573	.9562	.9551
2.0	.9540	.9529	.9519	.9508	.9498	.9488	.9478	.9468	.9459	.9449
2.1	.9440	.9431	.9422	.9413	.9404	.9396	.9387	.9379	.9371	.9363
2.2	.9355	.9347	.9339	.9332	.9325	.9317	.9310	.9303	.9297	.9290
2.3	.9283	.9277	.9270	.9264	.9258	.9252	.9246	.9241	.9235	.9229
2.4	.9224	.9219	.9213	.9208	.9203	.9198	.9194	.9189	.9184	.9180
2.5	.9175	.9171	.9167	.9163	.9158	.9154	.9151	.9147	.9143	.9139
2.6	.9136	.9132	.9129	.9126	.9122	.9119	.9116	.9113	.9110	.9107
2.7	.9104	.9101	.9099	.9096	.9093	.9091	.9088	.9086	.9084	.9081
2.8	.9079	.9077	.9075	.9073	.9071	.9069	.9067	.9065	.9063	.9061
2.9	.9060	.9058	.9056	.9055	.9053	.9051	.9050	.9048	.9047	.9046
3.0	.9044	.9043	.9042	.9040	.9039	.9038	.9037	.9036	.9035	.9034
3.1	.9033	.9032	.9031	.9030	.9029	.9028	.9027	.9026	.9025	.9025
3.2	.9024	.9023	.9022	.9022	.9021	.9020	.9020	.9019	.9018	.9018
3.3	.9017	.9017	.9016	.9016	.9015	.9015	.9014	.9014	.9013	.9013
3.4	.9012	.9012	.9012	.9011	.9011	.9010	.9010	.9010	.9009	.9009
3.5	.9009	.9008	.9008	.9008	.9008	.9007	.9007	.9007	.9007	.9006
3.6	.9006	.9006	.9006	.9005	.9005	.9005	.9005	.9005	.9005	.9004
3.7	.9004	.9004	.9004	.9004	.9004	.9004	.9003	.9003	.9003	.9003
3.8	.9003	.9003	.9003	.9003	.9003	.9002	.9002	.9002	.9002	.9002
3.9	.9002	.9002	.9002	.9002	.9002	.9002	.9002	.9002	.9001	.9001

Reproduced, with permission, from "Introduction to the Theory of Statistics", by Alexander M. Mood, 1963, McGraw-Hill Book Company, Inc., pages 428 and 430.

CUMULATIVE NORMAL DISTRIBUTION

This tabulates

$$F(x) = \int_{-\infty}^x \frac{1}{\sqrt{2\pi}} e^{-t^2/2} dt$$

for values of x between zero and 3.5 at intervals of

0.01. For negative values of x, one uses the relation $F(-x) = 1 - F(x)$. Values of x corresponding to a few round values of F are given separately beneath the main table.

x	.00	.01	.02	.03	.04	.05	.06	.07	.08	.09
.0	.5000	.5040	.5080	.5120	.5160	.5199	.5239	.5279	.5319	.5359
.1	.5398	.5438	.5478	.5517	.5557	.5596	.5636	.5675	.5714	.5753
.2	.5793	.5832	.5871	.5910	.5948	.5987	.6026	.6064	.6103	.6141
.3	.6179	.6217	.6255	.6293	.6331	.6368	.6406	.6443	.6480	.6517
.4	.6554	.6591	.6628	.6664	.6700	.6736	.6772	.6808	.6844	.6879
.5	.6915	.6950	.6985	.7019	.7054	.7088	.7123	.7157	.7190	.7224
.6	.7257	.7291	.7324	.7357	.7389	.7422	.7454	.7486	.7517	.7549
.7	.7580	.7611	.7642	.7673	.7704	.7734	.7764	.7794	.7823	.7852
.8	.7881	.7910	.7939	.7967	.7995	.8023	.8051	.8078	.8106	.8133
.9	.8159	.8186	.8212	.8238	.8264	.8289	.8315	.8340	.8365	.8389
1.0	.8413	.8438	.8461	.8485	.8508	.8531	.8554	.8577	.8599	.8621
1.1	.8643	.8665	.8686	.8708	.8729	.8749	.8770	.8790	.8810	.8830
1.2	.8849	.8869	.8888	.8907	.8925	.8944	.8962	.8980	.8997	.9015
1.3	.9032	.9049	.9066	.9082	.9099	.9115	.9131	.9147	.9162	.9177
1.4	.9192	.9207	.9222	.9236	.9251	.9265	.9279	.9292	.9306	.9319
1.5	.9332	.9345	.9357	.9370	.9382	.9394	.9406	.9418	.9429	.9441
1.6	.9452	.9463	.9474	.9484	.9495	.9505	.9515	.9525	.9535	.9545
1.7	.9554	.9564	.9573	.9582	.9591	.9599	.9608	.9616	.9625	.9633
1.8	.9641	.9649	.9656	.9664	.9671	.9678	.9686	.9693	.9699	.9706
1.9	.9713	.9719	.9726	.9732	.9738	.9744	.9750	.9756	.9761	.9767
2.0	.9772	.9778	.9783	.9788	.9793	.9798	.9803	.9808	.9812	.9817
2.1	.9821	.9826	.9830	.9834	.9838	.9842	.9846	.9850	.9854	.9857
2.2	.9861	.9864	.9868	.9871	.9875	.9878	.9881	.9884	.9887	.9890
2.3	.9893	.9896	.9898	.9901	.9904	.9906	.9909	.9911	.9913	.9916
2.4	.9918	.9920	.9922	.9925	.9927	.9929	.9931	.9932	.9934	.9936
2.5	.9938	.9940	.9941	.9943	.9945	.9946	.9948	.9949	.9951	.9952
2.6	.9953	.9955	.9956	.9957	.9959	.9960	.9961	.9962	.9963	.9964
2.7	.9965	.9966	.9967	.9968	.9969	.9970	.9971	.9972	.9973	.9974
2.8	.9974	.9975	.9976	.9977	.9977	.9978	.9979	.9979	.9980	.9981
2.9	.9981	.9982	.9982	.9983	.9984	.9984	.9985	.9985	.9986	.9986
3.0	.9987	.9987	.9987	.9988	.9988	.9989	.9989	.9989	.9990	.9990
3.1	.9990	.9991	.9991	.9991	.9992	.9992	.9992	.9992	.9993	.9993
3.2	.9993	.9993	.9994	.9994	.9994	.9994	.9994	.9994	.9995	.9995
3.3	.9995	.9995	.9995	.9996	.9996	.9996	.9996	.9996	.9996	.9997
3.4	.9997	.9997	.9997	.9997	.9997	.9997	.9997	.9997	.9997	.9998

x	1.282	1.645	1.960	2.326	2.576	3.090	3.291	3.891	4.417
F(x)	.90	.95	.975	.99	.995	.999	.9995	.99995	.999995
2[1 - F(x)]	.20	.10	.05	.02	.01	.002	.001	.0001	.00001

Reproduced, with permission, from "Introduction to the Theory of Statistics", by Alexander M. Mood, 1963, McGraw-Hill Book Company, Inc., pages 428 and 431.

CUMULATIVE CHI-SQUARE DISTRIBUTION

This table gives values of u corresponding to a few selected values of F(u) where

$$F(u) = \int_0^u \frac{x^{(n-2)/2} e^{-x/2} dx}{2^{n/2} \Gamma(n/2)}$$

for n, the number of degrees of freedom, equal to 1, 2, . . . , 30. For larger values of n, a normal approximation is quite accurate. The quantity $\sqrt{2u} - \sqrt{2n - 1}$ is nearly normally distributed with

$n \backslash F$.005	.010	.025	.050	.100	.250	.500	.750	.900	.950	.975	.990	.995
1	.0393	.0157	.00982	.00393	.0158	.102	.455	1.32	2.71	3.84	5.02	6.63	7.88
2	.0100	.0201	.0506	.103	.211	.575	1.39	2.77	4.61	5.99	7.38	9.21	10.6
3	.0717	.115	.216	.352	.584	1.21	2.37	4.11	6.25	7.81	9.35	11.3	12.8
4	.207	.297	.484	.711	1.06	1.92	3.36	5.39	7.78	9.49	11.1	13.3	14.9
5	.412	.554	.831	1.15	1.61	2.67	4.35	6.63	9.24	11.1	12.8	15.1	16.7
6	.676	.872	1.24	1.64	2.20	3.45	5.35	7.84	10.6	12.6	14.4	16.8	18.5
7	.989	1.24	1.69	2.17	2.83	4.25	6.35	9.04	12.0	14.1	16.0	18.5	20.3
8	1.34	1.63	2.13	2.73	3.49	5.07	7.34	10.2	13.4	15.5	17.5	20.1	22.0
9	1.73	2.09	2.70	3.33	4.17	5.90	8.34	11.4	14.7	16.9	19.0	21.7	23.6
10	2.16	2.56	3.25	3.94	4.87	6.74	9.34	12.5	16.0	18.3	20.5	23.2	25.2
11	2.60	3.05	3.82	4.57	5.58	7.58	10.3	13.7	17.3	19.7	21.9	24.7	26.8
12	3.07	3.57	4.40	5.23	6.30	8.44	11.3	14.8	18.5	21.0	23.3	26.2	28.3
13	3.57	4.11	5.01	5.89	7.04	9.30	12.3	16.0	19.8	22.4	24.7	27.7	29.8
14	4.07	4.66	5.63	6.57	7.79	10.2	13.3	17.1	21.1	23.7	26.1	29.1	31.3
15	4.60	5.23	6.26	7.26	8.55	11.0	14.3	18.2	22.3	25.0	27.5	30.6	32.8
16	5.14	5.81	6.91	7.96	9.31	11.9	15.3	19.4	23.5	26.3	28.8	32.0	34.3
17	5.70	6.41	7.56	8.67	10.1	12.8	16.3	20.5	24.8	27.6	30.2	33.4	35.7
18	6.26	7.01	8.23	9.39	10.9	13.7	17.3	21.6	26.0	28.9	31.5	34.8	37.2
19	6.84	7.63	8.91	10.1	11.7	14.6	18.3	22.7	27.2	30.1	32.9	36.2	38.6
20	7.43	8.26	9.59	10.9	12.4	15.5	19.3	23.8	28.4	31.4	34.2	37.6	40.0
21	8.03	8.90	10.3	11.6	13.2	16.3	20.3	24.9	29.6	32.7	35.5	38.9	41.4
22	8.64	9.54	11.0	12.3	14.0	17.2	21.3	26.0	30.8	33.9	36.8	40.3	42.8
23	9.26	10.2	11.7	13.1	14.8	18.1	22.3	27.1	32.0	35.2	38.1	41.6	44.2
24	9.89	10.9	12.4	13.8	15.7	19.0	23.3	28.2	33.2	36.4	39.4	43.0	45.6
25	10.5	11.5	13.1	14.6	16.5	19.9	24.3	29.3	34.4	37.7	40.6	44.3	46.9
26	11.2	12.2	13.8	15.4	17.3	20.8	25.3	30.4	35.6	38.9	41.9	45.6	48.3
27	11.8	12.9	14.6	16.2	18.1	21.7	26.3	31.5	36.7	40.1	43.2	47.0	49.6
28	12.5	13.6	15.3	16.9	18.9	22.7	27.3	32.6	37.9	41.3	44.5	48.3	51.0
29	13.1	14.3	16.0	17.7	19.8	23.6	28.3	33.7	39.1	42.6	45.7	49.6	52.3
30	13.8	15.0	16.8	18.5	20.6	24.5	29.3	34.8	40.3	43.8	47.0	50.9	53.7

Reproduced, with permission, from "Introduction to the Theory of Statistics", by Alexander M. Mood, 1963, McGraw-Hill Book Company, Inc., pages 428 and 432, as abridged, and with permission, from "Tables of percentage points of the incomplete beta function and of the chi-square distribution", *Biometrika*, vol. 32, 1941, by Catherine M. Thompson.

zero mean and unit variance. Thus u_a , the a point of the distribution, may be computed by

$$u_a = 1/2 (x_a + \sqrt{2n - 1})^2$$

where x_a is the a point of the cumulative normal distribution. As an illustration, we may compute the .95 value of u for n = 30 degrees of freedom:

$$u_{.95} = 1/2 (1.645 + \sqrt{59})^2 = 43.5$$

which is in error by less than 1 percent.

CUMULATIVE "STUDENT'S" DISTRIBUTION

This table gives values of t corresponding to a few selected values of

$$F(t) = \int_{-\infty}^t \frac{\left(\frac{n-1}{2}\right)!}{\left(\frac{n-2}{2}\right)! \sqrt{\pi n} \left(1 + \frac{x^2}{n}\right)^{(n+1)/2}} dx$$

with n = 1, 2, . . . , 30, 40, 60, 120, ∞. Since the density is symmetric in t, it follows that F(-t) = 1 - F(t). One should not interpolate linearly between degrees of freedom but on the reciprocal of the degrees of freedom, if good accuracy in the last digit is desired. As an illustration, we shall compute the .975 value for 40 degrees of freedom. The values for 30 and 60 are 2.042 and 2.000. Using the reciprocals of n, the interpolated value is

$$2.042 - \frac{1/30 - 1/40}{1/30 - 1/60} (2.042 - 2.000) = 2.021$$

which is the correct value. Interpolating linearly, one would have obtained 2.028

F n	.75	.90	.95	.975	.99	.995	.9995
1	1.000	3.078	6.314	12.706	31.821	63.657	636.619
2	.816	1.886	2.920	4.303	6.965	9.925	31.598
3	.765	1.638	2.353	3.182	4.541	5.841	12.941
4	.741	1.533	2.132	2.776	3.747	4.604	8.610
5	.727	1.476	2.015	2.571	3.365	4.032	6.859
6	.718	1.440	1.943	2.447	3.143	3.707	5.959
7	.711	1.415	1.895	2.365	2.998	3.499	5.405
8	.706	1.397	1.860	2.306	2.896	3.355	5.041
9	.703	1.383	1.833	2.262	2.821	3.250	4.781
10	.700	1.372	1.812	2.228	2.764	3.169	4.587
11	.697	1.363	1.796	2.201	2.718	3.106	4.437
12	.695	1.356	1.782	2.179	2.681	3.055	4.318
13	.694	1.350	1.771	2.160	2.650	3.012	4.221
14	.692	1.345	1.761	2.145	2.624	2.977	4.140
15	.691	1.341	1.753	2.131	2.602	2.947	4.073
16	.690	1.337	1.746	2.120	2.583	2.921	4.015
17	.689	1.333	1.740	2.110	2.567	2.898	3.965
18	.688	1.330	1.734	2.101	2.552	2.878	3.922
19	.688	1.328	1.729	2.093	2.539	2.861	3.883
20	.687	1.325	1.725	2.086	2.528	2.845	3.850
21	.686	1.323	1.721	2.080	2.518	2.831	3.819
22	.686	1.321	1.717	2.074	2.508	2.819	3.792
23	.685	1.319	1.714	2.069	2.500	2.807	3.767
24	.685	1.318	1.711	2.064	2.492	2.797	3.745
25	.684	1.316	1.708	2.060	2.485	2.787	3.725
26	.684	1.315	1.706	2.056	2.479	2.779	3.707
27	.684	1.314	1.703	2.052	2.473	2.771	3.690
28	.683	1.313	1.701	2.048	2.467	2.763	3.674
29	.683	1.311	1.699	2.045	2.462	2.756	3.659
30	.683	1.310	1.697	2.042	2.457	2.750	3.646
40	.681	1.303	1.684	2.021	2.423	2.704	3.551
60	.679	1.296	1.671	2.000	2.390	2.660	3.460
120	.677	1.289	1.658	1.980	2.358	2.617	3.373
∞	.674	1.282	1.645	1.960	2.326	2.576	3.291

Reproduced, with permission, from "Introduction to the Theory of Statistics", by Alexander M. Mood, 1963, McGraw-Hill Book Company, Inc., pages 429 and 433, as abridged, and with permission, from "Statistical Tables for Biological Agricultural, and Medical Research", by R. A. Fisher and Frank Yates, Oliver and Boyd, Ltd., Edinburgh.

CUMULATIVE F DISTRIBUTION

This table gives values of F corresponding to five values of

$$G(F) = \int_0^F \frac{\Gamma\left(\frac{m+n-2}{2}\right)! m^{m/2} n^{n/2} x^{m/2} (n+mx)^{-(m+n)/2} dx}{\left(\frac{m-2}{2}\right)! \left(\frac{n-2}{2}\right)!}$$

for selected values of m and n; m is the number of degrees of freedom in the numerator of F, and n is the number of degrees of freedom in the denominator of F. The table also provides values corresponding to G = .10, .05, .025, .01, and .005 because $F_{1-\alpha}$ for m and n degrees of freedom is the reciprocal of F_{α} for n and m degrees of freedom. Thus for G = .05 with 3 and 6 degrees, one finds

$$F_{.05}(3, 6) = \frac{1}{F_{.95}(6, 3)} = \frac{1}{8.94} = .112$$

CUMULATIVE F DISTRIBUTION

G	n	m	1	2	3	4	5	6	7	8	9	10	12	15	20	30	60	120	∞
.90	1		39.9	49.5	53.6	55.8	57.2	58.2	58.9	59.4	59.9	60.2	60.7	61.2	61.7	62.3	62.8	63.1	63.3
.95			161	200	216	225	230	234	237	239	241	242	244	246	248	250	252	253	254
.975			648	800	864	900	922	937	948	957	963	969	977	985	993	1000	1010	1010	1020
.99			4,050	5,000	5,400	5,620	5,760	5,860	5,930	5,980	6,020	6,060	6,110	6,160	6,210	6,260	6,310	6,340	6,370
.995			16,200	20,000	21,600	22,500	23,100	23,400	23,700	23,900	24,100	24,200	24,400	24,600	24,800	25,000	25,200	25,400	25,500
.90	2		8.53	9.00	9.16	9.24	9.29	9.33	9.35	9.37	9.38	9.39	9.41	9.42	9.44	9.46	9.47	9.48	9.49
.95			18.5	19.0	19.2	19.3	19.3	19.4	19.4	19.4	19.4	19.4	19.4	19.4	19.5	19.5	19.5	19.5	19.5
.975			38.5	39.0	39.2	39.2	39.3	39.3	39.4	39.4	39.4	39.4	39.4	39.4	39.4	39.5	39.5	39.5	39.5
.99			98.5	99.0	99.2	99.2	99.3	99.3	99.4	99.4	99.4	99.4	99.4	99.4	99.4	99.5	99.5	99.5	99.5
.995			199	199	199	199	199	199	199	199	199	199	199	199	199	199	199	199	199
.90	3		5.34	5.46	5.39	5.34	5.31	5.28	5.27	5.25	5.24	5.23	5.22	5.20	5.18	5.17	5.15	5.14	5.13
.95			10.1	9.55	9.28	9.12	9.01	8.94	8.89	8.85	8.81	8.79	8.74	8.70	8.66	8.62	8.57	8.55	8.53
.975			12.4	16.0	15.4	15.1	14.9	14.7	14.6	14.5	14.5	14.4	14.3	14.2	14.2	14.1	14.0	13.9	13.9
.99			34.1	30.8	29.5	28.7	28.2	27.9	27.7	27.5	27.3	27.2	27.1	26.9	26.7	26.5	26.3	26.2	26.1
.995			55.6	49.8	47.5	46.2	45.4	44.8	44.4	44.1	43.9	43.7	43.4	43.1	42.8	42.5	42.1	42.0	41.8
.90	4		4.54	4.32	4.19	4.11	4.05	4.01	3.98	3.95	3.93	3.92	3.90	3.87	3.84	3.82	3.79	3.78	3.78
.95			7.71	6.94	6.59	6.39	6.26	6.16	6.09	6.04	6.00	5.96	5.91	5.86	5.80	5.75	5.69	5.66	5.63
.975			12.2	10.6	9.98	9.60	9.36	9.20	9.07	8.98	8.90	8.84	8.75	8.66	8.56	8.46	8.36	8.31	8.26
.99			21.2	18.0	16.7	16.0	15.5	15.2	15.0	14.8	14.7	14.5	14.4	14.2	14.0	13.8	13.7	13.6	13.5
.995			31.3	26.3	24.3	23.2	22.5	22.0	21.6	21.4	21.1	21.0	20.7	20.4	20.2	19.9	19.6	19.5	19.3
.90	5		4.06	3.78	3.62	3.52	3.45	3.40	3.37	3.34	3.32	3.30	3.27	3.24	3.21	3.17	3.14	3.12	3.11
.95			6.61	5.79	5.41	5.19	5.05	4.95	4.88	4.82	4.77	4.74	4.68	4.62	4.56	4.50	4.43	4.40	4.37
.975			10.0	8.43	7.76	7.39	7.15	6.98	6.85	6.76	6.68	6.62	6.52	6.43	6.33	6.23	6.12	6.07	6.02
.99			16.2	13.3	12.1	11.4	11.0	10.7	10.5	10.3	10.2	10.1	9.89	9.72	9.55	9.38	9.20	9.11	9.02
.995			22.8	18.3	16.5	15.6	14.9	14.5	14.2	14.0	13.8	13.6	13.4	13.1	12.9	12.7	12.4	12.3	12.1
.90	6		3.78	3.46	3.29	3.18	3.11	3.05	3.01	2.98	2.96	2.94	2.90	2.87	2.84	2.80	2.76	2.74	2.72
.95			5.99	5.14	4.76	4.53	4.39	4.28	4.21	4.15	4.10	4.06	4.00	3.94	3.87	3.81	3.74	3.70	3.67
.975			8.81	7.26	6.60	6.23	5.99	5.82	5.70	5.60	5.52	5.46	5.37	5.27	5.17	5.07	4.96	4.90	4.85
.99			13.7	10.9	9.78	9.15	8.75	8.47	8.26	8.10	7.98	7.87	7.72	7.56	7.40	7.23	7.06	6.97	6.88
.995			18.6	14.5	12.9	12.0	11.5	11.1	10.8	10.6	10.4	10.2	10.0	9.81	9.59	9.36	9.12	9.00	8.88
.90	7		3.59	3.26	3.07	2.96	2.88	2.83	2.78	2.75	2.72	2.70	2.67	2.63	2.59	2.56	2.51	2.49	2.47
.95			5.59	4.74	4.35	4.12	3.97	3.87	3.79	3.73	3.68	3.64	3.57	3.51	3.44	3.38	3.30	3.27	3.23
.975			8.07	6.54	5.89	5.52	5.29	5.12	4.99	4.90	4.82	4.76	4.67	4.57	4.47	4.36	4.25	4.20	4.14
.99			12.2	9.55	8.45	7.85	7.46	7.19	6.99	6.84	6.72	6.62	6.47	6.31	6.16	5.99	5.82	5.74	5.65
.995			16.2	12.4	10.9	10.1	9.52	9.16	8.89	8.68	8.51	8.38	8.18	7.97	7.75	7.53	7.31	7.19	7.08
.90	8		3.46	3.11	2.92	2.81	2.73	2.67	2.62	2.59	2.56	2.54	2.50	2.46	2.42	2.38	2.34	2.31	2.29
.95			5.32	4.46	4.07	3.84	3.69	3.58	3.50	3.44	3.39	3.35	3.28	3.22	3.15	3.08	3.01	2.97	2.93
.975			7.57	6.06	5.42	5.05	4.82	4.65	4.53	4.43	4.36	4.30	4.20	4.10	4.00	3.89	3.78	3.73	3.67
.99			11.3	8.65	7.59	7.01	6.63	6.37	6.18	6.03	5.91	5.81	5.67	5.52	5.36	5.20	5.03	4.95	4.86
.995			14.7	11.0	9.60	8.81	8.30	7.95	7.69	7.50	7.34	7.21	7.01	6.81	6.61	6.40	6.18	6.06	5.95
.90	9		3.36	3.01	2.81	2.69	2.61	2.55	2.51	2.47	2.44	2.42	2.38	2.34	2.30	2.25	2.21	2.18	2.16
.95			5.12	4.26	3.86	3.63	3.48	3.37	3.29	3.23	3.18	3.14	3.07	3.01	2.94	2.86	2.79	2.75	2.71
.975			7.21	5.71	5.08	4.72	4.48	4.32	4.20	4.10	4.03	3.96	3.87	3.77	3.67	3.56	3.45	3.39	3.33
.99			10.6	8.08	6.99	6.42	6.06	5.80	5.61	5.47	5.35	5.26	5.11	4.96	4.81	4.65	4.48	4.40	4.31
.995			13.6	10.3	8.72	7.96	7.47	7.13	6.85	6.69	6.54	6.42	6.23	6.03	5.83	5.62	5.41	5.30	5.19
.90	10		3.29	2.92	2.73	2.61	2.52	2.46	2.41	2.38	2.35	2.32	2.28	2.24	2.20	2.15	2.11	2.08	2.06
.95			4.96	4.10	3.69	3.48	3.33	3.22	3.14	3.07	3.02	2.98	2.91	2.84	2.77	2.70	2.62	2.58	2.54
.975			6.94	5.46	4.83	4.47	4.24	4.07	3.95	3.85	3.78	3.72	3.62	3.52	3.42	3.31	3.20	3.14	3.08
.99			10.0	7.56	6.55	5.99	5.64	5.39	5.20	5.06	4.94	4.85	4.71	4.56	4.41	4.25	4.08	4.00	3.91
.995			12.8	9.43	8.08	7.34	6.87	6.54	6.30	6.12	5.97	5.85	5.66	5.47	5.27	5.07	4.86	4.75	4.64
.90	12		3.18	2.81	2.61	2.48	2.39	2.33	2.28	2.24	2.21	2.19	2.15	2.10	2.06	2.01	1.96	1.93	1.90
.95			4.75	3.89	3.49	3.26	3.11	3.00	2.91	2.85	2.80	2.75	2.69	2.62	2.54	2.47	2.38	2.34	2.30
.975			6.55	5.10	4.47	4.12	3.89	3.73	3.61	3.51	3.44	3.37	3.28	3.18	3.07	2.96	2.85	2.79	2.72
.99			9.33	6.93	5.95	5.41	5.06	4.82	4.64	4.50	4.39	4.30	4.16	4.01	3.86	3.70	3.54	3.45	3.36
.995			11.8	8.51	7.23	6.52	6.07	5.76	5.52	5.35	5.20	5.09	4.91	4.72	4.53	4.33	4.12	4.01	3.90
.90	15		3.07	2.70	2.49	2.36	2.27	2.21	2.16	2.12	2.09	2.06	2.02	1.97	1.92	1.87	1.82	1.79	1.76
.95			4.54	3.68	3.29	3.06	2.90	2.79	2.71	2.64	2.59	2.54	2.48	2.40	2.32	2.25	2.16	2.11	2.07
.975			6.20	4.77	4.15	3.80	3.58	3.41	3.29	3.20	3.12	3.06	2.96	2.86	2.76	2.64	2.52	2.46	2.40
.99			8.68	6.36	5.42	4.89	4.56	4.32	4.14	4.00	3.89	3.80	3.67	3.52	3.37	3.21	3.05	2.96	2.87
.995			10.8	7.70	6.48	5.80	5.37	5.07	4.85	4.67	4.54	4.42	4.25	4.07	3.88	3.69	3.48	3.37	3.26
.90	20		2.97	2.59	2.38	2.25	2.16	2.09	2.04	2.00	1.96	1.94	1.89	1.84	1.79	1.74	1.68	1.64	1.61
.95			4.35	3.49	3.10	2.87	2.71	2.60	2.51	2.45	2.39	2.35	2.28	2.20	2.12	2.04	1.95	1.90	1.84
.975			5.87	4.46	3.86	3.51	3.29	3.13	3.01	2.91	2.84	2.77							

SUMMED POISSON DISTRIBUTION FUNCTION

This table contains values of

$$F(x^1) = \sum_{x=x^1}^{x=\infty} \frac{e^{-m} m^x}{(x)!}$$

for specified values x^1 and m . For example, if $m=0.6$ and $x^1=5$ then $F(x^1)=.0004$.

x'	m									
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
1	.0952	.1813	.2592	.3297	.3935	.4512	.5034	.5507	.5934	.6321
2	.0047	.0175	.0369	.0616	.0902	.1219	.1558	.1912	.2275	.2642
3	.0002	.0011	.0036	.0079	.0144	.0231	.0341	.0474	.0629	.0803
4	.0000	.0001	.0003	.0008	.0018	.0034	.0058	.0091	.0135	.0190
5	.0000	.0000	.0000	.0001	.0002	.0004	.0008	.0014	.0023	.0037
6	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0002	.0003	.0006
7	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001

x'	m									
	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2.0
0	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
1	.6671	.6988	.7275	.7543	.7769	.7981	.8173	.8347	.8504	.8647
2	.3010	.3374	.3732	.4082	.4422	.4751	.5068	.5372	.5663	.5940
3	.0996	.1205	.1429	.1665	.1912	.2166	.2428	.2694	.2963	.3233
4	.0257	.0338	.0431	.0537	.0656	.0788	.0932	.1087	.1253	.1429
5	.0054	.0077	.0107	.0143	.0186	.0237	.0296	.0364	.0441	.0527
6	.0010	.0015	.0022	.0032	.0045	.0060	.0080	.0104	.0132	.0166
7	.0001	.0003	.0004	.0006	.0009	.0013	.0019	.0026	.0034	.0045
8	.0000	.0000	.0001	.0001	.0002	.0003	.0004	.0006	.0008	.0011
9	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0001	.0002	.0002

x'	m									
	2.1	2.2	2.3	2.4	2.5	2.6	2.7	2.8	2.9	3.0
0	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
1	.8775	.8892	.8997	.9093	.9179	.9257	.9328	.9392	.9450	.9502
2	.6204	.6454	.6691	.6916	.7127	.7326	.7513	.7689	.7854	.8009
3	.3504	.3773	.4040	.4303	.4562	.4816	.5064	.5305	.5540	.5768
4	.1614	.1806	.2007	.2213	.2424	.2640	.2859	.3081	.3304	.3528
5	.0621	.0725	.0838	.0959	.1088	.1226	.1371	.1523	.1682	.1847
6	.0204	.0249	.0300	.0357	.0420	.0490	.0567	.0651	.0742	.0839
7	.0059	.0075	.0094	.0116	.0142	.0172	.0206	.0244	.0287	.0335
8	.0015	.0020	.0026	.0033	.0042	.0053	.0066	.0081	.0099	.0119
9	.0003	.0005	.0006	.0009	.0011	.0015	.0019	.0024	.0031	.0038
10	.0001	.0001	.0001	.0002	.0003	.0004	.0005	.0007	.0009	.0011
11	.0000	.0000	.0000	.0000	.0001	.0001	.0001	.0002	.0002	.0003
12	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0001

x'	m									
	3.1	3.2	3.3	3.4	3.5	3.6	3.7	3.8	3.9	4.0
0	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
1	.9550	.9592	.9631	.9666	.9698	.9727	.9753	.9776	.9798	.9817
2	.8153	.8288	.8414	.8532	.8641	.8743	.8838	.8926	.9008	.9084
3	.5988	.6201	.6406	.6603	.6792	.6973	.7146	.7311	.7469	.7619
4	.3752	.3975	.4197	.4416	.4634	.4848	.5058	.5265	.5468	.5665
5	.2018	.2194	.2374	.2558	.2746	.2936	.3128	.3322	.3516	.3712
6	.0943	.1054	.1171	.1295	.1424	.1559	.1699	.1844	.1994	.2149
7	.0388	.0446	.0510	.0579	.0653	.0733	.0818	.0909	.1005	.1107
8	.0142	.0168	.0198	.0231	.0267	.0308	.0352	.0401	.0454	.0511
9	.0047	.0057	.0069	.0083	.0099	.0117	.0137	.0160	.0185	.0214
10	.0014	.0018	.0022	.0027	.0033	.0040	.0048	.0058	.0069	.0081
11	.0004	.0005	.0006	.0008	.0010	.0013	.0016	.0019	.0023	.0028
12	.0001	.0001	.0002	.0002	.0003	.0004	.0005	.0006	.0007	.0009
13	.0000	.0000	.0000	.0001	.0001	.0001	.0001	.0002	.0002	.0003
14	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0001

Reproduced, with permission, from "Handbook of Mathematical Tables", Chemical Rubber Publishing Company, pages 449-452.

SUMMED POISSON DISTRIBUTION FUNCTION $\sum_{x=x'}^{x=\infty} \frac{e^{-m} m^x}{(x)!}$

		<i>m</i>									
<i>x'</i>	4.1	4.2	4.3	4.4	4.5	4.6	4.7	4.8	4.9	5.0	
0	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	
1	.9834	.9850	.9864	.9877	.9889	.9899	.9909	.9918	.9926	.9933	
2	.9155	.9220	.9281	.9337	.9389	.9437	.9482	.9523	.9561	.9596	
3	.7762	.7898	.8026	.8149	.8264	.8374	.8477	.8575	.8667	.8753	
4	.5858	.6046	.6228	.6406	.6577	.6743	.6903	.7058	.7207	.7350	
5	.3907	.4102	.4296	.4488	.4679	.4868	.5054	.5237	.5418	.5595	
6	.2607	.2469	.2633	.2801	.2971	.3142	.3316	.3490	.3665	.3840	
7	.1214	.1325	.1442	.1564	.1689	.1820	.1954	.2092	.2233	.2378	
8	.0573	.0639	.0710	.0786	.0866	.0951	.1040	.1133	.1231	.1334	
9	.0245	.0279	.0317	.0358	.0403	.0451	.0503	.0558	.0618	.0681	
10	.0095	.0111	.0129	.0149	.0171	.0195	.0222	.0251	.0283	.0318	
11	.0034	.0041	.0048	.0057	.0067	.0078	.0090	.0104	.0120	.0137	
12	.0011	.0014	.0017	.0020	.0024	.0029	.0034	.0040	.0047	.0055	
13	.0003	.0004	.0005	.0007	.0008	.0010	.0012	.0014	.0017	.0020	
14	.0001	.0001	.0002	.0002	.0003	.0003	.0004	.0005	.0006	.0007	
15	.0000	.0000	.0000	.0001	.0001	.0001	.0001	.0001	.0002	.0002	
16	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0001	
		<i>m</i>									
<i>x'</i>	5.1	5.2	5.3	5.4	5.5	5.6	5.7	5.8	5.9	6.0	
0	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	
1	.9939	.9945	.9950	.9955	.9959	.9963	.9967	.9970	.9973	.9975	
2	.9628	.9658	.9686	.9711	.9734	.9756	.9776	.9794	.9811	.9826	
3	.8835	.8912	.8984	.9052	.9116	.9176	.9232	.9285	.9334	.9380	
4	.7487	.7619	.7746	.7867	.7983	.8094	.8200	.8300	.8396	.8488	
5	.5769	.5939	.6105	.6267	.6425	.6579	.6728	.6873	.7013	.7149	
6	.4016	.4191	.4365	.4539	.4711	.4881	.5050	.5217	.5381	.5543	
7	.2526	.2676	.2829	.2983	.3140	.3297	.3456	.3616	.3776	.3937	
8	.1440	.1551	.1665	.1783	.1905	.2030	.2159	.2290	.2424	.2560	
9	.0748	.0819	.0894	.0974	.1056	.1143	.1234	.1328	.1426	.1528	
10	.0356	.0397	.0441	.0488	.0538	.0591	.0648	.0708	.0772	.0839	
11	.0156	.0177	.0200	.0225	.0253	.0282	.0314	.0349	.0386	.0426	
12	.0063	.0073	.0084	.0096	.0110	.0125	.0141	.0160	.0179	.0201	
13	.0024	.0028	.0033	.0038	.0045	.0051	.0059	.0068	.0078	.0088	
14	.0008	.0010	.0012	.0014	.0017	.0020	.0023	.0027	.0031	.0036	
15	.0003	.0003	.0004	.0005	.0006	.0007	.0009	.0010	.0012	.0014	
16	.0001	.0001	.0001	.0002	.0002	.0002	.0003	.0004	.0004	.0005	
17	.0000	.0000	.0000	.0001	.0001	.0001	.0001	.0001	.0001	.0002	
18	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001	
		<i>m</i>									
<i>x'</i>	6.1	6.2	6.3	6.4	6.5	6.6	6.7	6.8	6.9	7.0	
0	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	
1	.9978	.9980	.9982	.9983	.9985	.9986	.9988	.9989	.9990	.9991	
2	.9841	.9854	.9866	.9877	.9887	.9897	.9905	.9913	.9920	.9927	
3	.9423	.9464	.9502	.9537	.9570	.9600	.9629	.9656	.9680	.9704	
4	.8575	.8658	.8736	.8811	.8882	.8948	.9012	.9072	.9129	.9182	
5	.7281	.7408	.7531	.7649	.7763	.7873	.7978	.8080	.8177	.8270	
6	.5702	.5859	.6012	.6163	.6310	.6453	.6594	.6730	.6863	.6993	
7	.4098	.4258	.4418	.4577	.4735	.4892	.5047	.5201	.5353	.5503	
8	.2699	.2840	.2983	.3127	.3272	.3419	.3567	.3715	.3864	.4013	
9	.1633	.1741	.1852	.1967	.2084	.2204	.2327	.2452	.2580	.2709	
10	.0910	.0984	.1061	.1142	.1226	.1314	.1404	.1498	.1505	.1695	
11	.0469	.0514	.0563	.0614	.0668	.0726	.0786	.0849	.0916	.0985	
12	.0224	.0250	.0277	.0307	.0339	.0373	.0409	.0448	.0490	.0534	
13	.0100	.0113	.0127	.0143	.0160	.0179	.0199	.0221	.0245	.0270	
14	.0042	.0048	.0055	.0063	.0071	.0080	.0091	.0102	.0115	.0128	
15	.0016	.0019	.0022	.0026	.0030	.0034	.0039	.0044	.0050	.0057	
16	.0006	.0007	.0008	.0010	.0012	.0014	.0016	.0018	.0021	.0024	
17	.0002	.0003	.0003	.0004	.0004	.0005	.0006	.0007	.0008	.0010	
18	.0001	.0001	.0001	.0001	.0002	.0002	.0002	.0003	.0003	.0004	
19	.0000	.0000	.0000	.0000	.0001	.0001	.0001	.0001	.0001	.0001	

SUMMED POISSON DISTRIBUTION FUNCTION $\sum_{x=x'}^{x=\infty} \frac{e^{-m} m^x}{(x)!}$

x'	7.1	7.2	7.3	7.4	7.5 ^m	7.6	7.7	7.8	7.9	8.0
0	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
1	.9992	.9993	.9993	.9994	.9994	.9995	.9995	.9996	.9996	.9997
2	.9933	.9939	.9944	.9949	.9953	.9957	.9961	.9964	.9967	.9970
3	.9725	.9745	.9764	.9781	.9797	.9812	.9826	.9839	.9851	.9862
4	.9233	.9281	.9326	.9368	.9409	.9446	.9482	.9515	.9547	.9576
5	.8359	.8445	.8527	.8605	.8679	.8751	.8819	.8883	.8945	.9004
6	.7119	.7241	.7360	.7474	.7586	.7693	.7797	.7897	.7994	.8088
7	.5651	.5796	.5940	.6080	.6218	.6354	.6486	.6616	.6743	.6866
8	.4162	.4311	.4459	.4607	.4754	.4900	.5044	.5188	.5330	.5470
9	.2840	.2973	.3108	.3243	.3380	.3518	.3657	.3796	.3935	.4075
10	.1798	.1904	.2012	.2123	.2236	.2351	.2469	.2589	.2710	.2834
11	.1058	.1133	.1212	.1293	.1378	.1465	.1555	.1648	.1743	.1841
12	.0580	.0629	.0681	.0735	.0792	.0852	.0915	.0980	.1048	.1119
13	.0297	.0327	.0358	.0391	.0427	.0464	.0504	.0546	.0591	.0638
14	.0143	.0159	.0176	.0195	.0216	.0238	.0261	.0286	.0313	.0342
15	.0065	.0073	.0082	.0092	.0103	.0114	.0127	.0141	.0156	.0173
16	.0028	.0031	.0036	.0041	.0046	.0052	.0059	.0066	.0074	.0082
17	.0011	.0013	.0015	.0017	.0020	.0022	.0026	.0029	.0033	.0037
18	.0004	.0005	.0006	.0007	.0008	.0009	.0011	.0012	.0014	.0016
19	.0002	.0002	.0002	.0003	.0003	.0004	.0004	.0005	.0006	.0006
20	.0001	.0001	.0001	.0001	.0001	.0001	.0002	.0002	.0002	.0003
21	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0001	.0001	.0001
x'	8.1	8.2	8.3	8.4	8.5 ^m	8.6	8.7	8.8	8.9	9.0
0	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
1	.9997	.9997	.9998	.9998	.9998	.9998	.9998	.9998	.9999	.9999
2	.9972	.9975	.9977	.9979	.9981	.9982	.9984	.9985	.9987	.9988
3	.9873	.9882	.9891	.9900	.9907	.9914	.9921	.9927	.9932	.9938
4	.9604	.9630	.9654	.9677	.9699	.9719	.9738	.9756	.9772	.9788
5	.9060	.9113	.9163	.9211	.9256	.9299	.9340	.9379	.9416	.9450
6	.8178	.8264	.8347	.8427	.8504	.8578	.8648	.8716	.8781	.8843
7	.6987	.7104	.7219	.7330	.7438	.7543	.7645	.7744	.7840	.7932
8	.5609	.5746	.5881	.6013	.6144	.6272	.6398	.6522	.6643	.6761
9	.4214	.4353	.4493	.4631	.4769	.4906	.5042	.5177	.5311	.5443
10	.2959	.3085	.3212	.3341	.3470	.3600	.3731	.3863	.3994	.4126
11	.1942	.2045	.2150	.2257	.2366	.2478	.2591	.2706	.2822	.2940
12	.1193	.1269	.1348	.1429	.1513	.1600	.1689	.1780	.1874	.1970
13	.0687	.0739	.0793	.0850	.0909	.0971	.1035	.1102	.1171	.1242
14	.0372	.0405	.0439	.0476	.0514	.0555	.0597	.0642	.0689	.0739
15	.0190	.0209	.0229	.0251	.0274	.0299	.0325	.0353	.0383	.0415
16	.0092	.0102	.0113	.0125	.0138	.0152	.0168	.0184	.0202	.0220
17	.0042	.0047	.0053	.0059	.0066	.0074	.0082	.0091	.0101	.0111
18	.0018	.0021	.0023	.0027	.0030	.0034	.0038	.0043	.0048	.0053
19	.0008	.0009	.0010	.0011	.0013	.0015	.0017	.0019	.0022	.0024
20	.0003	.0003	.0004	.0005	.0005	.0006	.0007	.0008	.0009	.0011
21	.0001	.0001	.0002	.0002	.0002	.0002	.0003	.0003	.0004	.0004
22	.0000	.0000	.0001	.0001	.0001	.0001	.0001	.0001	.0002	.0002
23	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0001
x'	9.1	9.2	9.3	9.4	9.5 ^m	9.6	9.7	9.8	9.9	10
0	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
1	.9999	.9999	.9999	.9999	.9999	.9999	.9999	.9999	1.0000	1.0000
2	.9989	.9990	.9991	.9991	.9992	.9993	.9993	.9994	.9995	.9995
3	.9942	.9947	.9951	.9955	.9958	.9962	.9965	.9967	.9970	.9972
4	.9802	.9816	.9828	.9840	.9851	.9862	.9871	.9880	.9889	.9897
5	.9483	.9514	.9544	.9571	.9597	.9622	.9645	.9667	.9688	.9707
6	.8902	.8959	.9014	.9065	.9115	.9162	.9207	.9250	.9290	.9329
7	.8022	.8108	.8192	.8273	.8351	.8426	.8498	.8567	.8634	.8699
8	.6877	.6990	.7101	.7208	.7313	.7416	.7515	.7612	.7706	.7798
9	.5574	.5704	.5832	.5958	.6082	.6204	.6324	.6442	.6558	.6672

SUMMED POISSON DISTRIBUTION FUNCTION $\sum_{x=x'}^{x=\infty} \frac{e^{-m} m^x}{(x)!}$

x'	m									
	9.1	9.2	9.3	9.4	9.5	9.6	9.7	9.8	9.9	10
10	.4258	.4389	.4521	.4651	.4782	.4911	.5040	.5168	.5295	.5421
11	.3059	.3180	.3301	.3424	.3547	.3671	.3795	.3920	.4045	.4170
12	.2068	.2168	.2270	.2374	.2480	.2588	.2697	.2807	.2919	.3032
13	.1316	.1393	.1471	.1552	.1636	.1721	.1809	.1899	.1991	.2084
14	.0790	.0844	.0900	.0958	.1019	.1081	.1147	.1214	.1284	.1355
15	.0448	.0483	.0520	.0559	.0600	.0643	.0688	.0735	.0784	.0835
16	.0240	.0262	.0285	.0309	.0335	.0362	.0391	.0421	.0454	.0487
17	.0122	.0135	.0148	.0162	.0177	.0194	.0211	.0230	.0249	.0270
18	.0059	.0066	.0073	.0081	.0089	.0098	.0108	.0119	.0130	.0143
19	.0027	.0031	.0034	.0038	.0043	.0048	.0053	.0059	.0065	.0072
20	.0012	.0014	.0015	.0017	.0020	.0022	.0025	.0028	.0031	.0035
21	.0005	.0006	.0007	.0008	.0009	.0010	.0011	.0013	.0014	.0016
22	.0002	.0002	.0003	.0003	.0004	.0004	.0005	.0005	.0006	.0007
23	.0001	.0001	.0001	.0001	.0001	.0002	.0002	.0002	.0003	.0003
24	.0000	.0000	.0000	.0000	.0001	.0001	.0001	.0001	.0001	.0001
x'	m									
	11	12	13	14	15	16	17	18	19	20
0	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
1	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
2	.9998	.9999	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
3	.9988	.9995	.9998	.9999	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
4	.9951	.9977	.9990	.9995	.9998	.9999	1.0000	1.0000	1.0000	1.0000
5	.9849	.9924	.9963	.9982	.9991	.9996	.9998	.9999	1.0000	1.0000
6	.9625	.9797	.9893	.9945	.9972	.9986	.9993	.9997	.9998	.9999
7	.9214	.9542	.9741	.9858	.9924	.9960	.9979	.9990	.9995	.9997
8	.8568	.9105	.9460	.9684	.9820	.9900	.9946	.9971	.9985	.9992
9	.7680	.8450	.9002	.9379	.9626	.9780	.9874	.9929	.9961	.9979
10	.6595	.7576	.8342	.8906	.9301	.9567	.9739	.9846	.9911	.9950
11	.5401	.6528	.7483	.8243	.8815	.9226	.9509	.9696	.9817	.9892
12	.4207	.5384	.6468	.7400	.8152	.8730	.9153	.9451	.9653	.9786
13	.3113	.4240	.5369	.6415	.7324	.8069	.8650	.9083	.9394	.9610
14	.2187	.3185	.4270	.5356	.6368	.7255	.7991	.8574	.9016	.9339
15	.1460	.2280	.3249	.4296	.5343	.6325	.7192	.7919	.8503	.8951
16	.0926	.1556	.2364	.3306	.4319	.5333	.6285	.7133	.7852	.8435
17	.0559	.1013	.1645	.2441	.3359	.4340	.5323	.6250	.7080	.7789
18	.0322	.0630	.1095	.1728	.2511	.3407	.4360	.5314	.6216	.7030
19	.0177	.0374	.0698	.1174	.1805	.2577	.3450	.4378	.5305	.6186
20	.0093	.0213	.0427	.0765	.1248	.1878	.2637	.3491	.4394	.5297
21	.0047	.0116	.0250	.0479	.0830	.1318	.1945	.2693	.3528	.4409
22	.0023	.0061	.0141	.0288	.0531	.0892	.1385	.2009	.2745	.3563
23	.0010	.0030	.0076	.0167	.0327	.0582	.0953	.1449	.2069	.2794
24	.0005	.0015	.0040	.0093	.0195	.0367	.0733	.1011	.1510	.2125
25	.0002	.0007	.0020	.0050	.0112	.0223	.0406	.0683	.1067	.1568
26	.0001	.0003	.0010	.0026	.0062	.0131	.0252	.0446	.0731	.1122
27	.0000	.0001	.0005	.0013	.0033	.0075	.0152	.0282	.0486	.0779
28	.0000	.0001	.0002	.0006	.0017	.0041	.0088	.0173	.0313	.0525
29	.0000	.0000	.0001	.0003	.0009	.0022	.0050	.0103	.0195	.0343
30	.0000	.0000	.0000	.0001	.0004	.0011	.0027	.0059	.0118	.0218
31	.0000	.0000	.0000	.0001	.0002	.0006	.0014	.0033	.0070	.0135
32	.0000	.0000	.0000	.0000	.0001	.0003	.0007	.0018	.0040	.0081
33	.0000	.0000	.0000	.0000	.0000	.0001	.0004	.0010	.0022	.0047
34	.0000	.0000	.0000	.0000	.0000	.0001	.0002	.0005	.0012	.0027
35	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0002	.0006	.0015
36	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0003	.0008
37	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0002	.0004
38	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0002
39	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001
40	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001

SUMMED BINOMIAL DISTRIBUTION FUNCTION

These table entries are values of

$$F(x') = \sum_{x=x'}^{x=n} \binom{n}{x} p^x q^{n-x}$$

for the specified values n , x' , and p . If $p > 0.5$, the

values for $\sum_{x=x'}^{x=n} \binom{n}{x} p^x q^{n-x}$ are obtained by using

the corresponding results obtained from

$$1 - \sum_{x=n-x'+1}^{x=n} p^{n-x} q^{x'}$$

For example, if $m=10$, $x'=4$ and $p = .15$, then $F(x') = .0500$.

n	x'	p									
		.05	.10	.15	.20	.25	.30	.35	.40	.45	.50
2	1	.0975	.1900	.2775	.3600	.4375	.5100	.5775	.6400	.6975	.7500
	2	.0025	.0100	.0225	.0400	.0625	.0900	.1225	.1600	.2025	.2500
3	1	.1426	.2710	.3859	.4880	.5781	.6570	.7254	.7800	.8336	.8750
	2	.0072	.0280	.0608	.1040	.1562	.2160	.2818	.3520	.4252	.5000
	3	.0001	.0010	.0034	.0080	.0156	.0270	.0429	.0640	.0911	.1250
4	1	.1855	.3439	.4780	.5904	.6836	.7599	.8215	.8704	.9085	.9375
	2	.0140	.0523	.1095	.1808	.2617	.3483	.4370	.5248	.6090	.6875
	3	.0005	.0037	.0120	.0272	.0508	.0837	.1265	.1792	.2415	.3125
	4	.0000	.0001	.0005	.0016	.0039	.0081	.0150	.0256	.0410	.0625
5	1	.2262	.4095	.5563	.6723	.7627	.8319	.8840	.9222	.9497	.9688
	2	.0226	.0815	.1648	.2627	.3672	.4718	.5716	.6630	.7438	.8125
	3	.0012	.0086	.0266	.0579	.1035	.1631	.2352	.3174	.4069	.5000
	4	.0000	.0005	.0022	.0067	.0156	.0308	.0540	.0870	.1312	.1875
	5	.0000	.0000	.0001	.0003	.0010	.0024	.0053	.0102	.0185	.0312
6	1	.2649	.4686	.6229	.7379	.8220	.8824	.9246	.9533	.9723	.9844
	2	.0328	.1143	.2235	.3446	.4661	.5798	.6809	.7667	.8364	.8906
	3	.0022	.0158	.0473	.0989	.1694	.2557	.3529	.4557	.5585	.6562
	4	.0001	.0013	.0059	.0170	.0376	.0705	.1174	.1792	.2553	.3438
	5	.0000	.0001	.0004	.0016	.0046	.0109	.0223	.0410	.0692	.1094
	6	.0000	.0000	.0000	.0001	.0002	.0007	.0018	.0041	.0083	.0156
7	1	.3017	.5217	.6794	.7903	.8665	.9176	.9510	.9720	.9848	.9922
	2	.0444	.1497	.2834	.4233	.5551	.6706	.7662	.8414	.8976	.9375
	3	.0038	.0257	.0738	.1480	.2436	.3529	.4677	.5801	.6836	.7734
	4	.0002	.0027	.0121	.0333	.0706	.1260	.1998	.2898	.3917	.5000
	5	.0000	.0002	.0012	.0047	.0129	.0288	.0556	.0963	.1529	.2266
	6	.0000	.0000	.0001	.0004	.0013	.0038	.0090	.0188	.0357	.0625
	7	.0000	.0000	.0000	.0000	.0001	.0002	.0006	.0016	.0037	.0078
8	1	.3366	.5956	.7275	.8322	.8999	.9424	.9681	.9832	.9916	.9961
	2	.0572	.1869	.3428	.4967	.6329	.7447	.8309	.8936	.9368	.9648
	3	.0058	.0381	.1052	.2031	.3215	.4482	.5722	.6846	.7799	.8555
	4	.0004	.0050	.0214	.0563	.1138	.1941	.2936	.4059	.5230	.6367
	5	.0000	.0004	.0029	.0104	.0273	.0580	.1061	.1737	.2604	.3633
	6	.0000	.0000	.0002	.0012	.0042	.0113	.0253	.0498	.0885	.1445
	7	.0000	.0000	.0000	.0001	.0004	.0013	.0036	.0085	.0181	.0352
	8	.0000	.0000	.0000	.0000	.0000	.0001	.0002	.0007	.0017	.0039

Linear interpolation will be accurate at most to two decimal places.

Reproduced, with permission, from "Handbook of Mathematical Tables", Chemical Rubber Publishing Company, pages 441-444.

SUMMED BINOMIAL DISTRIBUTION FUNCTION

$$\sum_{x=x'}^{x=n} \binom{n}{x} p^x q^{n-x}, \quad p + q = 1$$

<i>n</i> <i>x'</i>	.05	.10	.15	.20	.25	<i>p</i>	.30	.35	.40	.45	.50
9 1	.3698	.6126	.7684	.8658	.9249		.9596	.9793	.9899	.9954	.9980
2	.0712	.2252	.4005	.5638	.6997		.8040	.8789	.9295	.9615	.9805
3	.0084	.0530	.1409	.2618	.3993		.5372	.6627	.7682	.8505	.9102
4	.0006	.0083	.0339	.0856	.1657		.2703	.3911	.5174	.6386	.7461
5	.0000	.0009	.0056	.0196	.0489		.0988	.1717	.2666	.3786	.5000
6	.0000	.0001	.0006	.0031	.0100		.0253	.0536	.0994	.1658	.2539
7	.0000	.0000	.0000	.0003	.0013		.0043	.0112	.0250	.0498	.0898
8	.0000	.0000	.0000	.0000	.0001		.0004	.0014	.0038	.0091	.0195
9	.0000	.0000	.0000	.0000	.0000		.0000	.0001	.0003	.0008	.0020
10 1	.4013	.6513	.8031	.8926	.9437		.9718	.9865	.9940	.9975	.9990
2	.0861	.2639	.4557	.6242	.7560		.8507	.9140	.9536	.9767	.9893
3	.0115	.0702	.1798	.3222	.4744		.6172	.7384	.8327	.9004	.9453
4	.0010	.0128	.0500	.1209	.2241		.3504	.4862	.6177	.7340	.8281
5	.0001	.0016	.0099	.0328	.0781		.1503	.2845	.3669	.4956	.6230
6	.0000	.0001	.0014	.0064	.0197		.0473	.0949	.1662	.2616	.3770
7	.0000	.0000	.0001	.0009	.0035		.0106	.0260	.0548	.1020	.1719
8	.0000	.0000	.0000	.0001	.0004		.0016	.0048	.0123	.0274	.0547
9	.0000	.0000	.0000	.0000	.0000		.0001	.0005	.0017	.0045	.0107
10	.0000	.0000	.0000	.0000	.0000		.0000	.0000	.0001	.0003	.0010
11 1	.4312	.6862	.8327	.9141	.9578		.9802	.9912	.9964	.9986	.9995
2	.1019	.3026	.5078	.6779	.8029		.8870	.9394	.9698	.9861	.9941
3	.0152	.0896	.2212	.3826	.5448		.6873	.7999	.8811	.9348	.9673
4	.0016	.0185	.0694	.1611	.2867		.4304	.5744	.7037	.8089	.8867
5	.0001	.0028	.0139	.0504	.1146		.2103	.3317	.4672	.6029	.7256
6	.0000	.0003	.0027	.0117	.0343		.0782	.1487	.2465	.3669	.5000
7	.0000	.0000	.0003	.0020	.0076		.0216	.0501	.0994	.1738	.2744
8	.0000	.0000	.0000	.0002	.0012		.0043	.0122	.0293	.0610	.1133
9	.0000	.0000	.0000	.0000	.0001		.0006	.0020	.0059	.0148	.0327
10	.0000	.0000	.0000	.0000	.0000		.0000	.0002	.0007	.0022	.0059
11	.0000	.0000	.0000	.0000	.0000		.0000	.0000	.0000	.0002	.0005
12 1	.4596	.7176	.8578	.9313	.9683		.9862	.9943	.9978	.9992	.9998
2	.1184	.3410	.5565	.7251	.8416		.9150	.9576	.9804	.9917	.9968
3	.0196	.1109	.2642	.4417	.6093		.7472	.8487	.9166	.9579	.9807
4	.0022	.0256	.0922	.2054	.3512		.5075	.6533	.7747	.8655	.9270
5	.0002	.0043	.0239	.0726	.1576		.2763	.4167	.5618	.6956	.8062
6	.0000	.0005	.0046	.0194	.0544		.1178	.2127	.3348	.4731	.6128
7	.0000	.0001	.0007	.0039	.0143		.0386	.0846	.1582	.2607	.3872
8	.0000	.0000	.0001	.0006	.0028		.0095	.0255	.0573	.1117	.1938
9	.0000	.0000	.0000	.0001	.0004		.0017	.0056	.0153	.0356	.0730
10	.0000	.0000	.0000	.0000	.0000		.0002	.0008	.0028	.0079	.0193
11	.0000	.0000	.0000	.0000	.0000		.0000	.0001	.0003	.0011	.0032
12	.0000	.0000	.0000	.0000	.0000		.0000	.0000	.0000	.0001	.0002
13 1	.4867	.7458	.8791	.9450	.9762		.9903	.9963	.9987	.9996	.9999
2	.1354	.3787	.6017	.7664	.8733		.9363	.9704	.9874	.9951	.9983
3	.0245	.1339	.2704	.4983	.6674		.7975	.8868	.9421	.9731	.9888
4	.0031	.0342	.0967	.2527	.4157		.5794	.7217	.8314	.9071	.9539
5	.0003	.0065	.0260	.0991	.2060		.3457	.4995	.6470	.7721	.8666
6	.0000	.0009	.0053	.0300	.0802		.1654	.2841	.4256	.5732	.7095
7	.0000	.0001	.0013	.0070	.0243		.0624	.1295	.2288	.3653	.5000
8	.0000	.0000	.0002	.0012	.0056		.0182	.0462	.0977	.1788	.2905
9	.0000	.0000	.0000	.0002	.0010		.0040	.0126	.0321	.0698	.1334
10	.0000	.0000	.0000	.0000	.0001		.0007	.0025	.0078	.0203	.0461
11	.0000	.0000	.0000	.0000	.0000		.0001	.0003	.0013	.0041	.0112
12	.0000	.0000	.0000	.0000	.0000		.0000	.0000	.0001	.0005	.0017
13	.0000	.0000	.0000	.0000	.0000		.0000	.0000	.0000	.0000	.0001

SUMMED BINOMIAL DISTRIBUTION FUNCTION

$$\sum_{x=x'}^{x=n} \binom{n}{x} p^x q^{n-x}, \quad p + q = 1$$

<i>n</i>	<i>x'</i>	.05	.10	.15	.20	.25	.30	.35	.40	.45	.50
14	1	.5123	.7712	.8972	.9560	.9822	.9932	.9976	.9992	.9998	.9999
	2	.1530	.4154	.6433	.8021	.8990	.9525	.9795	.9919	.9971	.9991
	3	.0301	.1584	.3521	.5519	.7189	.8392	.9161	.9602	.9830	.9935
	4	.0042	.0441	.1465	.3018	.4787	.6448	.7795	.8757	.9368	.9713
	5	.0004	.0092	.0467	.1298	.2585	.4158	.5773	.7207	.8328	.9102
	6	.0000	.0015	.0115	.0439	.1117	.2195	.3595	.5141	.6627	.7880
	7	.0000	.0002	.0022	.0116	.0383	.0933	.1836	.3075	.4539	.6047
	8	.0000	.0000	.0003	.0024	.0103	.0315	.0753	.1501	.2586	.3953
	9	.0000	.0000	.0000	.0004	.0022	.0083	.0243	.0583	.1189	.2120
	10	.0000	.0000	.0000	.0000	.0003	.0017	.0060	.0175	.0426	.0898
	11	.0000	.0000	.0000	.0000	.0000	.0002	.0011	.0039	.0114	.0287
	12	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0006	.0022	.0065
	13	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0003	.0009
	14	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001
15	1	.5367	.7941	.9126	.9648	.9866	.9953	.9984	.9995	.9999	1.0000
	2	.1710	.4510	.6814	.8329	.9198	.9647	.9858	.9948	.9983	.9995
	3	.0362	.1841	.3958	.6020	.7639	.8732	.9383	.9729	.9893	.9963
	4	.0055	.0556	.1773	.3518	.5387	.7031	.8273	.9095	.9576	.9824
	5	.0006	.0127	.0617	.1642	.3135	.4845	.6481	.7827	.8796	.9408
	6	.0001	.0022	.0168	.0611	.1484	.2784	.4357	.5968	.7392	.8491
	7	.0000	.0003	.0036	.0181	.0566	.1311	.2452	.3902	.5478	.6964
	8	.0000	.0000	.0006	.0042	.0173	.0500	.1132	.2131	.3465	.5000
	9	.0000	.0000	.0001	.0008	.0042	.0152	.0422	.0950	.1818	.3036
	10	.0000	.0000	.0000	.0001	.0008	.0037	.0124	.0338	.0769	.1509
	11	.0000	.0000	.0000	.0000	.0001	.0007	.0028	.0093	.0255	.0592
	12	.0000	.0000	.0000	.0000	.0000	.0001	.0005	.0019	.0063	.0176
	13	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0003	.0011	.0037
	14	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0005
	15	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
16	1	.5599	.8147	.9257	.9719	.9900	.9967	.9990	.9997	.9999	1.0000
	2	.1892	.4853	.7161	.8593	.9365	.9739	.9902	.9967	.9990	.9997
	3	.0429	.2108	.4386	.6482	.8029	.9006	.9549	.9817	.9934	.9979
	4	.0070	.0684	.2101	.4019	.5950	.7541	.8661	.9349	.9719	.9894
	5	.0009	.0170	.0791	.2018	.3698	.5501	.7108	.8334	.9147	.9616
	6	.0001	.0033	.0235	.0817	.1897	.3402	.5100	.6712	.8024	.8949
	7	.0000	.0005	.0056	.0267	.0796	.1753	.3119	.4728	.6340	.7228
	8	.0000	.0001	.0011	.0070	.0271	.0744	.1594	.2839	.4371	.5892
	9	.0000	.0000	.0002	.0015	.0075	.0257	.0671	.1423	.2559	.4018
	10	.0000	.0000	.0000	.0002	.0016	.0071	.0229	.0583	.1241	.2272
	11	.0000	.0000	.0000	.0000	.0003	.0016	.0062	.0191	.0486	.1051
	12	.0000	.0000	.0000	.0000	.0000	.0003	.0013	.0049	.0149	.0384
	13	.0000	.0000	.0000	.0000	.0000	.0000	.0002	.0009	.0035	.0106
	14	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0006	.0021
	15	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0003
	16	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
17	1	.5819	.8332	.9369	.9775	.9925	.9977	.9993	.9998	1.0000	1.0000
	2	.2078	.5182	.7475	.8818	.9499	.9807	.9933	.9979	.9994	.9999
	3	.0503	.2382	.4802	.6904	.8363	.9226	.9673	.9877	.9959	.9988
	4	.0088	.0826	.2444	.4511	.6470	.7981	.8972	.9536	.9816	.9936
	5	.0012	.0221	.0987	.2418	.4261	.6113	.7652	.8740	.9404	.9755
	6	.0001	.0047	.0319	.1057	.2347	.4032	.5803	.7361	.8529	.9283
	7	.0000	.0008	.0083	.0377	.1071	.2248	.3812	.5522	.7098	.8338
	8	.0000	.0001	.0017	.0109	.0402	.1046	.2128	.3595	.5257	.6855
	9	.0000	.0000	.0003	.0026	.0124	.0403	.0994	.1989	.3374	.5000
	10	.0000	.0000	.0000	.0005	.0031	.0127	.0383	.0919	.1834	.3145
	11	.0000	.0000	.0000	.0001	.0006	.0032	.0120	.0348	.0826	.1662
	12	.0000	.0000	.0000	.0000	.0001	.0007	.0030	.0106	.0301	.0717
	13	.0000	.0000	.0000	.0000	.0000	.0001	.0006	.0025	.0086	.0245
	14	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0005	.0019	.0064
	15	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0003	.0012
	16	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001
	17	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000

SUMMED BINOMIAL DISTRIBUTION FUNCTION

$$\sum_{x=x}^{x=n} \binom{n}{x} p^x q^{n-x}, \quad p + q = 1$$

n x'	p									
	.05	.10	.15	.20	.25	.30	.35	.40	.45	.50
18 1	.6028	.8499	.9464	.9820	.9944	.9984	.9996	.9999	1.0000	1.0000
2	.2265	.5497	.7759	.9009	.9605	.9858	.9954	.9987	.9997	.9999
3	.0581	.2662	.5203	.7287	.8647	.9400	.9764	.9918	.9975	.9993
4	.0109	.0982	.2798	.4990	.6943	.8354	.9217	.9672	.9880	.9962
5	.0015	.0282	.1206	.2836	.4813	.6673	.8114	.9058	.9589	.9846
6	.0002	.0064	.0419	.1329	.2825	.4656	.6450	.7912	.8923	.9519
7	.0000	.0012	.0118	.0513	.1390	.2783	.4509	.6257	.7742	.8811
8	.0000	.0002	.0027	.0163	.0569	.1407	.2717	.4366	.6085	.7597
9	.0000	.0000	.0005	.0043	.0193	.0596	.1391	.2632	.4222	.5927
10	.0000	.0000	.0001	.0009	.0054	.0210	.0597	.1347	.2527	.4073
11	.0000	.0000	.0000	.0002	.0012	.0061	.0212	.0576	.1280	.2403
12	.0000	.0000	.0000	.0000	.0002	.0014	.0062	.0203	.0537	.1189
13	.0000	.0000	.0000	.0000	.0000	.0003	.0014	.0058	.0183	.0481
14	.0000	.0000	.0000	.0000	.0000	.0000	.0003	.0013	.0049	.0154
15	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0002	.0010	.0038
16	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0007
17	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001
18	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
19 1	.6226	.8649	.9544	.9856	.9958	.9989	.9997	.9999	1.0000	1.0000
2	.2453	.5797	.8015	.9171	.9690	.9896	.9969	.9992	.9998	1.0000
3	.0665	.2946	.5587	.7631	.8887	.9538	.9830	.9945	.9985	.9996
4	.0132	.1150	.3159	.5449	.7369	.8668	.9409	.9770	.9923	.9978
5	.0020	.0352	.1444	.3267	.5346	.7178	.8500	.9304	.9720	.9904
6	.0002	.0086	.0537	.1631	.3322	.5261	.7032	.8371	.9223	.9682
7	.0000	.0017	.0163	.0676	.1749	.3345	.5188	.6919	.8273	.9165
8	.0000	.0003	.0041	.0233	.0775	.1820	.3344	.5122	.6831	.8204
9	.0000	.0000	.0008	.0067	.0287	.0839	.1855	.3325	.5060	.6762
10	.0000	.0000	.0001	.0016	.0089	.0326	.0875	.1861	.3290	.5000
11	.0000	.0000	.0000	.0003	.0023	.0105	.0347	.0885	.1841	.3283
12	.0000	.0000	.0000	.0000	.0005	.0028	.0114	.0352	.0871	.1796
13	.0000	.0000	.0000	.0000	.0001	.0006	.0031	.0116	.0342	.0835
14	.0000	.0000	.0000	.0000	.0000	.0001	.0007	.0031	.0109	.0318
15	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0006	.0028	.0096
16	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0005	.0022
17	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0004
18	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
19	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
20 1	.6415	.8784	.9612	.9885	.9968	.9992	.9998	1.0000	1.0000	1.0000
2	.2642	.6083	.8244	.9308	.9757	.9924	.9979	.9995	.9999	1.0000
3	.0755	.3231	.5951	.7939	.9087	.9645	.9879	.9964	.9991	.9998
4	.0159	.1330	.3523	.5886	.7748	.8929	.9556	.9840	.9951	.9987
5	.0026	.0432	.1702	.3704	.5852	.7625	.8818	.9490	.9811	.9941
6	.0003	.0113	.0673	.1958	.3828	.5836	.7546	.8744	.9447	.9793
7	.0000	.0024	.0219	.0867	.2142	.3920	.5834	.7500	.8701	.9423
8	.0000	.0004	.0059	.0321	.1018	.2277	.3990	.5841	.7480	.8684
9	.0000	.0001	.0013	.0100	.0409	.1133	.2376	.4044	.5857	.7483
10	.0000	.0000	.0002	.0026	.0139	.0480	.1218	.2447	.4086	.5881
11	.0000	.0000	.0000	.0006	.0039	.0171	.0532	.1275	.2493	.4119
12	.0000	.0000	.0000	.0001	.0009	.0051	.0196	.0565	.1308	.2517
13	.0000	.0000	.0000	.0000	.0002	.0013	.0060	.0210	.0580	.1316
14	.0000	.0000	.0000	.0000	.0000	.0003	.0015	.0065	.0214	.0577
15	.0000	.0000	.0000	.0000	.0000	.0000	.0003	.0016	.0064	.0207
16	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0003	.0015	.0059
17	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0003	.0013
18	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0002
19	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
20	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000

RANDOM UNITS (UNIFORM)

Use of Table. — If one wishes to select a random sample of N items from a universe of M items, the following procedure may be applied. ($M > N$.)

1. Decide upon some arbitrary scheme of selecting entries from the table. For example, one may decide to use the entries in the first line, second column; second line, third column; third line, fourth column; etc.

2. Assign numbers to each of the items in the universe from 1 to M . Thus, if $M = 500$, the items would be numbered from 001 to 500, and therefore, each designated item is associated with a three-digit number.

3. Decide upon some arbitrary scheme of selecting positional digits from each entry chosen according to Step 1. Thus, if $M = 500$, one may decide to use the first, third and fourth digit of each entry selected, and as a consequence a three-digit number is created for each entry choice.

4. If the number formed is $\leq M$, the correspondingly designated item in the universe is chosen for the random sample of N items. If a number formed is $> M$ or is a repeated number of one already chosen, it is passed over and the next desirable number is taken. This process is continued until the random sample of N items is selected.

Reproduced, with permission, from "Handbook of Mathematical Tables", Chemical Rubber Publishing Company, pages 277-283, and from Statement No. 4914, "Table of 105,000 Random Decimal Digits", 1949, Interstate Commerce Commission, Bureau of Transport and Statistics, pages 1-4.

RANDOM UNITS (UNIFORM)

Line/Col.	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)
1	10480	15011	01536	02011	81647	91646	69179	14194	62590	36207	20969	99570	91291	90700
2	22368	46573	25595	85393	30995	89198	27982	53402	93965	34095	52666	19174	39615	99505
3	24130	48360	22527	97265	76393	64809	15179	24830	49340	32081	30680	19655	63348	58629
4	42167	93093	06243	61680	07856	16376	39440	53537	71341	57004	00849	74917	97758	16379
5	37570	39975	81837	16656	06121	91782	60468	81305	49684	60672	14110	06927	01263	54613
6	77921	06907	11008	42751	27756	53498	18602	70659	90655	15053	21916	81825	44394	42880
7	99562	72905	56420	69994	98872	31016	71194	18738	44013	48840	63213	21069	10634	12952
8	96301	91977	05463	07972	18876	20922	94595	56869	69014	60045	18425	84903	42508	32307
9	89579	14342	63661	10281	17453	18103	57740	84378	25331	12566	58678	44947	05585	56941
10	85475	36857	53342	53988	53060	59533	38867	62300	08158	17983	16439	11458	18593	64952
11	28918	69578	88231	33276	70997	79936	56865	05859	90106	31595	01547	85590	91610	78188
12	63553	40961	48235	03427	49626	69445	18663	72695	52180	20847	12234	90511	33703	90322
13	09429	93969	52636	92737	88974	33488	36320	17617	30015	08272	84115	27156	30613	74952
14	10365	61129	87529	85689	48237	52267	67689	93394	01511	26358	85104	20285	29975	89868
15	07119	97336	71048	08178	77233	13916	47564	81056	97735	85977	29372	74461	28551	90707
16	51085	12765	51821	51259	77452	16308	60756	92144	49442	53900	70960	63990	75601	40719
17	02368	21382	52404	60268	89368	19885	55322	44819	01188	65255	64835	44919	05944	55157
18	01011	54092	33362	94904	31273	04146	18594	29852	71585	85030	51132	01915	92747	64951
19	52162	53916	46369	58586	23216	14513	83149	98736	23495	64350	94738	17752	35156	35749
20	07056	97628	33787	09998	42698	06691	76988	13602	51851	46104	88916	19509	25625	58104
21	48663	91245	85828	14346	09172	30168	90229	04734	59193	22178	30421	61666	99904	32812
22	54164	58492	22421	74103	47070	25306	76468	26384	58151	06646	21524	15227	96909	44592
23	32639	32363	05597	24200	13363	38005	94342	28728	35806	06912	17012	64161	18296	22851
24	29334	27001	87637	97308	58731	00256	45834	15398	46557	41135	10367	07684	36188	18510
25	02488	33062	28834	07351	19731	92420	60952	61280	50001	67658	32586	86679	50720	94953

Line/Col.	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)
26	81525	72295	04839	96423	24878	82651	66566	14778	76797	14780	13300	87074	79666	95725
27	29676	20591	68086	26432	46901	20849	89768	81536	86645	12659	92259	57102	80428	25280
28	00742	57392	39064	66432	84673	40027	32832	61362	98947	96067	64760	64584	96096	98253
29	05366	04213	25669	26422	44407	44048	37937	63904	45766	66134	75470	66520	34693	90449
30	91921	26418	64117	94305	26766	25940	39972	22209	71500	64568	91402	42416	07844	69618
31	00582	04711	87917	77341	42206	35126	74087	99547	81817	42607	43808	76655	62028	76630
32	00725	69884	62797	56170	86324	88072	76222	36086	84637	93161	76038	65855	77919	88006
33	69011	65795	95876	55293	18988	27354	26575	08625	40801	59920	29841	80150	12777	48501
34	25976	57948	29888	88604	67917	48708	18912	82271	65424	69774	33611	54262	85963	03547
35	09763	83473	73557	12908	30883	18317	28290	35797	05998	41688	34952	37888	38917	88050
36	91567	42595	27958	30134	04024	86385	29880	99730	55536	84855	29080	09250	97656	73211
37	17955	56349	90999	49127	20044	59931	06115	20542	18059	02008	73708	83517	36103	42791
38	46503	18584	18845	49618	02304	51038	20655	58727	28168	15475	56942	53389	20562	87338
39	92157	89634	94824	78171	84610	82834	09922	25417	44137	48413	25555	21246	35509	20468
40	14577	62765	35605	81263	39667	47358	65873	56307	61607	49518	89656	20103	77490	18062
41	98427	07523	33362	64270	01638	92477	66969	98420	04880	45585	46565	04102	46880	45709
42	34914	63976	88720	82765	34476	17032	87589	40836	32427	70002	70663	88863	77775	69348
43	70060	28277	39475	46473	23219	53416	94970	25832	69975	94884	19661	72828	00102	66794
44	53976	54914	06990	67245	68350	82948	11398	42878	80287	88267	47363	46634	06541	97809
45	76072	29515	40980	07391	58745	25774	22987	80059	39911	96189	41151	14222	60697	59583
46	90725	52210	83974	29992	65831	38857	50490	83765	55657	14361	31720	57375	56228	41546
47	64364	67412	33339	31926	14883	24413	59744	92351	97473	89286	35931	04110	23726	51900
48	08962	00358	31662	25388	61642	34072	81249	35648	56891	69352	48373	45578	78547	81788
49	95012	68379	93526	70765	10592	04542	76463	54328	02349	17247	28865	14777	62730	92277
50	15664	10493	20492	38391	91132	21999	59516	81652	27195	48223	46751	22923	32261	85653
51	16408	81899	04153	53381	79401	24438	83035	92350	36693	31238	59649	91754	72772	02338
52	18629	81953	05520	91962	04739	13092	97662	24822	94730	06496	35090	04822	86774	98289
53	73115	35101	47498	87637	99016	71060	88824	71013	18735	20286	23153	72924	35165	43040
54	57491	16703	23167	49323	45021	33132	12544	41035	80780	45393	44812	12515	98931	91202
55	30405	83946	23792	14422	15059	45799	22716	19792	09983	74353	68668	30429	70735	25499
56	16631	35006	85900	98275	32388	52390	16815	69298	82732	38480	73817	32523	41961	44437
57	96773	20206	42559	78985	05300	22164	24369	54224	35083	19687	11052	91491	60383	19746
58	38935	64202	14349	82674	66523	44133	00697	35522	35970	19124	63318	29686	03387	59846
59	31624	76384	17403	53363	44167	64486	64758	75366	76554	31601	12614	33072	60332	92325
60	78919	19474	23632	27889	47914	02584	37680	20801	72152	39339	34806	08930	85001	87820

Reproduced, with permission, from "Handbook of Mathematical Tables", Chemical Rubber Publishing Company, pages 277-283, and from Statement No. 4914, "Table of 105,000 Random Decimal Digits", 1949, Interstate Commerce Commission, Bureau of Transport and Statistics, pages 1-4.

RANDOM UNITS (UNIFORM)

Line/Col.	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)
61	03931	33309	57047	74211	63445	17361	62825	39908	05607	91284	68833	25570	38818	46920
62	74426	33278	43972	10119	89917	15665	52872	73823	73144	88662	88970	74492	51805	99378
63	09066	00903	20795	95452	92648	45454	09552	88815	16553	51125	79375	97596	16296	66092
64	42238	12426	87025	14267	20979	04508	64535	31355	86064	29472	47689	05974	52468	16834
65	16153	08002	26504	41744	81959	65642	74240	56302	00033	67107	77510	70625	28725	34191
66	21457	40742	29820	96783	29400	21840	15035	34537	33310	06116	95240	15957	16572	06004
67	21581	57802	02050	89728	17937	37621	47075	42080	97403	48626	68995	43805	33386	21597
68	55612	78095	83197	33732	05810	24813	86902	60397	16489	03264	88525	42786	05269	92532
69	44657	66999	99324	51281	84463	60563	79312	93454	68876	25471	93911	25600	12682	73572
70	91340	84979	46949	81973	37949	61023	43997	15263	80644	43942	89203	71795	99533	50501
71	91227	21199	31935	27022	84067	05462	35216	14486	29891	68607	41867	14951	91696	85065
72	50001	38140	66321	19924	72163	09538	12151	06878	91903	18749	34405	56087	82790	70925
73	65390	05224	72958	28609	81406	39147	25549	48542	42627	45233	57202	94617	23772	07896
74	27504	96131	83944	41575	10573	08619	64482	73923	36152	05184	94142	25299	84387	34925
75	37169	94851	39117	89632	00959	16487	65536	49071	39782	17095	02330	74301	00275	48280
76	11508	70225	51111	38351	19444	66499	71945	05422	13442	78675	84081	66938	93654	59894
77	37449	30362	06694	54690	04052	53115	62757	95348	78662	11163	81651	50245	34971	52924
78	46515	70331	85922	38329	57015	15765	97161	17869	45349	61796	66345	81073	49106	79860
79	30986	81223	42416	58353	21532	30502	32305	86482	05174	07901	54339	58861	74818	46942
80	63798	64995	46583	09785	44160	78128	83991	42865	92520	83531	80377	35909	81250	54238
81	82486	84846	99254	67632	43218	50076	21361	64816	51202	88124	41870	52689	51275	83556
82	21885	32906	92431	09060	64297	51674	64126	62570	26123	05155	59194	52799	28225	85762
83	60336	98782	07408	53458	13564	59089	26445	29789	85205	41001	12535	12133	14645	23541
84	43937	46891	24010	25560	86355	33941	25786	54990	71899	15475	95434	98227	21824	19585
85	97656	63175	89303	16275	07100	92063	21942	18611	47348	20203	18534	03862	78095	50136
86	03299	01221	05418	38982	55758	92237	26759	86367	21216	98442	08303	56613	91511	75928
87	79626	06486	03574	17668	07785	76020	79924	25651	83325	88428	85076	72811	22717	50585
88	85636	68335	47539	03129	65651	11977	02510	26113	99447	68645	34327	15152	55230	93448
89	18039	14367	61337	06177	12143	46609	32989	74014	64708	00533	35398	58408	13261	47908
90	08362	15656	60627	36478	65648	16764	53412	09013	07832	41574	17639	82163	60859	75567
91	79556	29068	04142	16268	15387	12856	66227	38358	22478	73373	88732	09443	82558	05250
92	92608	82674	27072	32534	17075	27698	98204	63863	11951	34648	88022	56148	34925	57031
93	23982	25835	40055	67006	12293	02753	14827	23235	35071	99704	37543	11601	35503	85171
94	09915	96306	05908	97901	28395	14186	00821	80703	70426	75647	76310	88717	37890	40129
95	59037	33300	26695	62247	69927	76123	50842	43834	86654	70959	79725	93872	28117	19233

Line/Col.	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)
96	42488	78077	69882	61657	34136	79180	97526	43092	04098	73571	80799	76536	71255	64239
97	46764	86273	63003	93017	31204	36692	40202	35275	57306	55543	53203	18098	47625	88684
98	03237	45430	55417	63282	90816	17349	88298	90183	36600	78406	06216	95787	42579	90730
99	86591	81482	52667	61582	14972	90053	89534	76036	49199	43716	97548	04379	46370	28672
100	38534	01715	94964	87288	65680	43772	12918	86537	12918	62738	19636	51132	25739	56947
101	13284	16834	74151	92027	24670	36665	00770	22878	02179	51602	07270	76517	97275	45960
102	21224	00370	30420	03883	94648	89428	41583	17564	27395	63904	41548	49197	82277	24120
103	99052	47887	81085	64933	66279	80432	65793	83287	34142	13241	30590	97760	35848	91983
104	00199	50993	98603	38452	87890	94624	69721	57484	67501	77638	44331	11257	71131	11059
105	60578	06483	28733	37867	07936	98710	98539	27186	31237	80612	44488	97819	70401	95419
106	91240	18312	17441	01929	18163	69201	31211	54288	39296	37318	65724	90401	79017	62077
107	97458	14229	12063	59611	32249	90466	33216	19358	02591	54263	88449	01912	07436	50813
108	35249	38646	34475	72417	60514	69257	12489	51924	86871	92446	36607	11458	30440	52639
109	38980	46600	11759	11900	46743	27860	77940	39298	97838	95145	32378	68038	89351	37005
110	10750	52745	38749	87365	58959	53731	89295	59062	39404	13198	59960	70408	29812	83126
111	36247	27850	73958	20673	37800	63835	71051	84724	52492	22342	78071	17456	96104	18327
112	70994	66986	99744	72438	01174	42159	11392	20724	54322	36923	70009	23233	65438	59685
113	99638	94702	11463	18148	81386	80431	90628	52506	02016	85151	88598	47821	00265	82525
114	72055	15774	43857	99805	10419	76939	25993	03544	21560	83471	43989	90770	22965	44247
115	24038	65541	85788	55835	38835	59399	13790	35112	01324	39520	76210	22467	83275	32286
116	74976	14631	35908	28221	39470	91548	12854	30166	09073	75887	36782	00268	97121	57676
117	35553	71628	70189	26436	63407	91178	90348	55359	80392	41012	36270	77786	89578	21059
118	35676	12797	51434	82976	42010	26344	92920	92155	58807	54644	58581	95331	78629	73344
119	74815	67523	72985	23183	02446	63594	98924	20633	58842	85961	07648	70164	34994	67662
120	45246	88048	65173	50989	91060	89894	36036	32819	68559	99221	49475	50558	34698	71800
121	76509	47069	86378	41797	11910	49672	88575	97966	32466	10083	54728	81972	58975	30761
122	19689	90332	04315	21358	97248	11188	39062	63312	52496	07349	79178	33692	57352	72862
123	42751	35318	97513	61537	54955	08159	00337	80778	27507	95478	21252	12746	37554	97775
124	11946	22681	45045	13964	57517	59419	58045	44067	58716	58840	45557	96345	33271	53464
125	96518	48688	20996	11090	48396	57177	83867	86464	14342	21545	46717	72364	86954	55580
126	35726	58643	76869	84622	39098	36083	72505	92265	23107	60278	05822	46760	44294	07672
127	39737	42750	48968	70536	84864	64952	38404	94317	65402	13589	01055	79044	19308	83623
128	97025	66492	56177	04049	80312	48028	26408	43591	75528	65341	49004	95495	81256	53214
129	62814	08075	09788	56350	76787	51591	54509	49295	85830	59860	30883	89660	96142	18354
130	25578	22950	15227	83291	41737	79599	96191	71845	86899	70694	24290	01551	80092	82118

RANDOM UNITS (UNIFORM)

Line/Col.	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)
131	68763	69576	88991	49662	46704	63362	56625	00481	73323	91427	15264	06969	57048	54149
132	17900	00813	64361	60725	88974	61005	99709	30666	26451	11528	44323	34778	60342	60388
133	71944	60227	63551	71109	05624	43836	58254	26160	32116	63403	35404	57146	10909	07346
134	54684	93691	85132	64399	29182	44324	14491	55226	78793	34107	30374	48429	51376	09559
135	25946	27623	11258	65204	52832	50880	22273	05554	99521	73791	85744	29276	70326	60251
136	01353	39318	44961	44972	91766	90262	56073	06606	51826	18893	83448	31915	97764	75091
137	99083	88191	27662	99113	57174	35571	99884	13951	71057	53961	61448	74909	07322	80960
138	52021	45406	37945	75234	24327	86978	22644	87779	23753	99926	63898	54886	18051	96314
139	78755	47744	43776	83098	03225	14281	83637	55984	13300	52212	58781	14905	46502	04472
140	25282	69106	59180	16257	22810	43609	12224	25643	89884	31149	85423	32581	34374	70873
141	11959	94202	02743	86847	79725	51811	12998	76844	05320	54236	53891	70226	38632	84776
142	11644	13792	88190	01424	30078	28197	55583	05197	47714	68440	22016	79204	06862	94451
143	06307	97912	68110	59812	95448	43244	31262	88880	13040	16458	43813	89416	42482	33939
144	76285	75714	89585	99296	52640	46518	55486	90754	88932	19937	57119	23251	55619	23679
145	55322	07598	39600	60866	63007	20007	66819	84164	61131	81429	60676	42807	78286	29015
146	78017	90928	90220	92503	83375	26986	74399	30885	88567	29169	72816	53357	15428	86932
147	44768	43342	20696	26331	43140	69744	82928	24988	94237	46138	77426	39039	55596	12655
148	25100	19336	14605	86603	51680	97678	24261	02464	86563	74812	60069	71674	15478	47642
149	83612	46623	62876	85197	07824	91392	58317	37726	84628	42221	10268	20692	15699	29167
150	41347	81666	82961	60413	71020	83658	02415	33322	66036	98712	46795	16308	28413	05417
151	38128	51178	75096	13609	16110	73533	42564	59870	29399	67834	91055	89917	51096	89011
152	60950	00455	73254	96067	50717	13878	03216	78274	65863	37011	91283	33914	91303	49326
153	90524	17320	29832	96118	75792	25326	22940	24904	80523	38928	91374	55597	97567	38914
154	49897	18278	67160	39408	97056	43517	84426	59650	20247	19293	02019	14790	02852	05819
155	18494	99209	81060	19488	65596	59787	47939	91225	98768	43688	00438	05548	09443	82897
156	65373	72984	30171	37741	70203	94094	87261	30056	58124	70133	18936	02138	59372	09075
157	40653	12843	04213	70925	95360	55774	76439	61768	52817	81151	52188	31940	54273	49032
158	51638	22238	56344	44587	83231	50317	74541	07719	25472	41602	77318	15145	57515	07633
159	69742	99303	62578	83575	30337	07488	51941	84316	42067	49692	28616	29101	03013	73449
160	58012	74072	67488	47992	69482	69482	58624	17106	47538	13452	22620	24260	40155	74716
161	18348	19855	42887	08279	43206	47077	42637	45606	00011	20662	14642	49984	94509	56380
162	59614	09193	58064	29086	44385	45740	70752	05663	49081	26960	57454	99264	24142	74648
163	75688	28630	39210	52897	62748	72658	98059	67202	72789	01869	13496	14663	87645	89713
164	13941	77802	69101	70061	35460	34576	15412	81304	58757	35498	94830	75521	00603	97701
165	96656	86420	96475	86458	54463	96419	55417	41375	76886	19008	66877	35934	59801	00497

Line/Col	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)
166	03363	82042	15942	14549	38324	87094	19069	67590	11087	68570	22591	65232	85915	91499
167	70366	08390	69155	25496	13240	57407	91407	49160	07379	34444	94567	66035	38918	65708
168	47870	36605	12927	16043	53257	93796	52721	73120	48025	76074	95605	67422	41646	14557
169	79504	77606	22761	30518	28373	73898	30550	76684	77366	32276	04690	61667	64798	66276
170	46967	74841	50923	15339	37755	98995	40162	89561	69199	42257	11647	47603	48779	97907
171	14558	50769	35444	59030	87516	48193	02945	00922	48189	04724	21263	20892	92955	90251
172	12440	25057	01132	38611	28135	68089	10954	10097	54243	06460	50856	65435	79377	53890
173	32293	29938	68653	10497	98919	46587	77701	99119	93165	67788	17638	23097	21468	36992
174	10640	21875	72462	77981	56550	55999	87310	69643	45124	00349	25748	00844	96831	30651
175	47615	23169	39571	56972	20628	21788	51736	33133	72696	32605	41569	76148	91544	21121
176	16948	11128	71624	72754	49084	96303	27830	45817	67867	18062	87453	17226	72904	71474
177	21258	61092	66634	70335	92448	17354	83432	49608	66520	06442	59664	20420	39201	69549
178	15072	48853	15178	30730	47481	48490	41436	25015	49932	20474	53821	51015	79841	32405
179	99154	57412	09858	65671	70655	71479	63520	31357	56968	06729	34465	70685	04184	25250
180	08759	61089	23706	32994	35426	36666	63988	98844	37533	08269	27021	45886	22835	78451
181	67323	57839	61114	62192	47547	58023	64630	34886	98777	75442	95592	06141	45096	73117
182	09255	13986	84834	20764	72206	89393	34548	93438	88730	61805	78955	18952	46436	58740
183	36304	74712	00374	10107	85061	69228	81969	92216	03568	39630	81869	52824	50937	27954
184	15884	67429	86612	47367	10242	44880	12060	44309	46629	55105	66793	93173	00480	13311
185	18745	32031	35303	08134	33925	03044	59929	95418	04917	57596	24878	61733	92834	64454
186	72934	40086	88292	65728	38300	42323	64068	98373	48971	09049	59943	36538	05976	82118
187	17626	02944	20910	57662	80181	38579	24580	90529	52303	50436	29401	57824	86039	81062
188	27117	61399	50967	41399	81636	16663	15634	79717	94696	59240	25543	97989	63306	90946
189	93995	18678	90012	63645	85701	85269	62263	68331	00389	72571	15210	20769	44686	96176
190	67392	89421	09623	80725	62620	84162	87368	29560	00519	84545	08004	24526	41252	14521
191	04910	12261	37566	80016	21245	69377	50420	85658	55263	68667	78770	04533	14513	18099
192	81453	20283	79929	59839	23875	13245	46808	74124	74703	35769	95588	21014	37078	39170
193	19480	75790	48539	23703	15537	48885	02861	86587	74539	65227	90799	58789	96257	02708
194	21456	13162	74608	81011	55512	07481	93551	72189	76261	91206	89941	15132	37738	59284
195	89406	20912	46189	76376	25538	87212	20748	12831	57166	35026	16817	79121	18929	40628
196	09866	07414	55977	16419	01101	69343	13305	94302	80703	57910	36933	57771	42546	03003
197	86541	24681	23421	13521	28000	94917	07423	57523	97234	63951	42876	46829	09781	58160
198	10414	96941	06205	72222	57167	83902	07460	69507	10600	08858	07685	44472	64220	27040
199	49942	06683	41479	58982	56288	42853	92196	20632	62045	78812	35895	51851	83534	10689
200	23995	68882	42291	23374	24299	27024	67460	94783	40937	16961	26053	78749	46704	21983

MATHEMATICAL TABLES

Mathematical Symbols and Abbreviations

+	Plus or Positive	sec	Secant
-	Minus or Negative	csc	Cosecant
±	{ Plus or minus Positive or negative	vers	Versed sine
∓	{ Minus or plus Negative or positive	covers	Covered sine
×	Multipled by	exsec	Exsecant
÷ or :	Divided by	$\sin^{-1}a$ or	{ Anti-sine a Angle whose sine is a
= or ::	Equals, as	$\arcsin a$	{ Inverse sine a
≠, ≠	Does not equal	sinh	Hyperbolic sine
≈	{ Equals approximately Congruent	cosh	Hyperbolic cosine
>	Greater than	tanh	Hyperbolic tangent
<	Less than	$\sinh^{-1}a$ or	{ Anti-hyperbolic sine a Angle whose hyperbolic sine is a
≥	Greater than or equal to	$\operatorname{arcsinh} a$	
≤	Less than or equal to	$P(x,y)$	Rect. coord. of point P
∝	Similar to	$P(r, \theta)$	Polar coord. of point P
∴	Therefore	$f(x), F(x)$	{ Function of x
$\sqrt{\quad}$	Square root	or $\Phi(x)$	
$\sqrt[n]{\quad}$	n th root	Δy	Increment of y
a^n	n th power of a	\rightarrow or \rightarrow	Approaches as a limit
\log or \log_{10}	{ Common logarithm Briggsian	Σ	Summation of
\ln or \log_e	{ Natural logarithm Hyperbolic	∞	Infinity
or \log	{ Napierian	dy	Differential of y
e or e	Base (2.718) of natural system of logarithms	$\frac{dy}{dx}$ or $f'(x)$	Derivative of $y=f(x)$ with respect to x
π	Pi (3.1416)	$\frac{d^2y}{dx^2}$ or $f''(x)$	Second deriv. of $y=f(x)$ with respect to x
\angle	Angle	$\frac{d^ny}{dx^n}$ or $f^{(n)}(x)$	n th deriv. of $y = f(x)$ with respect to x
\perp	Perpendicular to	$\frac{\partial z}{\partial x}$	Partial derivative of z with respect to x
\parallel	Parallel to	$\frac{\partial^2 z}{\partial x \partial y}$	Second partial deriv. of z with respect to y and x
a°	a degrees (angle)	\int	Integral of
a'	{ a minutes (angle) a prime	\int_a^b	Integral between the limits a and b
a''	{ a seconds (angle) a second a double-prime	j or i	{ Imaginary quantity $(\sqrt{-1}), i^2 = -1$
a'''	{ a third a triple-prime	$x = a + jb$	Symbolic vector notation
a_n	a sub n	$n! = 1 \cdot 2 \cdot 3 \cdots n$	
sin	Sine		
cos	Cosine		
tan	Tangent		
cot or ctn	Cotangent		

Greek Alphabet

A	a	Alpha	N	ν	Nu
B	β	Beta	Ξ	ξ	Xi
Γ	γ	Gamma	Ο	ο	Omicron
Δ	δ	Delta	Π	π	Pi
E	ε	Epsilon	P	ρ	Rho
Z	ζ	Zeta	Σ	σ	Sigma
H	η	Eta	T	τ	Tau
Θ	θ	Theta	Υ	υ	Upsilon
I	ι	Iota	Φ	φ	Phi
K	κ	Kappa	X	χ	Chi
Λ	λ	Lambda	Ψ	ψ	Psi
M	μ	Mu	Ω	ω	Omega

Reproduced from "Handbook of Mathematical Tables and Formulas", by R. S. Burington, with permission of McGraw-Hill Book Co., Inc. © 1933, 1940 and 1948, frontispiece.

TABLE OF RECIPROALS

n	1000/n	n	1000/n	n	1000/n	n	1000/n
1	1000.000	50	20.000 00	100	10.000 000	150	6.666 667
2	500.000 0	51	19.607 84	101	9.900 990	151	6.622 517
3	333.333 3	52	19.230 77	102	9.803 922	152	6.578 947
4	250.000 0	53	18.867 92	103	9.708 738	153	6.535 948
5	200.000 0	54	18.518 52	104	9.615 385	154	6.493 506
6	166.666 7	55	18.181 82	105	9.523 810	155	6.451 613
7	142.857 1	56	17.857 14	106	9.433 962	156	6.410 296
8	125.000 0	57	17.543 86	107	9.345 794	157	6.369 427
9	111.111 1	58	17.241 38	108	9.259 259	158	6.329 114
10	100.000 0	59	16.949 15	109	9.174 312	159	6.289 308
11	90.909 09	60	16.666 67	110	9.090 909	160	6.250 000
12	83.333 33	61	16.393 44	111	9.009 009	161	6.211 180
13	76.923 08	62	16.129 03	112	8.928 571	162	6.172 840
14	71.428 57	63	15.873 02	113	8.849 558	163	6.134 969
15	66.666 67	64	15.625 00	114	8.771 930	164	6.097 561
16	62.500 00	65	15.384 62	115	8.695 652	165	6.060 606
17	58.823 53	66	15.151 52	116	8.620 690	166	6.024 096
18	55.555 56	67	14.925 37	117	8.547 009	167	5.988 024
19	52.631 58	68	14.705 88	118	8.474 576	168	5.952 381
20	50.000 00	69	14.492 75	119	8.403 361	169	5.917 160
21	47.619 05	70	14.285 71	120	8.333 333	170	5.882 353
22	45.454 55	71	14.084 51	121	8.264 463	171	5.847 953
23	43.478 26	72	13.888 89	122	8.196 721	172	5.813 953
24	41.666 67	73	13.698 63	123	8.130 081	173	5.780 347
25	40.000 00	74	13.513 51	124	8.064 516	174	5.747 126
26	38.461 54	75	13.333 33	125	8.000 000	175	5.714 286
27	37.037 04	76	13.157 89	126	7.936 508	176	5.681 818
28	35.714 29	77	12.987 01	127	7.874 016	177	5.649 718
29	34.482 76	78	12.820 51	128	7.812 500	178	5.617 978
30	33.333 33	79	12.658 23	129	7.751 938	179	5.586 592
31	32.258 06	80	12.500 00	130	7.692 308	180	5.555 556
32	31.250 00	81	12.345 68	131	7.633 588	181	5.524 862
33	30.303 03	82	12.195 12	132	7.575 758	182	5.494 505
34	29.411 76	83	12.048 19	133	7.518 797	183	5.464 481
35	28.571 43	84	11.904 76	134	7.462 687	184	5.434 783
36	27.777 78	85	11.764 71	135	7.407 407	185	5.405 405
37	27.027 03	86	11.627 91	136	7.352 941	186	5.376 344
38	26.315 79	87	11.494 25	137	7.299 270	187	5.347 594
39	25.641 03	88	11.363 64	138	7.246 377	188	5.319 149
40	25.000 00	89	11.235 96	139	7.194 245	189	5.291 005
41	24.390 24	90	11.111 11	140	7.142 857	190	5.263 158
42	23.809 52	91	10.989 01	141	7.092 199	191	5.235 602
43	23.255 81	92	10.869 57	142	7.042 254	192	5.208 333
44	22.727 27	93	10.752 69	143	6.993 007	193	5.181 347
45	22.222 22	94	10.638 30	144	6.944 444	194	5.154 639
46	21.739 13	95	10.526 32	145	6.896 552	195	5.128 205
47	21.276 60	96	10.416 67	146	6.849 315	196	5.102 041
48	20.833 33	97	10.309 28	147	6.802 721	197	5.076 142
49	20.408 16	98	10.204 08	148	6.756 757	198	5.050 505
		99	10.101 01	149	6.711 409	199	5.025 126

Reproduced from "Handbook of Mathematical Tables and Formulas", by R. S. Burington, with permission of McGraw-Hill Book Co., Inc., © 1933, 1940 and 1948, pages 226, 227.

TABLE OF BINOMIAL COEFFICIENTS

n	$\binom{n}{0}$	$\binom{n}{1}$	$\binom{n}{2}$	$\binom{n}{3}$	$\binom{n}{4}$	$\binom{n}{5}$	$\binom{n}{6}$	$\binom{n}{7}$	$\binom{n}{8}$	$\binom{n}{9}$	$\binom{n}{10}$
0	1										
1	1	1									
2	1	2	1								
3	1	3	3	1							
4	1	4	6	4	1						
5	1	5	10	10	5	1					
6	1	6	15	20	15	6	1				
7	1	7	21	35	35	21	7	1			
8	1	8	28	56	70	56	28	8	1		
9	1	9	36	84	126	126	84	36	9	1	
10	1	10	45	120	210	252	210	120	45	10	1
11	1	11	55	165	330	462	462	330	165	55	11
12	1	12	66	220	495	792	924	792	495	220	66
13	1	13	78	286	715	1287	1716	1716	1287	715	286
14	1	14	91	364	1001	2002	3003	3432	3003	2002	1001
15	1	15	105	455	1365	3003	5005	6435	6435	5005	3003
16	1	16	120	560	1820	4368	8008	11440	12870	11440	8008
17	1	17	136	680	2380	6188	12376	19448	24310	24310	19448
18	1	18	153	816	3060	8568	18564	31824	43758	48620	43758
19	1	19	171	969	3876	11628	27132	50388	75582	92378	92378
20	1	20	190	1140	4845	15504	38760	77520	125970	167960	184756

Reproduced from "Handbook of Mathematical Tables and Formulas", by R. S. Burington, with permission of McGraw-Hill Book Co., Inc., © 1933, 1940 and 1948, page 256.

TABLE OF FACTORIALS AND THEIR RECIPROALS

n	n!	n	n!	n	1/n!	n	1/n!
1	1	11	39916800	1	1	11	.25052 × 10 ⁻⁷
2	2	12	479001600	2	0.5	12	.20877 × 10 ⁻⁸
3	6	13	6227020800	3	.16667	13	.16059 × 10 ⁻⁹
4	24	14	87178291200	4	.41667 × 10 ⁻¹	14	.11471 × 10 ⁻¹⁰
5	120	15	1307674368000	5	.83333 × 10 ⁻²	15	.76472 × 10 ⁻¹²
6	720	16	20922789888000	6	.13889 × 10 ⁻²	16	.47795 × 10 ⁻¹³
7	5040	17	355687428096000	7	.19841 × 10 ⁻³	17	.28115 × 10 ⁻¹⁴
8	40320	18	6402373705728000	8	.24802 × 10 ⁻⁴	18	.15619 × 10 ⁻¹⁵
9	362880	19	121645100408832000	9	.27557 × 10 ⁻⁵	19	.82206 × 10 ⁻¹⁷
10	3628800	20	2432902008176640000	10	.27557 × 10 ⁻⁶	20	.41103 × 10 ⁻¹⁸

Reproduced from "Handbook of Mathematical Tables and Formulas", by R. S. Burington, with permission of McGraw-Hill Book Co., Inc., © 1933, 1940 and 1948, page 255.

PRIME NUMBERS

TABLE OF PRIMES 1-10651

1	2	3	4	5	6	7	8	9	10	11	12	13
1	541	1223	1987	2741	3571	4409	5279	6133	6997	7919	8831	9733
2	47	29	93	49	81	21	81	43	7001	27	37	39
3	57	31	97	53	83	23	97	51	13	33	39	43
4	63	37	99	67	93	41	303	63	19	37	49	49
5	69	49	2003	77	607	47	09	73	27	49	61	67
6	11	71	59	11	89	13	51	23	37	39	51	63
7	13	77	17	91	17	57	33	99	43	63	67	81
8	17	87	27	97	23	63	47	203	57	93	87	87
9	19	93	83	29	801	31	81	51	11	69	8009	93
10	23	99	89	39	03	37	83	81	17	79	11	23
11	29	601	91	53	19	43	93	87	21	103	17	29
12	31	07	97	63	33	59	507	93	29	09	39	33
13	37	13	301	69	37	71	13	99	47	21	53	41
14	41	17	03	81	43	73	17	407	57	27	59	51
15	43	19	07	83	51	77	19	13	63	29	69	63
16	47	31	19	87	57	91	23	17	69	51	81	69
17	53	41	21	89	61	97	47	19	71	59	87	71
18	59	43	27	99	79	701	49	31	77	77	89	99
19	61	47	61	111	87	03	51	37	87	87	93	99
20	67	57	67	13	97	13	67	41	99	93	101	07
21	71	69	73	29	903	27	83	43	301	207	11	11
22	73	61	81	31	09	33	91	49	11	11	17	13
23	79	73	99	37	17	39	97	71	17	13	23	29
24	83	77	409	41	27	61	603	77	23	19	47	41
25	89	83	23	43	39	67	21	79	29	29	61	43
26	97	91	27	53	53	69	37	83	37	37	67	49
27	101	701	29	61	57	79	39	501	43	43	71	59
28	03	09	33	79	63	93	43	03	53	47	79	67
29	07	19	39	203	69	97	49	07	59	53	91	67
30	09	27	47	07	71	803	51	19	61	63	209	103
31	13	33	51	13	99	21	57	21	67	97	19	09
32	17	39	53	31	3001	23	63	27	73	307	21	27
33	21	43	59	37	11	33	73	31	79	09	31	33
34	37	51	71	39	19	47	79	57	89	21	33	37
35	39	57	81	43	23	51	91	63	97	31	37	39
36	49	61	83	51	37	53	703	69	421	33	43	57
37	51	69	87	67	41	63	21	73	27	49	63	61
38	57	73	89	69	49	77	23	81	49	51	69	73
39	63	87	93	73	61	81	29	91	51	69	73	81
40	67	97	99	81	67	89	33	623	69	93	87	87
41	73	809	511	87	79	907	51	39	73	41	91	99
42	79	11	23	93	83	11	59	41	81	17	93	201
43	81	21	31	97	89	17	83	17	93	37	97	11
44	91	23	43	309	109	19	87	51	521	51	311	21
45	93	27	49	11	19	23	89	53	29	57	17	27
46	97	29	53	33	21	29	93	57	47	59	29	39
47	99	39	59	39	37	31	99	59	51	77	53	41
48	211	53	67	41	63	43	801	69	53	81	63	57
49	23	57	71	47	67	47	13	83	63	87	69	77
50	27	59	79	51	69	67	17	89	69	89	77	81
51	29	63	83	57	81	89	31	93	71	99	87	83
52	33	77	97	71	87	4001	61	701	77	507	89	93
53	39	81	601	77	91	03	71	11	81	17	419	311
54	41	83	07	81	203	07	77	17	99	23	23	19
55	51	87	09	83	09	13	89	37	607	29	29	23
56	57	907	13	89	17	19	903	41	19	37	97	09
57	11	93	19	93	23	21	09	43	37	41	31	47
58	69	19	21	99	29	27	19	49	53	47	47	43
59	71	29	27	411	51	49	31	79	59	49	61	49
60	77	37	37	17	53	51	33	83	61	59	67	71
61	81	41	57	23	57	57	37	91	73	61	501	77
62	83	47	63	37	59	73	43	801	79	73	13	91
63	93	53	67	41	71	79	51	07	89	77	21	89
64	307	67	69	47	99	91	57	13	91	83	27	403
65	11	71	93	59	301	93	67	21	701	89	37	13
66	13	77	97	67	07	99	69	27	03	91	39	19
67	17	83	99	73	13	111	73	39	09	603	49	21
68	31	91	709	77	19	27	87	43	19	07	63	31
69	37	97	21	503	23	29	93	49	33	21	73	33
70	47	1009	23	21	29	33	99	51	37	39	81	37
71	49	13	33	31	31	39	5003	57	61	43	97	39
72	53	19	41	39	43	53	09	61	63	49	99	61
73	59	21	47	43	47	57	11	67	79	69	609	63
74	67	31	53	49	59	59	21	69	81	73	23	67
75	73	33	59	51	61	77	23	79	91	81	27	73
76	79	39	77	57	71	201	39	81	93	87	29	79
77	83	49	83	79	73	11	51	97	803	91	41	91
78	89	51	87	91	89	17	59	903	23	99	47	97
79	97	61	89	93	91	19	77	23	27	703	63	511
80	401	63	801	609	407	29	81	27	29	17	69	21
81	09	11	17	13	31	87	39	33	23	77	33	59
82	19	87	23	21	33	41	99	53	41	27	81	39
83	21	91	31	33	49	43	101	81	57	41	89	47
84	31	93	47	47	57	53	07	87	63	53	93	51
85	33	97	61	57	61	59	13	6007	69	57	99	87
86	39	103	67	59	63	61	19	11	71	59	707	601
87	43	09	71	63	67	71	47	29	83	89	13	13
88	49	17	73	71	69	73	53	37	99	93	19	19
89	57	23	77	77	91	83	67	43	907	817	31	23
90	61	29	79	83	99	89	71	47	11	23	37	29
91	63	51	89	87	511	97	79	53	17	29	41	31
92	67	53	901	89	17	327	89	67	47	41	47	43
93	79	63	07	93	27	37	97	73	49	53	53	49
94	87	71	13	99	29	209	79	59	67	61	61	601
95	91	81	31	707	33	49	27	89	61	73	79	07
96	99	87	33	11	39	57	31	91	67	77	83	79
97	503	93	49	13	41	63	33	101	71	79	803	89
98	09	201	51	19	47	73	37	13	77	83	07	97
99	21	13	73	29	57	91	61	21	83	901	19	719
100	23	17	79	31	59	97	73	31	91	07	21	21

TABLE OF SPECIAL PRIMES
(PRECEDING 20000, 30000, 40000, ...)

IMMEDIATELY PRECEDING:

20000: 19979

91

93

97

30000: 29947

59

83

89

40000: 39971

79

83

89

50000: 49957

91

93

99

60000: 59957

71

81

99

70000: 69941

59

91

97

80000: 79979

87

97

99

90000: 89963

77

83

89

100000: 99961

71

89

91

200000: 199933

61

67

99

Reproduced from "List of Prime Numbers", by D. N. Lehmer, with permission of Hafner Publishing Co., ©1956, p. 1.

TABLE OF SQUARES, CUBES, SQUARE ROOTS

Table with columns: n, n^2, n^3, sqrt(n), sqrt(10n) for n values from 1 to 99.

The Tables on this page and the next are Reproduced from "Handbook of Mathematical Tables and Formulas", by R. S. Burington with permission of McGraw-Hill Book Co., Inc., © 1933, 1940 and 1948, pages 206-215.

TABLE OF POWERS OF TWO

2^n	n	2^{-n}
1	0	1.0
2	1	0.5
4	2	0.25
8	3	0.125
16	4	0.062 5
32	5	0.031 25
64	6	0.015 625
128	7	0.007 812 5
256	8	0.003 906 25
512	9	0.001 953 125
1 024	10	0.000 976 562 5
2 048	11	0.000 488 281 25
4 096	12	0.000 244 140 625
8 192	13	0.000 122 070 312 5
16 384	14	0.000 061 035 156 25
32 768	15	0.000 030 517 578 125
65 536	16	0.000 015 258 789 062 5
131 072	17	0.000 007 629 394 531 25
262 144	18	0.000 003 814 697 265 625
524 288	19	0.000 001 907 348 632 812 5
1 048 576	20	0.000 000 953 674 316 406 25
2 097 152	21	0.000 000 476 837 158 203 125
4 194 304	22	0.000 000 238 418 579 101 562 5
8 388 608	23	0.000 000 119 209 289 550 781 25
16 777 216	24	0.000 000 059 604 644 775 390 625
33 554 432	25	0.000 000 029 802 322 387 695 312 5
67 108 864	26	0.000 000 014 901 161 193 847 656 25
134 217 728	27	0.000 000 007 450 580 596 923 828 125
268 435 456	28	0.000 000 003 725 290 298 461 914 062 5
536 870 912	29	0.000 000 001 862 645 149 230 957 031 25
1 073 741 824	30	0.000 000 000 931 322 574 615 478 515 625
2 147 483 648	31	0.000 000 000 465 661 287 307 739 257 812 5
4 294 967 296	32	0.000 000 000 232 830 643 653 869 628 906 25
8 589 934 592	33	0.000 000 000 116 415 321 826 934 814 453 125
17 179 869 184	34	0.000 000 000 058 207 660 913 467 407 226 562 5
34 359 738 368	35	0.000 000 000 029 103 830 456 733 703 613 281 25
68 719 476 736	36	0.000 000 000 014 551 915 228 366 851 806 640 625
137 438 953 472	37	0.000 000 000 007 275 957 614 183 425 903 320 312 5
274 877 906 944	38	0.000 000 000 003 637 978 807 091 712 951 660 156 25
549 755 813 888	39	0.000 000 000 001 818 989 403 545 856 475 830 078 125

DIGITAL COMPUTER ORGANIZATION

Since systems engineers should have a fundamental reference to the principles of digital computer operation, we shall discuss the subject from a generalized point of view. Additional general information will be found in the references given at the end of the paper, while specific computer information will be found in the appropriate IBM manuals.

THE CENTRAL PROCESSING UNIT

Storage

Storage elements allow the acceptance, holding, and removal of information in a computer. This information is usually made up of instructions, constants, data to be processed and reference data associated with a particular program.

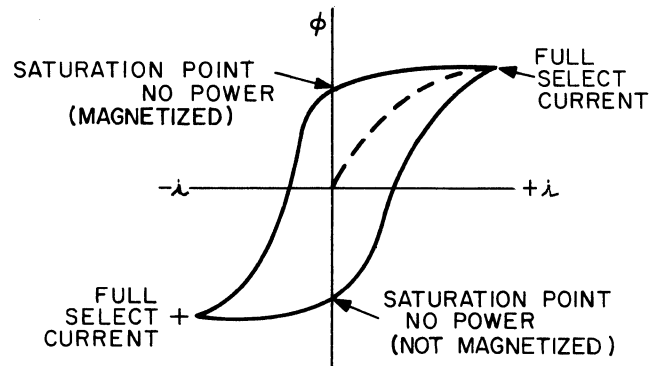
There are several different types of storage elements. The most commonly used are magnetic cores and magnetic drums.

Magnetic Core Storage

Cores are tiny rings of ferromagnetic material. They are used as storage elements because a single core can be permanently magnetized in a few microseconds, as follows:

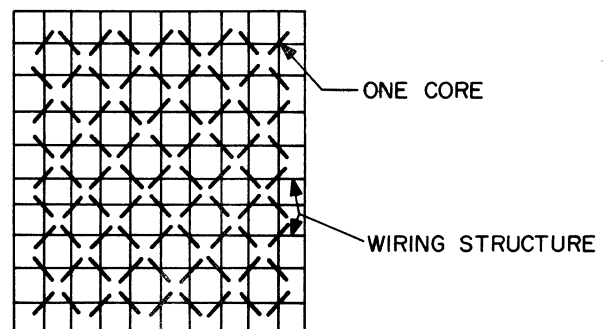
An electric current on one or more wires passing through the core will induce (create) magnetism in the core. The magnetic field of the core will vary with the strength of the current up to a certain limit. When this limit is reached, the core is said to be saturated. The polarity (direction) of the magnetic field induced in the core is determined by the direction of the current in the wire(s). When the core has been saturated, the current may be turned off, and the core will remain magnetized at near-saturation indefinitely or until a current is passed through the core in the opposite direction.

This property of retention of the magnetic effect in the core is known as hysteresis. It may be described quantitatively by means of a graph, as shown below, in which magnetic field strength ϕ is plotted against magnetizing current i for a typical core. The graph is called a hysteresis loop, and its exact shape will depend upon the specific core material being considered.

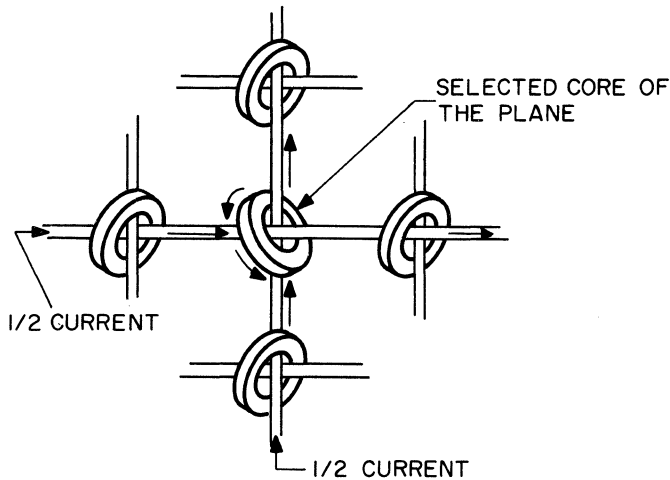


In addition to the wire(s) which transmit pulses of electrical current through the core, another wire which does not carry an imposed current is passed through the core. This is known as the sense wire. When the current-carrying wires transmit a pulse which changes the magnetic polarity of the core, an electrical current pulse will be induced in the sense wire. The direction of this induced pulse is associated with the direction of change of the magnetic polarity in the core. Thus, the sense wire pulses effectively transmit the pattern of magnetic polarities in a magnetic core storage to any part of the computer to which these patterns are to be removed from storage. To sum up, readout from core is destructive (it causes all cores being read to take on the same polarity), and provision must be made, through features not discussed here, to restore or regenerate the original pattern of polarities immediately after each readout, in order to preserve the information for possible future use.

In actual practice the configuration of cores and wires used in storage devices, known as a core plane, is that shown here.



The following diagram shows how the desired core alone in a plane can be selected to have new information read into it (i. e. , to have its polarity changed).



Information in core storage is represented by the polarity pattern of a group of cores. The IBM 7090 core storage is organized in groups of 36 cores. Each group, commonly called a word, provides 2^{36} polarity patterns. The IBM 1401 core storage is organized in groups of eight bits. Each group, commonly referred to as a position of storage, provides six information bits or 64 polarity patterns.

The data representation area of this handbook will provide additional information about the use of core storage to represent information.

Magnetic Drum Storage

This type is used as the main storage of the computer or as an input/output storage medium. The principles of operation of the magnetic drum are discussed in the input/output portion of this area.

Addressing the Storage Element

Computer storage is referred to in terms of storage words or positions of storage. Each word or position of storage is identified by a numerical, alphabetic or alphanumeric address.

A computer instruction will tell the computer what to do (the operation code) and what to do it to (the data address). Some computers also allow other uses of the data address or provide more than one data address.

Uses of the Data Address

1. Immediate address: The address portion is used as actual data, rather than to specify the location in storage of the information required.

2. Direct address: The address portion is used to specify the location of the information. This is the most used type of address.

3. Indirect address: The address portion is used as the address of a location which in turn gives the address of the information.

4. Indexed address: Index registers, when supplied in computers, hold quantities which may be algebraically added to the address portions of instructions as they are executed. The address so created is called an effective address, and is utilized as if it had been the direct address given in the instruction in the first place. The actual address given in the instruction remains unchanged by this indexing procedure. The determination whether to so modify an actual address by applying the contents of an index register is made by specifying the number of the index register (there are generally more than one) in a special portion of the instruction to be indexed.

Example:

Operation	Index	Address
ADD	03	1 4 0 0 0

An index portion of 00 might indicate no indexing to be applied.

In general, machines having index registers will also have special instructions for manipulating and testing their contents to facilitate the use of instruction modification.

5. Relative address: The address portion is used to achieve program relocation. All routines are coded relative to location zero, and their addresses are modified by a base address as they are loaded into the machine.

Three additional uses of addresses, listed below, are sometimes referred to as relative addressing. However, the first two items impinge on the area of indexing, and the third involves assembly program functions.

- a. The address portion of the instruction is automatically added to some prearranged base address (that of the start of a table, say) to produce an effective address to be accessed.

- b. The address portion is used as an increment to the current setting of the instruction counter (a machine register) to provide an effective address.
- c. The address portion represents an increment for locating subfields, relative to a control field or key.

6. Implied address: An address, usually of some specialized register that is "understood" by the operation specified in the instruction. Thus, in an "add" instruction, the accumulator may be implied as the address of one of the operands.

Single- and Multiple-Address Instructions

1. Single-address instructions

Example: ADD B
 (Accumulator + B → Accumulator)

Characteristics:

- More frequent use of implied addresses.
- Short instructions.
- More instructions required for a given job.
- Special registers or memory areas, such as accumulators, required.

2. Two-address instructions

Example: MOVE A, B
 (A → B)

Example: MULTIPLY A, B
 (A x B → B)

Characteristics:

- Frequent use of implied addresses.
- Fewer instructions needed for a given job than with single-address format.
- Functionally similar to three-address instructions.
- The second address in general wasted in unconditional branch instructions.
- Accumulators not required.

3. Three-address instructions

Example: ADD A, B, C
 (A + B → C)

Characteristics:

- Long instruction format.
- Accumulators not required.
- All three addresses not required by many operation codes.

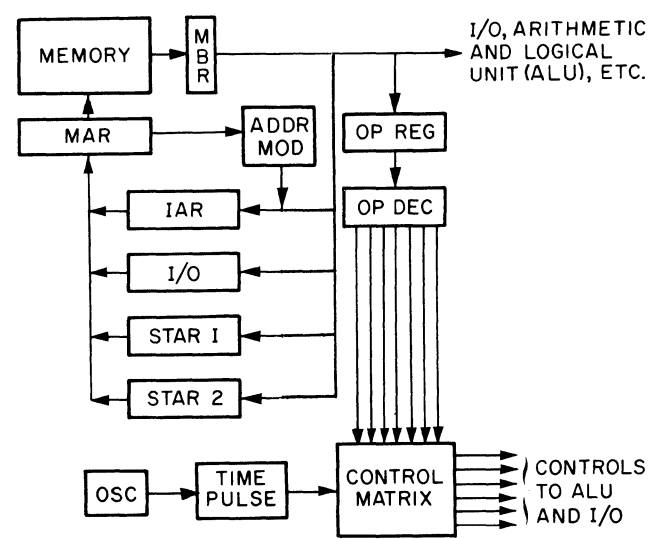
4. Variable address instructions: The use of more than one of the above formats in a single computer. This approach can be especially significant in machines with variable field (hence instruction) lengths to provide optimum instruction sets. The IBM 1400 series serves as an example.

Control

Control refers to the execution of instructions. It is carried out by the generation of appropriate signal pulses to govern (1) instruction fetch from storage; (2) instruction execution itself.

In a synchronous control design, all control functions take place in fixed time intervals. In an asynchronous control, conditions arising during execution itself determine when subsequent signal pulses will be generated to continue execution, initiate next instruction fetch, etc.

Instruction Fetch



The above diagram is a simplified schematic of the typical internal paths used to fetch an instruction from storage. This is done during the first basic machine cycle (governed by pulses from an electronic oscillator), commonly called an I (instruction) cycle, or I time. The general sequence of I time events is:

1. Send location of next instruction, assumed to be already in IAR (instruction address register), to MAR (memory address register).

2. Use MAR contents to start storage readout from correct location, and at same time send MAR contents through the address modifier, where it is incremented by 1. Send resulting incremented address on to IAR.

3. Storage readout cycle ends with arrival of first character of instruction in MBR (memory buffer register). (We assume serial-by-character readout.) Assuming first character is operation code, send this to OP REG (operation register).

4. Repeat steps 1 and 2 to obtain each character of the one or more addresses of the instruction, sending these to STAR 1 (storage address register), STAR 2, or I/O (input/output unit address register), as required. This is determined by the OP DEC (operation decoder) having determined what type of address(es) will be required.

5. When last character goes to STAR 1, etc., begin E (execution) time by a signal to the execution controls.

Instruction Execution

Continuing to refer to the diagram, the events of execution are:

1. OP DEC sends a single signal over one of the lines to the control matrix, thus specifying on the basis of the operation code the operation to be carried out.

2. The control matrix, in cycles "stepped off" by pulses from osc (oscillator), sends time pulse signals over the specific control lines to ALU (arithmetic and logical unit), I/O, etc., to control the steps of the machine.

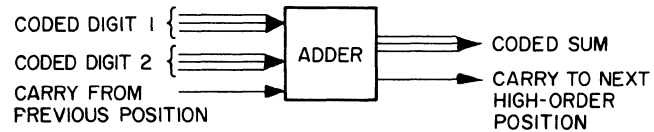
3. The control matrix also sends signals to choose required data from memory and send them to ALU, in proper sequence, as well as to send results back from ALU to memory.

4. At end of execution, the control matrix generates a pulse to signal the next instruction fetch.

Arithmetic

In most computers arithmetic is performed by a circuit device known as an adder and additional control circuits and modifications to the adder. This is based upon the fact that all arithmetic operations can be reduced to sequences of additions together with appropriate control functions. In its basic form an adder accepts as inputs two

digits to be summed and the carry from the previous position (e. g., units, tens, hundreds, etc.). The output is a sum and a carry to the next-higher position. The scheme is:



Subtraction and addition of numbers having unlike signs are accomplished by complementation.

The 9s complement of a number is formed by taking the 9s complement of each individual digit (difference between 9 and the digit).

0	0	3	1	8
---	---	---	---	---

 number

9	9	6	8	1
---	---	---	---	---

 9s complement

The 10s complement of a number is 1 more than its 9s complement. (In binary machines the 1s and 2s complements are used, and are similarly defined.)

The registers which hold the values being handled in arithmetic operations may be special parts of the arithmetic unit itself (e. g., IBM 650, 7070, 7090), or they may be either specially designated or arbitrary locations in memory (IBM 1620, 1400 series).

Addition and Subtraction

Subtraction is in general reduced to addition by complementation of the quantity to be subtracted (subtrahend). Addition (whether arising in the course of subtraction or not) will either involve numbers having like signs or numbers having unlike signs. When the signs are alike the addition involves a straightforward use of the adder, as introduced above, followed by attaching the common sign of the numbers to their sum. When the numbers have unlike signs, some one of the following procedures, or a variation, may be employed:

10s complement method. — Replace the negative number by its 10s complement and add to the positive number. If a carry occurs out of the most significant position in developing the sum, ignore it and make the sign of the sum positive. If no such carry occurs, take the 10s complement of the sum and make its sign negative.

9s complement method. — Replace the negative number by its 9s complement and add to the positive number. If a carry occurs out of the most significant position in developing the sum, add this unit carry to the units position of the sum and make its sign positive. If no such carry occurs, take the 9s complement of the sum and make its sign negative.

The above procedures assume a serial addition process -- that is, processing one digit position through the adder at a time, from low- to high-order positions.

Some computers (e. g., IBM 7090/94, 7072/74) employ parallel adders. Although the basic logic of complementation generally follows one of the above methods, the parallel scheme allows adding all positions in one simultaneous operation, followed by propagation of carries, as required.

A further variation is that of addition by table lookup, referring to an addition table stored in memory. The IBM 1620 utilizes this scheme, which has certain advantages in economy, as well as in allowing flexibility in performing arithmetic on varying number bases, depending upon what form of table is stored. This system is described in the 1620 reference manual (A26-4500).

Floating point arithmetic operations, available as either optional or standard features on many computer systems, are, like the fixed point operations discussed here, generally based on the addition operation.

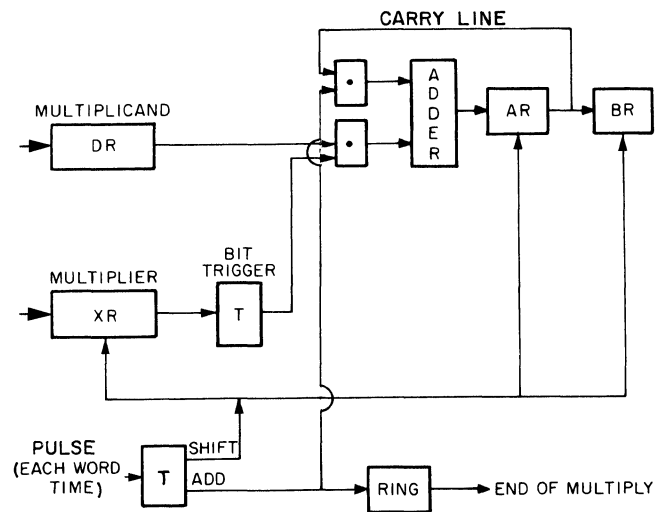
Multiplication

In most machines multiplication closely parallels the paper-and-pencil method. That is, each successive multiplier digit determines, by its value, the number of times the multiplicand will be added to a growing sum of partial products. Multiplier digits may be handled from right to left or from left to right. In the former case, each partial product is shifted one position to the left before it is added; in the latter case the shifts are to the right.

The following figure illustrates the multiplication logic for a 4-bit word length system. The binary scheme is chosen to facilitate explanation; extension to a decimal system is simply made.

Steps in the operation are:

1. Send multiplicand to register DR, multiplier to register XR. (All registers are 4 bits in length.)
2. Shift bits in XR to the right, least significant bit going to the Bit Trigger.



Bullet represents logical "AND" circuit; T represents trigger.

3. If Bit Trigger contains 1 as result of step 2, send multiplicand through adder, developing partial product in AR.

4. If Bit Trigger contains zero, do not send multiplicand through adder.

5. Shift sum in AR one place to the right (using register BR). (The effect is to shift next following partial product one place to the left.) At the same time, shift multiplier in XR one place to the right to move bit to Bit Trigger.

6. With each shift and add sequence, step the Ring by 1. When the Ring registers 4, it produces an end of multiply pulse. The completed product in AR and BR is 8 bits long, illustrating that, in general, the length of a product equals the sum of the lengths of the multiplier and the multiplicand.

The table lookup method mentioned under addition and subtraction is also applicable to multiplication. Again, the IBM 1620 exemplifies this approach, and its literature is a source of further information.

Division

Since circuit logic to control division may add substantially to the cost of an arithmetic unit, the divide operation is optional on some machines. The IBM 1401 and 1620 are examples (multiply is also optional on the 1401).

The following expression serves to define division terminology as used here:

$$\frac{\text{Dividend}}{\text{Divisor}} = \text{Quotient} + \frac{\text{Remainder}}{\text{Divisor}}$$

A basic problem in most computer arithmetic units is to assure that the number of digits developed for the quotient will not overflow the register

or memory area set aside for it. Typically, such overflow is prevented by specifying some such rule as that, in any division, the divisor, taken as an integer, must exceed in magnitude the magnitude of a high-order portion of the dividend which has the same number of digits as allowed for the entire divisor. The high-order dividend portion is also to be regarded as an integer.

Example: In a decimal machine which allows a 20-digit dividend and a 10-digit divisor and quotient, to prevent quotient overflow the magnitude of the divisor must exceed the magnitude of the number formed by the "upper" 10 digits of the dividend.

It will be seen that such a rule includes a proscription against division by zero.

In general, computer division is performed through a repetitive algorithm which develops quotient digits one after another, starting with the high-order position. The algorithm typically involves counting subtractions of the divisor from the dividend. Specific common procedures are:

Restoring method. — The steps in this method are:

1. Subtract divisor from high-order dividend portion and add 1 to high-order quotient digit.
2. Test for change in sign of the remaining partial dividend.
3. Repeat above until partial dividend changes sign.
4. Upon sign change, add divisor to partial dividend, thus restoring the original sign, and subtract 1 from high-order quotient digit.
5. Shift dividend one position to the left, or divisor right.
6. Repeat subtraction process, using next lower-order quotient position to accumulate 1s. Continue this entire process until all quotient digits are developed.
7. At conclusion the partial dividend has become the remainder.

In a decimal machine, this method requires an average of 6.5 add or subtract cycles per quotient digit developed, assuming random quotient digit distribution.

Non-restoring method. — This is a variation on the restoring method designed to save time at the expense of additional control circuitry. The procedure starts, as above, by subtracting divisor from dividend, and adding 1s to quotient. When dividend sign changes, no restoration is made. Instead, shifting is performed directly; then the divisor is added to the partial dividend and 1s

subtracted from the next quotient position.

These alternations in the procedure are maintained at each shift operation.

The result of this version of division is to require an average of 5.5 add or subtract cycles per (randomly distributed) quotient digit, as opposed to the 6.5 in the restoring form of the algorithm.

PERIPHERAL COMPONENTS

Peripheral components of the computing system are the means by which information enters and leaves the system. These means allow the use of punched cards, paper documents, paper tape, manual keyboard, and magnetic tape within the system. We shall discuss here only the principal peripheral components.

Devices

Magnetic Tape

Magnetic tape represents an extreme in that it offers storage for much information at a penalty of great access times. It is a serial storage medium, in the sense that, to locate a piece of information, all of a reel of tape must pass through the tape unit, from the beginning to the information in question. As a result, tape is primarily used for input/output: reading transactions to be processed, writing invoices, etc. Accordingly, the use of tape for auxiliary storage (a subroutine library tape, etc.) is secondary.

The tape unit contains read-write heads (as with drums and disks) and mechanisms for passing the tape by these heads. Information is recorded essentially as magnetized spots in parallel tracks along the length of the tape.

IBM tape systems use seven tracks (729 series of tape units), or ten tracks (Hypertape). They pass tape at 36 inches per second (7330 tape units), 75 or 112.5 inches per second (729 series), 112.5 or 22.5 inches per second (Hypertape), and record information at 200, 556, 800, or 1,500 (Hypertape) bits per inch in each track.

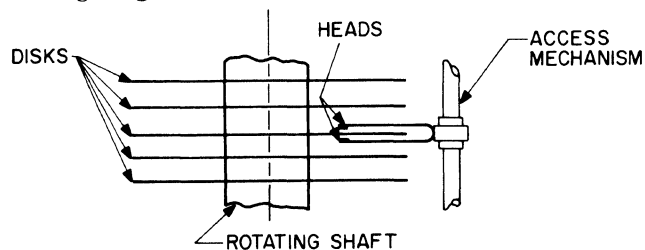
The various IBM manuals on tape units will be helpful for further information, as will the material in this handbook on magnetic tape units.

Magnetic Disk Storage

Magnetic disks vary in diameter, the largest being about two feet. Their two surfaces are coated with ferrous oxide material. A number of

disks are mounted on a rotating vertical shaft; the IBM 1301 disk storage, for example, rotates at 1790 rpm.

Data is written, stored, and read from the disk surfaces in the same manner as with magnetic drums (see below). The tracks are in the form of concentric rings on the disk surfaces. Read-write heads are mounted on access arms which move to any track, under computer control. Usually, an arm contains two heads, one for each surface of a disk. This is shown in the following diagram.



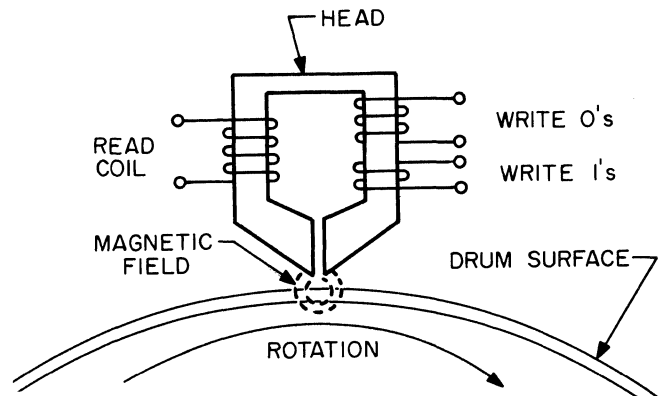
Not shown here is the fact that a disk file may be equipped with multiple arms, one for each disk. The IBM 1301 and 1311 Disk Storage units are so equipped.

Time to access disk information depends upon location of the information on the disk, disk motion speed, speed of arm motion across the disk surface, number of arms, and, for disk units not equipped with multiple arms, speed of arm motion between disks.

Magnetic Drum Storage

The cobalt-nickel alloy surface of a revolving cylinder forms magnetic drum storage. Small areas or cells of this surface, of the order of magnitude of .07 inch in length, can be permanently magnetized by passing pulses of current through magnetizing coils. The polarity of magnetization --- hence interpretation as a stored "zero" or "one" --- will depend upon the direction of current in the magnetizing coil, called a write coil.

For removing information from the drum, (that is, readout), a read coil is furnished. This will have an electrical pulse induced in it as the magnetic cell on the drum surface passes beneath it. The direction of the pulse will depend upon the magnetic polarity of the cell, allowing distinction between "zero" and "one" bits. The read and write coils are arranged on a read-write head, a typical arrangement being shown in the diagram below. In general, cells are arranged in tracks around the drum. Each track is equipped with its own read-write head.



Total waiting time to read or write information in a specific cell falls into two parts: (1) latent time (time for the cell to reach the read-write head as the drum rotates) and (2) access time (time for the cell to pass under the read-write head during actual reading or writing). An objective in programming drum-equipped machines is to reduce latent time as much as possible. This can be done through such techniques as optimal programming (e. g. , IBM 650) or such devices as revolving loops or multiple read-write heads.

Logical Input-Output Methods

The fundamental problem in handling input/output is to reconcile the speeds of central processing units with those of I/O devices. Obtaining one record (amount of information read or written in one operation) through an input device, say, a card reader, may require many milliseconds. Moving, adding, etc. , this record within the central processing unit (CPU) may be accomplished in a few microseconds. The following approaches to I/O organization are presented in the general order of the increasing efficiency with which they reconcile these disparate speeds. They should be considered in the light of the possibility or impossibility of overlapping the I/O functions with the processing in the CPU.

1. Direct to storage: no overlapping

The simplest I/O organization is that in which:

- Only one path exists from I/O device to storage.
- One instruction starts the I/O operation; it continues until signaled to stop by some end-of-record indication.
- No CPU processing can occur during the time of the data transfer between I/O device and CPU.

The IBM 1401 illustrates this scheme.

2. Direct to storage with copy logic

- Only one path exists from I/O device to storage.

- A separate instruction - "Copy" or "Transmit" - must be given for the transfer of each unit of data (character or word) between storage and the I/O device.
- Between the copy instructions, if certain strict timing rules are followed, it may be possible to program some CPU processing. This follows from the fact that the CPU will otherwise spend much time waiting for the slower I/O device.

The IBM 704 and the IBM 1401 with Serial I/O Device follow this approach.

3. Input and output (but not processing) at the same time.

- Two separate paths, each with an address register to control storage location references, are available.
- By appropriate programming, both input and output are started by a single instruction, or by two instructions "close together" in the program. The I/O operations, once begun, proceed without further instructions until both are complete.
- No CPU operations can occur until the longer of the input and output operations is complete.

The IBM 705 Read-While-Write operation illustrates the above. So also does the IBM 1401, using certain features under certain conditions.

4. Processing and the mechanical portion of input or output overlapped; buffer storage.

- A separate buffer storage unit, large enough to hold entire I/O records, interrupts the path(s) between I/O and storage.
- While the buffer is being filled from or emptied into the I/O device, processing continues in the CPU.
- Saves most of the time of mechanical operation of the I/O unit. Problem running time approaches processing time plus the rapid buffer-main storage transfer time.
- May be combined with a Read-While-Write feature (see 3, above) involving the buffer storage and two I/O devices.

The IBM 705 with tape record coordinator and the IBM 1401 with print storage exemplify this approach.

5. Input-Output-Compute overlap: Data channels.

- I/O devices are connected to a separate data channel unit containing control functions and registers to allow holding

a word in transit between I/O unit and storage and to keep track of the main storage locations involved.

- The channel is started by the CPU program and automatically handles the input or output of the designated main storage area, keeping the operation synchronized to I/O device speed. In general, only a single connected block of main storage information can be handled by the channel per reference to it by the program.
- The program continues processing during the intervals when there is no memory interference by the channel.
- As many individual I/O activities can thus be overlapped with processing as there are channels on the machine system.

The IBM 1410 illustrates this scheme. Its channels may also be used in a nonoverlap (with processing) mode.

6. Input-Output-Compute overlap through separately programmed "autonomous" channels.

- The channels are as discussed under item 5 above, in addition to which they can independently execute special I/O program steps which they retrieve automatically and sequentially from main storage.
- Channel execution of an I/O routine is initiated by instructions in the main program.
- In addition to the capabilities of data channels as set forth in item 5 above, autonomous channels allow:
 - (1) Reading from one input record to scattered areas of memory during one channel I/O routine execution.
 - (2) Writing an output record by gathering information from many memory areas during one I/O routine execution.
 - (3) Skipping unwanted portions of an input record during one I/O routine execution.
- Optionally, channels may be associated with an interrupt feature, also known as priority processing. This allows a channel, when certain conditions arise such as completing an I/O activity, immediately to force the CPU program to branch to a predetermined location, where a routine appropriate to the situation will be stored.

The IBM 7070 and 7090 systems employ autonomous data channels.

BASIC COMPUTER ELEMENTS AND TERMS

Short definitions of the basic technical terms involving computer elements, which the systems engineer will encounter as he consults more detailed references, are listed here. Since the list is brief, the items are given in logical rather than alphabetic order.

Triggers, latches, flip-flops. — Various constructed two-state devices which, when a pulse representing a "zero" or a "one" bit is received, continually send out a corresponding "zero" or "one" signal, even when the original pulse has finished, until a new input pulse is received. Thus these devices store and remember individual bits. Their basic principal of operation is that of feedback (using their own output to determine their continuing state).

Registers. — Circuits comprising triggers (see above) so arranged as to allow receiving, storing and reading out entire patterns of code, representing characters or words of information. When associated with appropriate control circuits, registers take on the specialized functions of accumulators, multiplier-quotient units, storage registers, address registers, and so on.

Gates. — Switching circuits which respond to the presence of "zero" or "one" input signals by opening or closing, to allow or prevent the flow of information signals. When associated with registers, gates are used to allow input to or readout from the registers.

Data paths. — The lines over which data signals flow, as permitted by the gates in the paths.

Paths are of two basic types:

1. Serial: Data is transferred bit after bit.
2. Parallel: All bits within a character or word are transferred simultaneously, there being enough separate transmission lines to accomplish this.

(See also serial and parallel addition, in the preceding portion of this paper entitled "Arithmetic".)

Synchronous and asynchronous operation. — Refers in general to CPU instruction executions. Synchronous operations (for example, an addition or a compare) are allotted prescribed time intervals determined by counting clock pulses (see "clocks", below) at the end of which the next operation is begun. Asynchronous operations occupy varying

time intervals, depending upon the requirements of the case at hand (length of fields, etc.), with the end of the operation generating a special signal to allow the next operation to begin. (See also under the preceding portion entitled "Control".)

Clocks. — Devices which generate regularly spaced timing pulses, used throughout the CPU to control operations. This assures that all impulses reach the proper destinations in the correct sequence, since the timing pulses govern all gating and signal emission.

Rings. — Interconnected sets of triggers which serve as counters. The function of a ring is to control the number of steps required in an operation. (See also the example of ring usage in a binary multiplier circuit in the preceding discussion under "Arithmetic".)

Decoders. — Devices which translate certain input signal configurations into modified output signal configurations. For instance, an operation decoder would accept the bit configuration of an operation code, and emit, as a result, the required pattern of signals to control the logical circuits which execute the intended operation. (See also the preceding discussion under "Control".)

BIBLIOGRAPHY

This list is a representative selection from the available texts on computer design and organization.

Bartee, Thomas C., Digital Computer Fundamentals, McGraw-Hill, 1960.

Buchholz, Werner, ed., Planning a Computer System, McGraw-Hill, 1962. A collection of papers on the design and implementation of STRETCH by those involved.

Flores, Ivan, Computer Logic, Prentice-Hall, 1960. Progresses from a general to a detailed coverage. Written for "those of some scientific training".

Irwin, Wayne C., Digital Computer Principles, D. Van Nostrand, 1960. A simple treatment for those without previous acquaintance with computers, electronics or mathematics.

Richards, R. K., Arithmetic Operations in Digital Computers, D. Van Nostrand, 1955. A comprehensive and respected work, stressing logical and symbolic representation rather than actual components and circuits.

Section	Area	Page
09	01	10

Richards, R. K. , Digital Computer Components and Circuits, D. Van Nostrand, 1957. A companion to Arithmetic Operations in Digital Computers, stressing realization of the logical and

symbolic concepts of that work in terms of the engineering aspects of a working machine.

Siegal, Paul, Understanding Digital Computers, John Wiley and Sons, 1961.

IBM SYSTEMS CONFIGURATORS

The inherent modular construction of IBM systems permits a choice in integration of a variety of functional components and features tailored to accomplish specifically defined customer data processing requirements.

Similarly, flexible modularity of system configuration facilitates subsequent system changes and capacity expansion when needed to keep pace with dynamic processing requirements. The detailed specification of all the necessary modules of each individual system, then, is a compatible selection from a complex assortment of various available components, control units and special features. (A 1410 system, for example, may be made up of as few as two or three components or as many as 50 or more components and features, depending upon need.)

To aid selection and specification a configurator is available for major IBM systems. A configurator is a chart which designates every possible component choice by number and model and, by family tree arrangement, specifies the control units and/or special features either required or optionally available to support the selected component. It also gives

the maximum number of each component which can operate on a single system, and specifies those cases where functionally similar components may be alternately selected.

The published system configurators (available from Stationery Stores) are kept up to date by the respective manufacturing divisions to reflect available system configurations. Note that while the configurators are the most current published, they may not necessarily reflect the most recent product announcements. The "DP Sales Manual" pages should be used to finally verify configurations which include newly announced equipment.

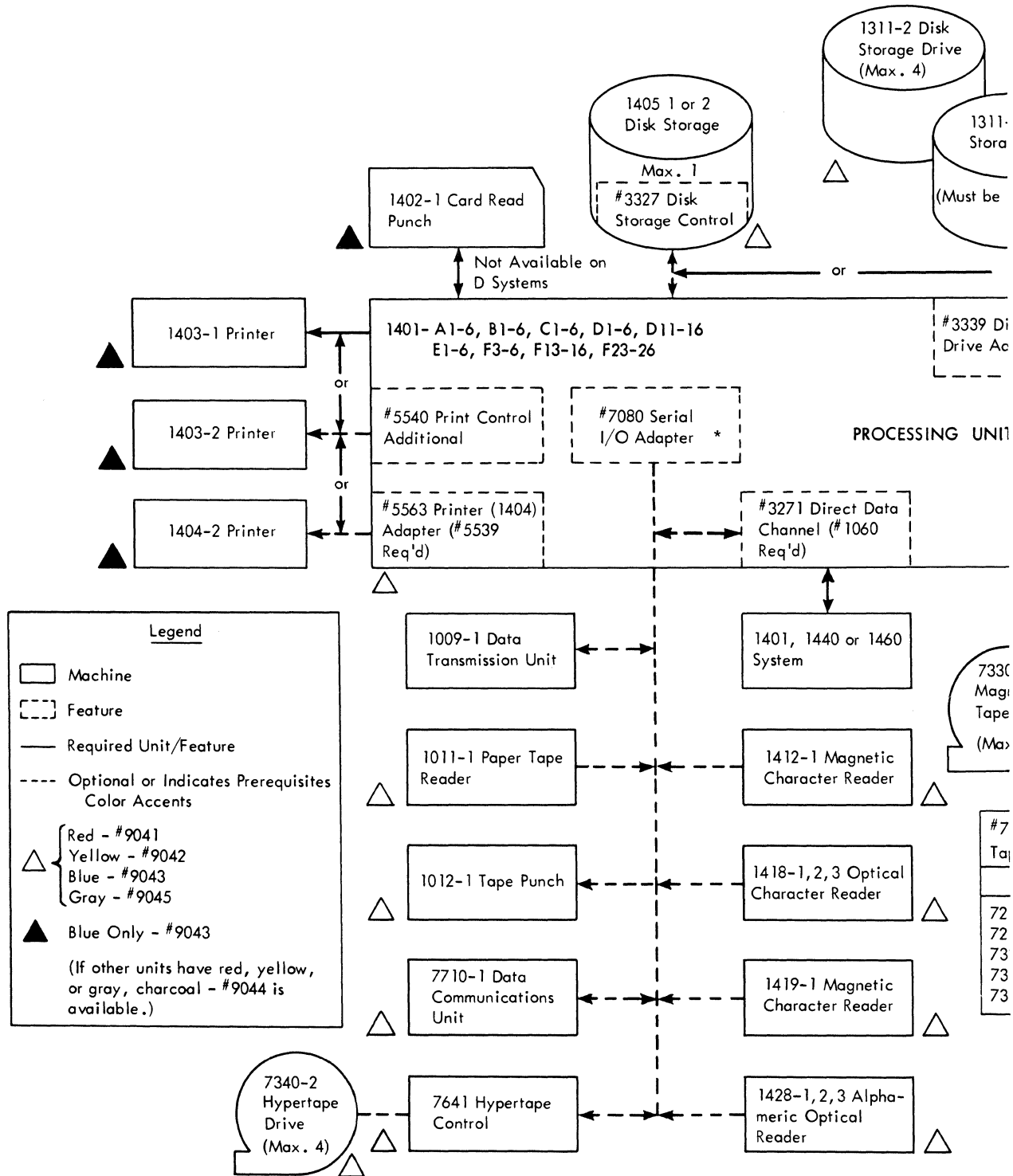
Configurators are not designed to take the place of professional system design and selection by the responsible systems engineer, but rather to be a handy checklist to assist in evaluation of all available system components and their alternatives, and to assure accurate specification of required components.

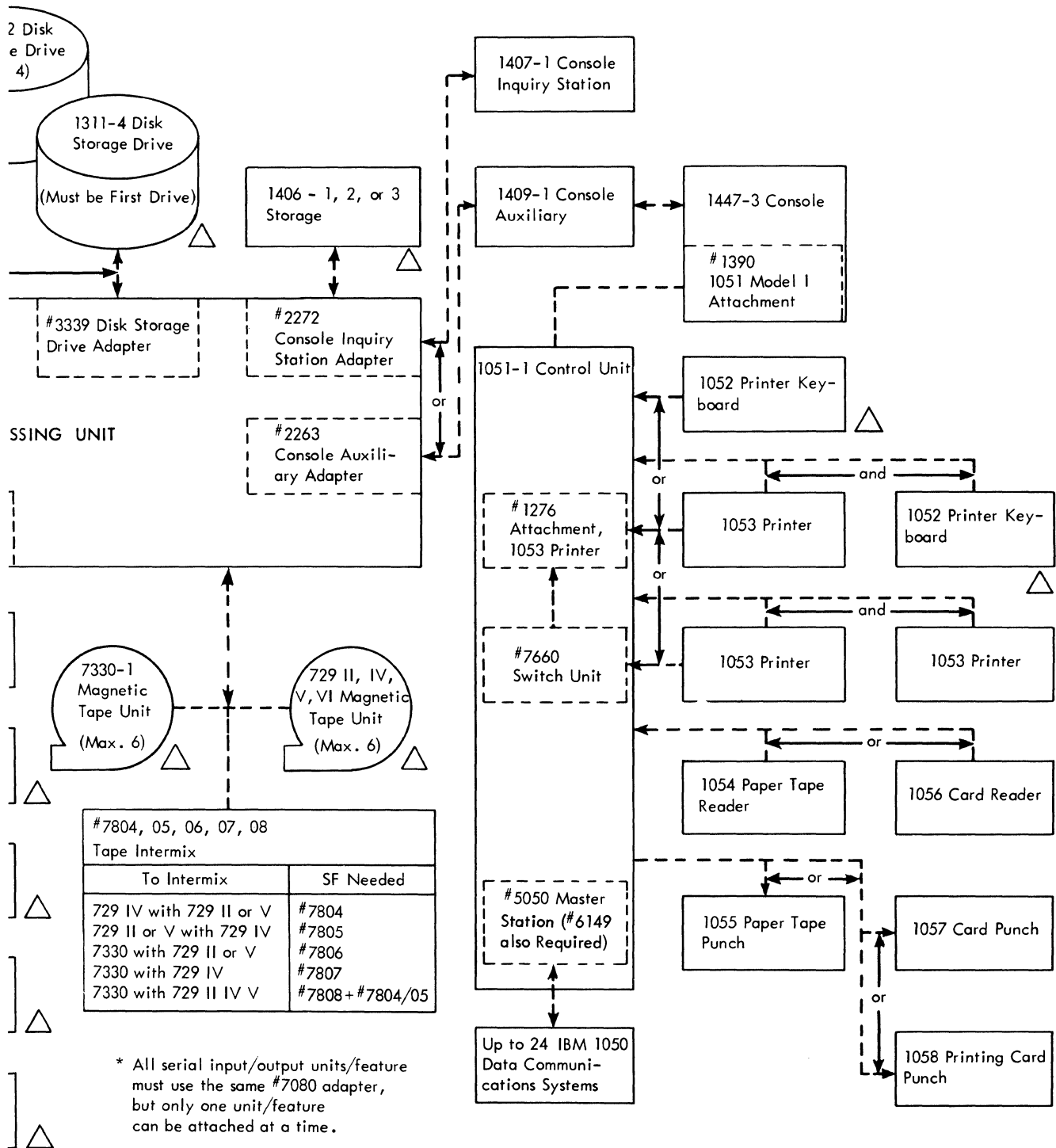
The configurators included in the handbook are, unlike other handbook material, arranged according to system number rather than assigned page number.

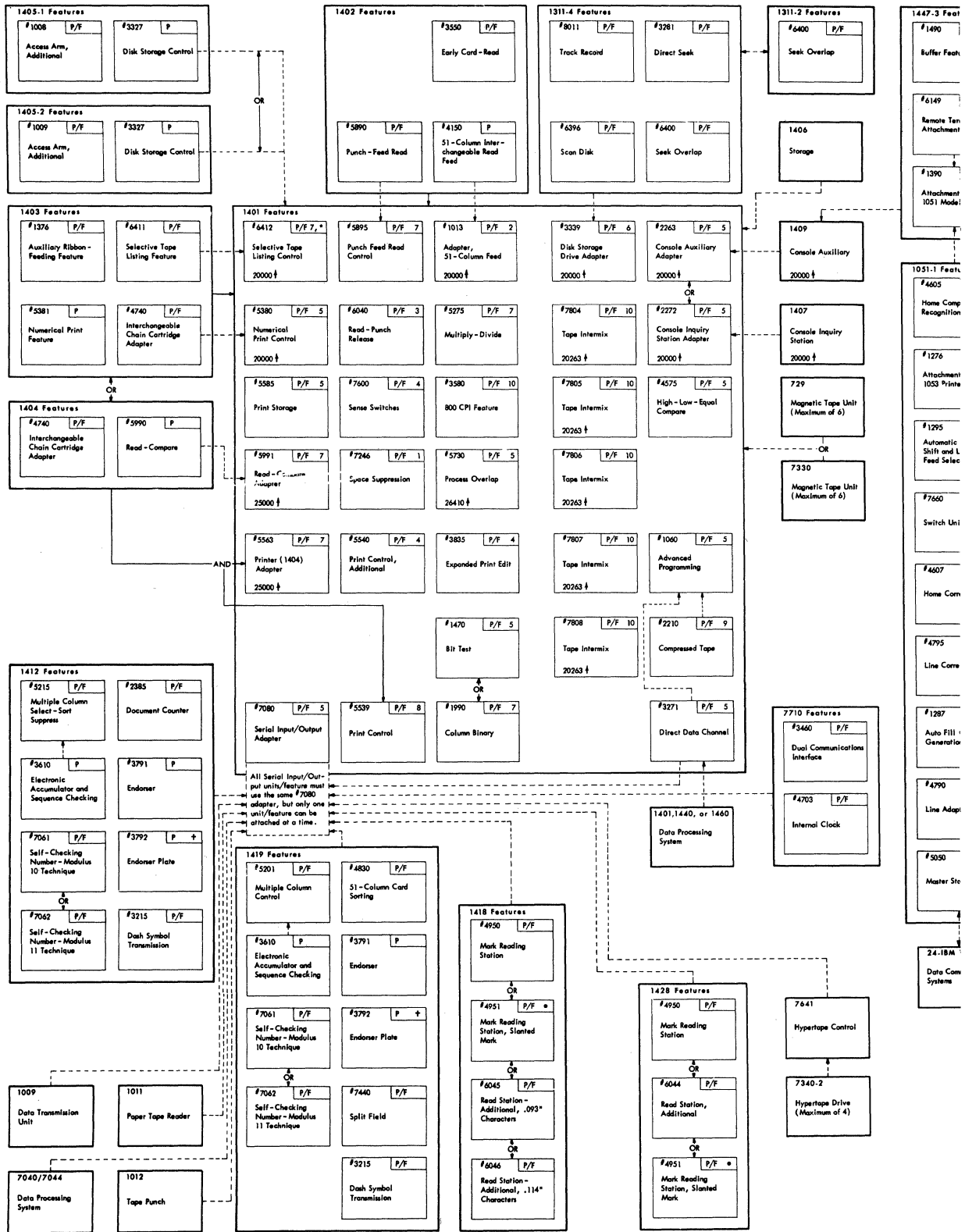




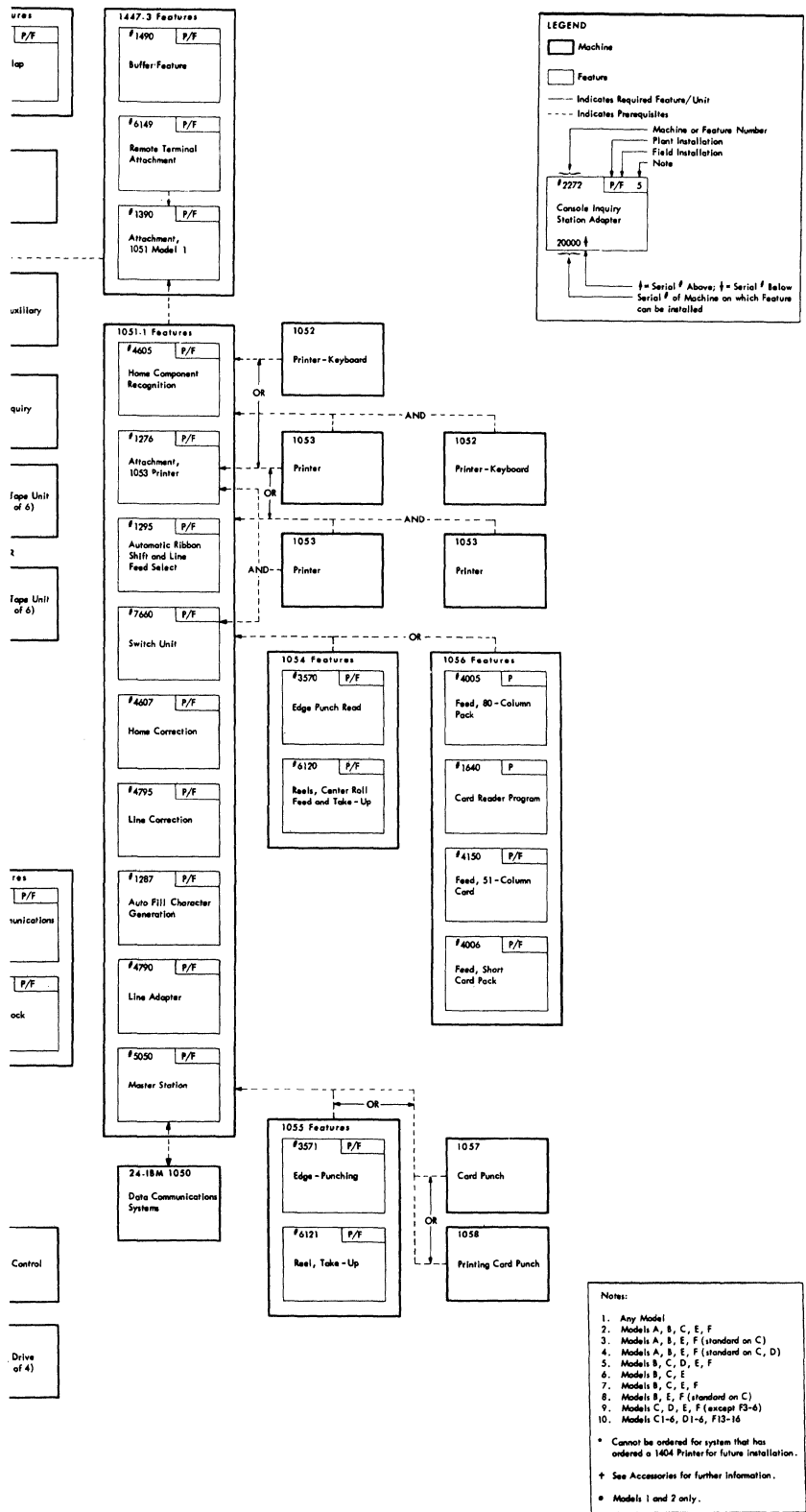
IBM 1401
Configurator







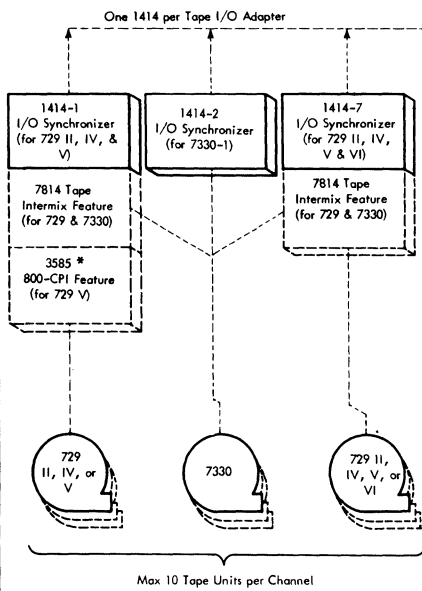
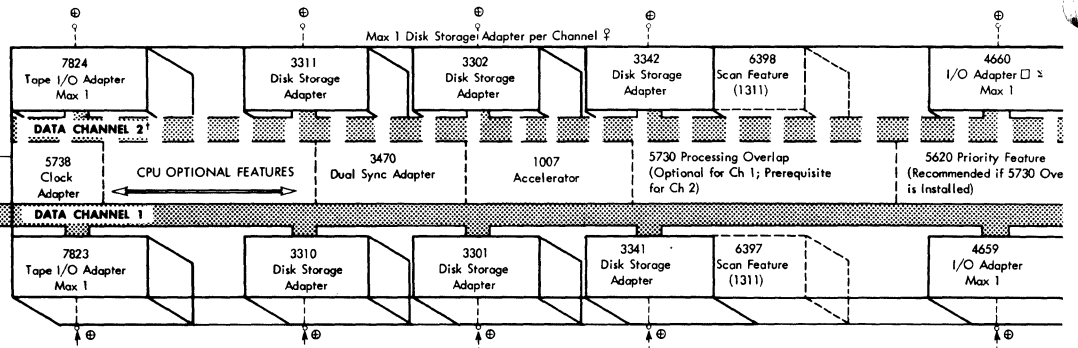
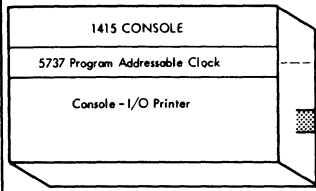
IBM
 International Business Machines Corporation
 Data Processing Division
 112 East Post Road, White Plains, New York







IBM 1410 Configurator



Max 10 Tape Units per Channel

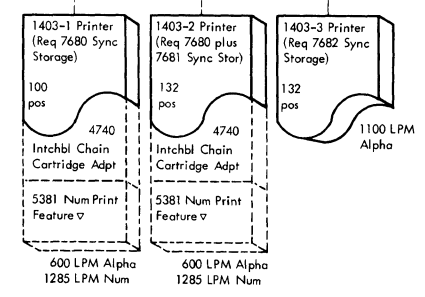
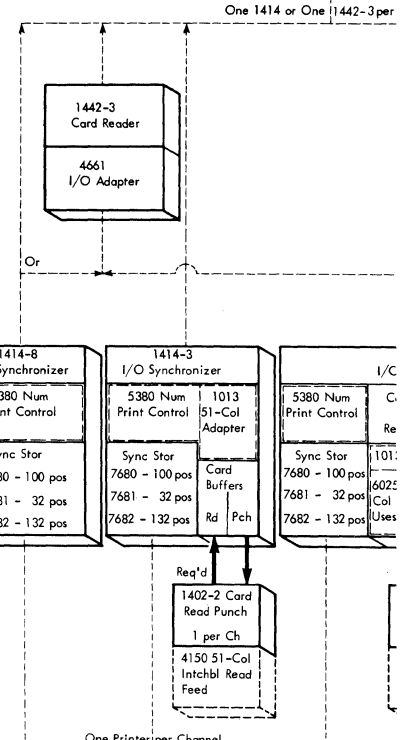
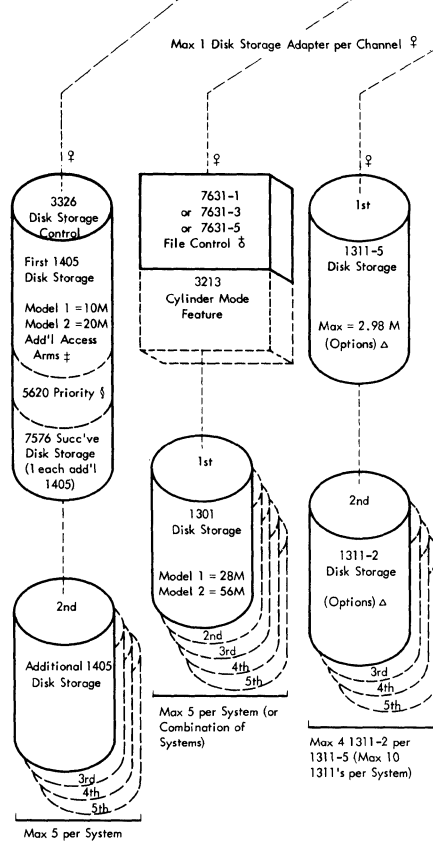
TO USE THIS 1410 CONFIGURATOR:

- Select devices needed.
- READ UP from selected devices to Processing Unit.

Solid lines (—) show requirements
Dashed lines (---) show options

BASE SYSTEM: 1411 Processing Unit
1415 Console
Data Channel 1

Copies of this and other IBM publications can be obtained through IBM branch offices. Address comments concerning the content of this publication to: IBM Corporation, Customer Manuals, Dept. B98, PO Box 390, Poughkeepsie, New York.

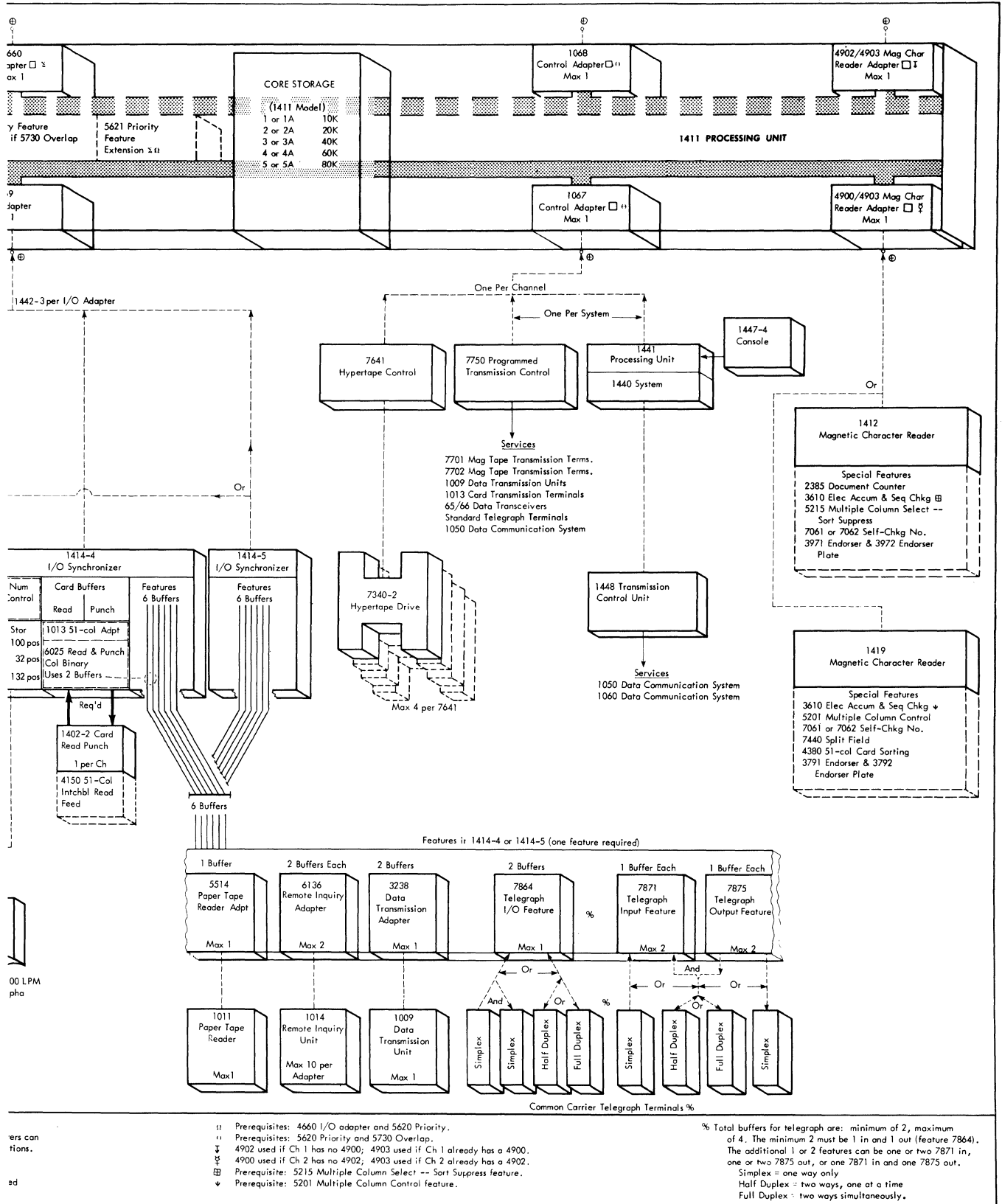


NOTES

* 3586 if on Data Channel 2.
† Prerequisite: 3470 Dual Sync Adapter.
‡ 1008 for Model 1, 1009 for Model 2. Max of 2 additional access arms per 1405 and 12 access arms per system.
§ Optional and recommended. Prerequisite: 5620 Priority on 1411.
¶ Cannot mix 1405 with 1301 or 1311 on system (can mix 1301 and 1311 if on different channels).

♠ 7631-3 permits sharing 1301's with 7000 series systems (except 7010 and 7072). 7631-5 permits shared use by a 7010 or another 1410.
△ Optional features on 1311-5: 6400 Seek Overlap, 6396 Scan Disk, and 8011 Track Record; on 1311-2: 6400 Seek Overlap only. Seek Overlap on 1311-5 necessitates Seek Overlap on all companion 1311-2's. 6396 Scan Disk on 1311 necessitates 6397 or 6398 Scan Feature on CPU.

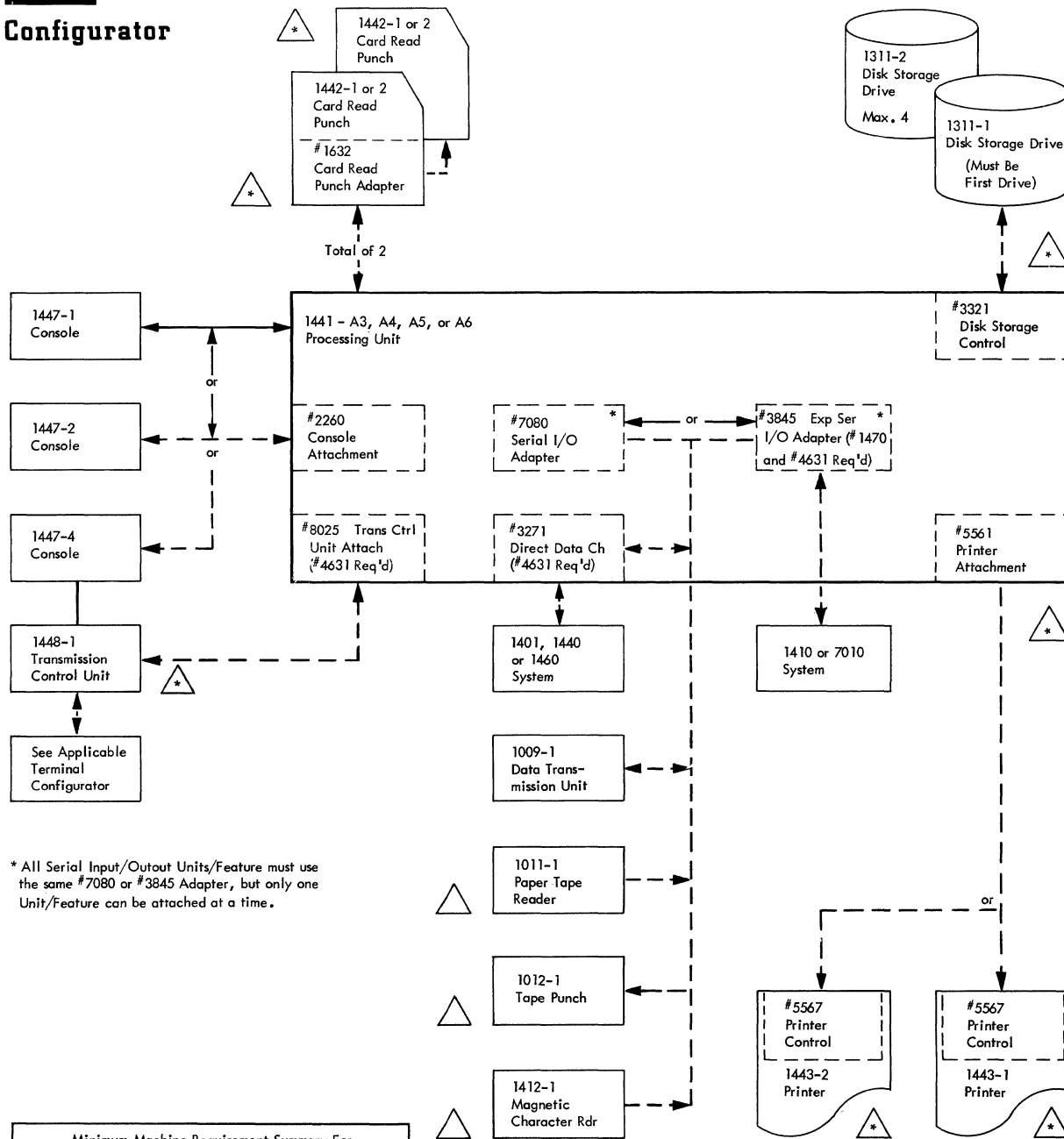
▽ Prerequisites: 4740 on printer and 5380 on 1414.
⊙ I/O devices and features shown attached to Channel 1 adapters can also be attached to Channel 2 adapters, with some restrictions.
□ Prerequisite: 1411 Model A1 through A5.
⋈ 5621 Priority Feature Extension required to extend system priority-interrupt capabilities to I/O operations transmitted through 4660 I/O adapter (Channel 2).





IBM 1440
Configurator

File No. 1440-00
Form A24-3017-4



* All Serial Input/Output Units/Feature must use the same #7080 or #3845 Adapter, but only one Unit/Feature can be attached at a time.

Program Name	Minimum Storage Capacity	1442	1443	1311
Basic Autocoder	4000	1	1	-
1440 Autocoder	4000	1	1	1
I/O Ctrl System	4000	1	1	1
Disk Utility	4000*	1	1	1
Disk File Org	4000	1	1	1
Sort 5	4000	1	1	1
Basic RPG	4000	1	1	-
RPG	4000	1	1	1

* 8K Necessary if Track Record is Used.

<ul style="list-style-type: none"> Machine Feature Required Unit/Feature Optional or Indicates Prerequisites 	<p>Color Accents</p> <ul style="list-style-type: none"> Red—#9041 Yellow—#9042 Blue—#9043 Also Grey—#9045
--	---

IBM 1441 PROCESSING UNIT

<u>All Systems</u>	<u>Model</u>
4,000 Pos. Storage	A3
8,000 Pos. Storage	A4
12,000 Pos. Storage	A5
16,000 Pos. Storage	A6

<u>Special Features</u>	<u>SF No.</u>
Bit Test	#1470
Console Attachment	#2260
Direct Data Channel (#4631 and either #7080 or #3845 req'd)	#3271
Direct Seek	#3281
Disk Storage Control	#3321
Expanded Print Edit	#3835
Expanded Serial Input/ Output Adapter (#1470 and #4631 req'd)	#3845
Indexing and Store	
Address Register	#4631
Multiply-Divide	#5275
Printer Attachment	#5561
Scan Disk	#6396
Seek Overlap Adapter (#6400 req'd on 1311's)	#6399
Serial Input/Output Adapter	#7080
Track Record	#8011
Transmission Control Unit Attachment (#4631 req'd)	#8025

IBM 1442 CARD READ PUNCH

<u>Description</u>	<u>Model</u>
Read 300 CPM; Punch 80 CPS	1
Read 400 CPM; Punch 160 CPS	2

<u>Special Features</u>	<u>SF No.</u>
Card Image (on first 1442 only)	#1531
Card Read-Punch Adapter (Required for first 1442 installed on system)	#1632
Selective Stacker (Model 1 only)	#6406

IBM 1443 PRINTER

<u>Description</u>	<u>Model</u>
150 LPM (52 character set)	1
240 LPM (52 character set)	2

<u>Special Features</u>	<u>SF No.</u>
Character Sets	
13-Character (Model 1)	#1890
13-Character (Model 2)	#1894
39-Character (Model 1)	#1891
39-Character (Model 2)	#1895
52-Character (Model 1)	#1892
52-Character (Model 2)	#1896
63-Character (Model 1)	#1893
63-Character (Model 2)	#1897
Print Positions, 24 Additional	#5559
Print Storage	#5585
Printer Control	#5567
Selective Character Set	#6401

IBM 1447 CONSOLE

<u>Description</u>	<u>Model</u>
Console Only	1
Console with I/O Printer	2
Console with I/O Printer & 1448 Controls	4

<u>Special Feature</u>	<u>SF No.</u>
Sense Switches	#7600

IBM 1311 DISK STORAGE DRIVE

<u>Description</u>	<u>Model</u>
First Drive	1
Additional Drive(s)	2

<u>Special Feature</u>	<u>SF No.</u>
Seek Overlap (#6399 req'd)	#6400

IBM 1412 MAGNETIC CHARACTER READER

<u>Description</u>	<u>Model</u>
950 DPM	1

<u>Special Features</u>	<u>SF No.</u>
Document Counter	#2385
* Electronic Accumulator and Sequence Checking (#5215 req'd)	#3610
* Endorser	#3791
Endorser Plate	#3792
Multiple Column Select-Sort Suppress	#5215
Self-Checking Number	
Modulus 10 Technique	#7061
Modulus 11 Technique	#7062

IBM 1009 DATA TRANSMISSION UNIT

<u>Model</u>
1

IBM 1011 PAPER TAPE READER

<u>Model</u>
1

IBM 1012 TAPE PUNCH

<u>Model</u>
1

IBM 1448 TRANSMISSION CONTROL UNIT

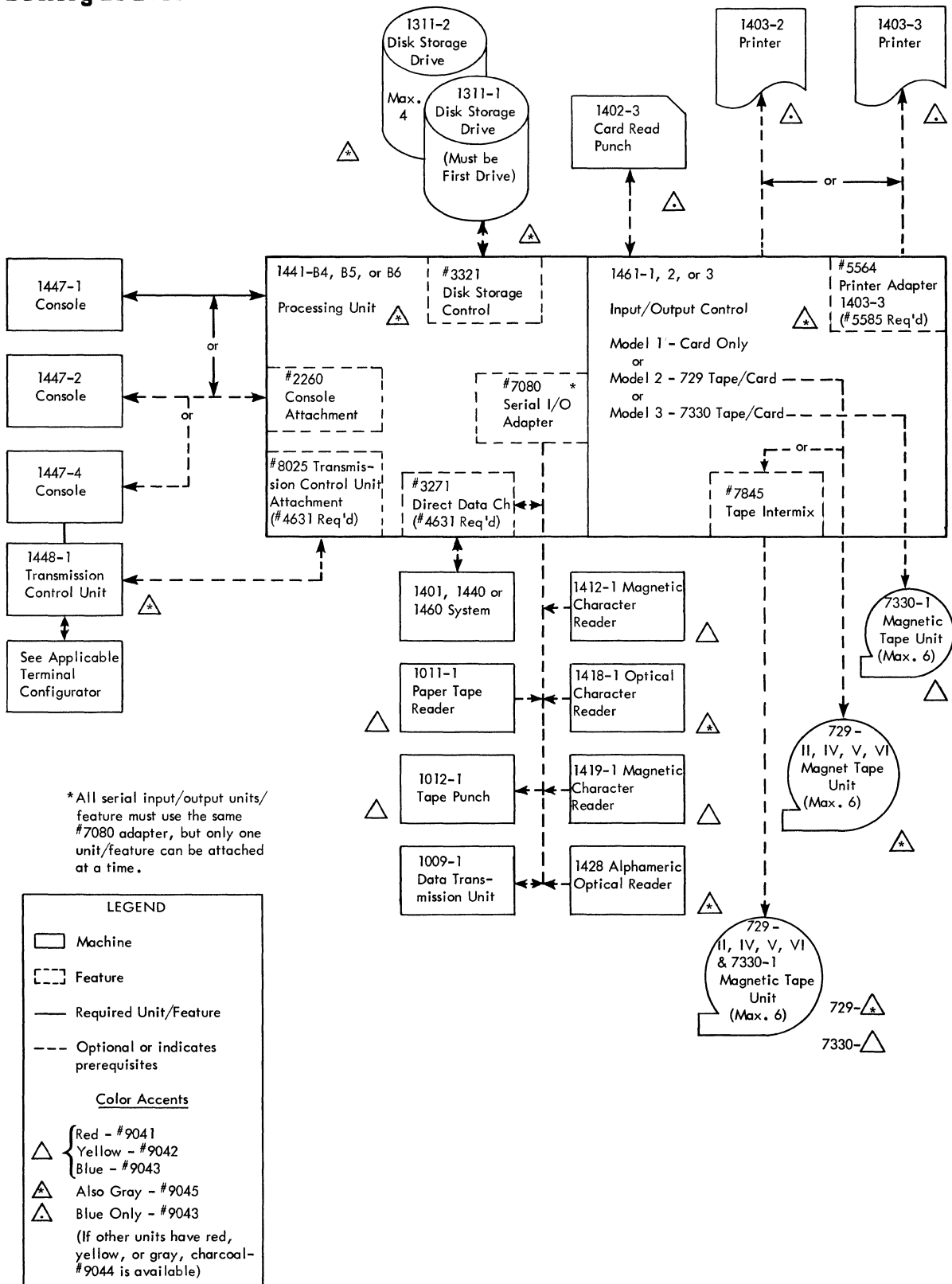
<u>Model</u>
1

Special Features
See Applicable Terminal Configurator

* Plant Installation Only

IBM 1460 Configurator

File Number 1460-00
Form A24-1497-4



IBM 1441 PROCESSING UNIT

All Systems	Model
8,000 Storage Positions	B4
12,000 Storage Positions	B5
16,000 Storage Positions	B6

Special Features	SF No
Bit Test	# 1470
Console Attachment	# 2260
Direct Data Chan (#4031 & #7080 req'd)	# 3271
Direct Seek	# 3281
Disk Storage Control	# 3321
Expanded Print Edit	# 3835
Indexing and Store Address Register	# 4631
Multiply/Divide	# 5275
Scan Disk	# 6396
Seek Ovlp Adapter (#6400 req'd on all attached T311's)	# 6399
Serial I/O Adapter	# 7080
Track Record	# 8011
Transmission Control Unit Attachment (#4631 req'd)	# 8025

IBM 1461 INPUT/OUTPUT CONTROL

All Systems	Model
Card System	1
729 Tape/Card System	2
7330 Tape/Card System	3

Special Features	SF No
Adapter, 51-Column Read Feed	# 1013
Binary Transfer	# 1468
Compressed Tape (Models 2 & 3)	# 2210
Numerical Print Control	# 5380
Print Storage (req'd on 1403 Model 3)	# 5585
Printer Adapter-1403, Mod 3 (req'd)	# 5564
Processing Overlap	# 5730
Punch-Feed Read Control	# 5895
Read-Punch Release	# 6040
Selective Tape Listing Control	# 6412
Tape Intermix (Mod 2 only)	
Attach 7330 (to 1461, Model 2)	# 7845

IBM 1447 CONSOLE

All Systems	Model
Console Only	1
Console with I/O Printer	2
Console with I/O Printer & 1448 Ctrl.	4

Special Features	SF No
Sense Switches	# 7600

IBM 1402 CARD READ-PUNCH, MODEL 3

Special Features	SF No
Feed, 51-Col Interchangeable Read (# 1013 req'd)	# 4150*
Punch-Feed Read (# 5895 req'd)	# 5890

IBM 1403 PRINTER

All Systems	Model
600 Lines per Min. (max.)	2
1100 Lines per Min. (max.)	3

Special Features (Model 2 only)	SF No
Auxiliary Ribbon Feeding	# 1376
Interchangeable Chain Cartridge Adapter	# 4740
Numerical Print (#4740 and #5380 req'd)	# 5381*
Models 2 and 3	
Selective Tape Listing (# 6412 req'd)	# 6411
Change Print Arrangement/Type Size	See Type Catalog
Non-Standard Print Arrangements	
Special Character Arrangements	

IBM 1311 DISK STORAGE DRIVE

All Systems	Model
First Drive	1
Additional Drive(s)	2

Special Feature	SF No
Seek Overlap (#6399 req'd)	# 6400

IBM 1412 MAGNETIC CHARACTER READER

IBM 1419 MAGNETIC CHARACTER READER

Special Features	SF No
Document Counter (1412 only)	# 2385
Electronic Accum and Seq Checking (Req #5215 on 1412; #5201 on 1419)	# 3610*
Multiple-Column Control (1419 only)	# 5201
Mult Col Sel-Sort Sup (1412 only)	# 5215
Self-Checking Number	
Modulus 10 Technique	# 7061
Modulus 11 Technique	# 7062
Split Field (1419 only)	# 7440
Endorser - 1412/1419	# 3791*
Endorser Plate	# 3792
51-Column Card Sorting (1419 only)	# 4380

IBM 1418 OPTICAL CHARACTER READER

All Systems	Model
Three Pockets	1
Thirteen Pockets	2
Three Pockets	3

Special Features	SF No
Mark Rdnng Stn (not with #6045, #6046)	# 4950
Read Station Add'l (not with #4950)	
For Characters .093" High	# 6045
For Characters .114" High	# 6046

IBM 1428 ALPHAMERIC OPTICAL READER

All Systems	Model
Three Pockets	1
Thirteen Pockets	2
Three Pockets	3

Special Features	SF No
Mark Rdnng Stn (not with #6044)	# 4950
Read Station, Add'l (not with #4950)	# 6044

IBM 729 MAGNETIC TAPE UNIT

Model II
Model IV
Model V
Model VI

IBM 7330 MAGNETIC TAPE UNIT

IBM 1009 DATA TRANSMISSION UNIT

IBM 1011 PAPER TAPE READER

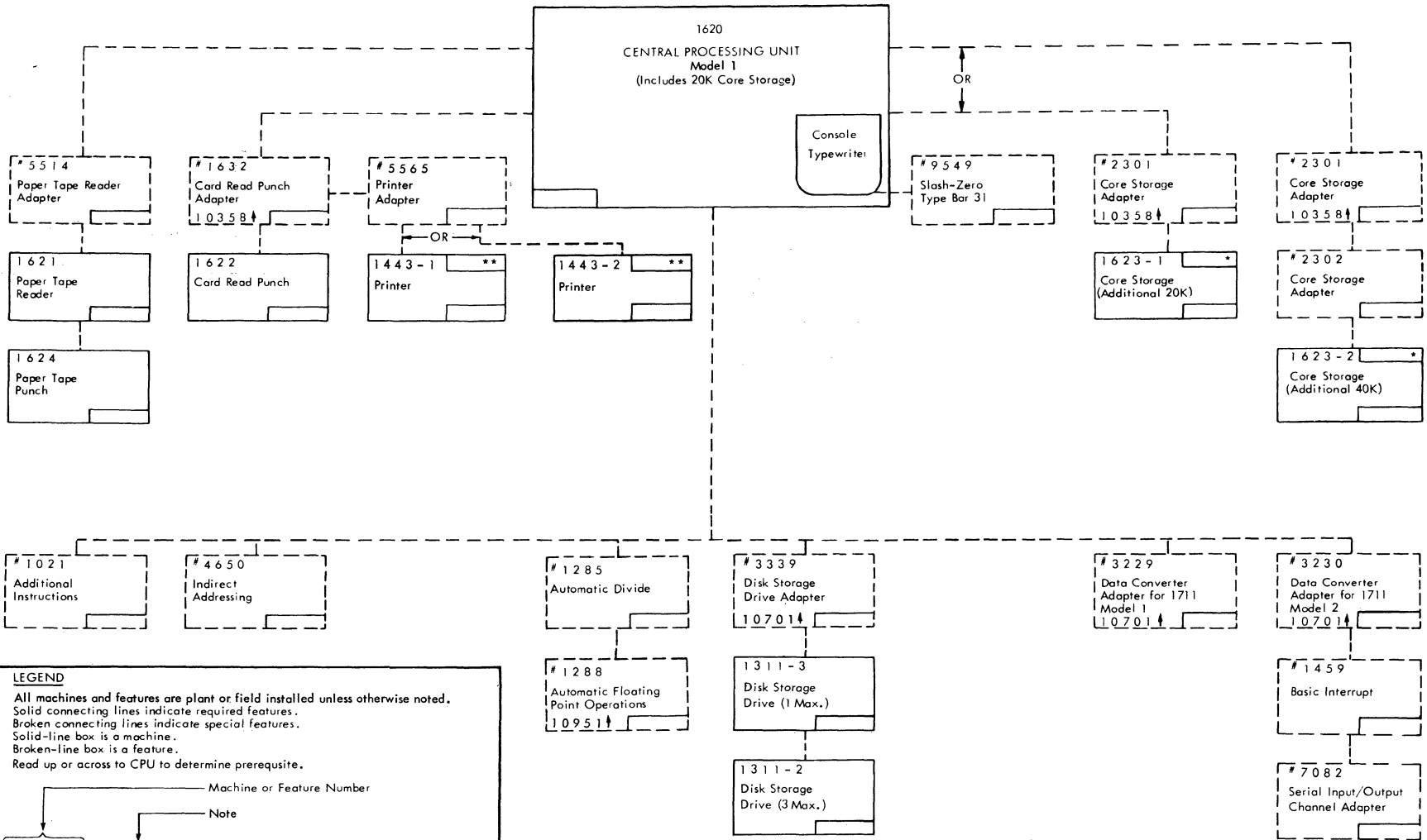
IBM 1012 TAPE PUNCH

IBM 1448 TRANSMISSION CONTROL UNIT

Special Features
See Applicable Terminal Configurator

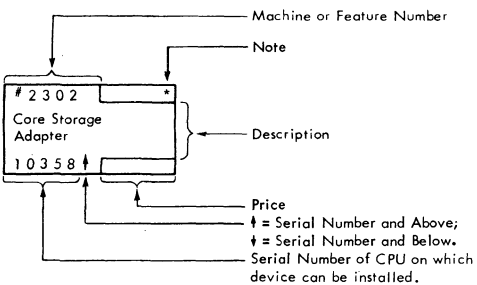
* Factory Installation Only

1620 Model 1 Configurator



LEGEND

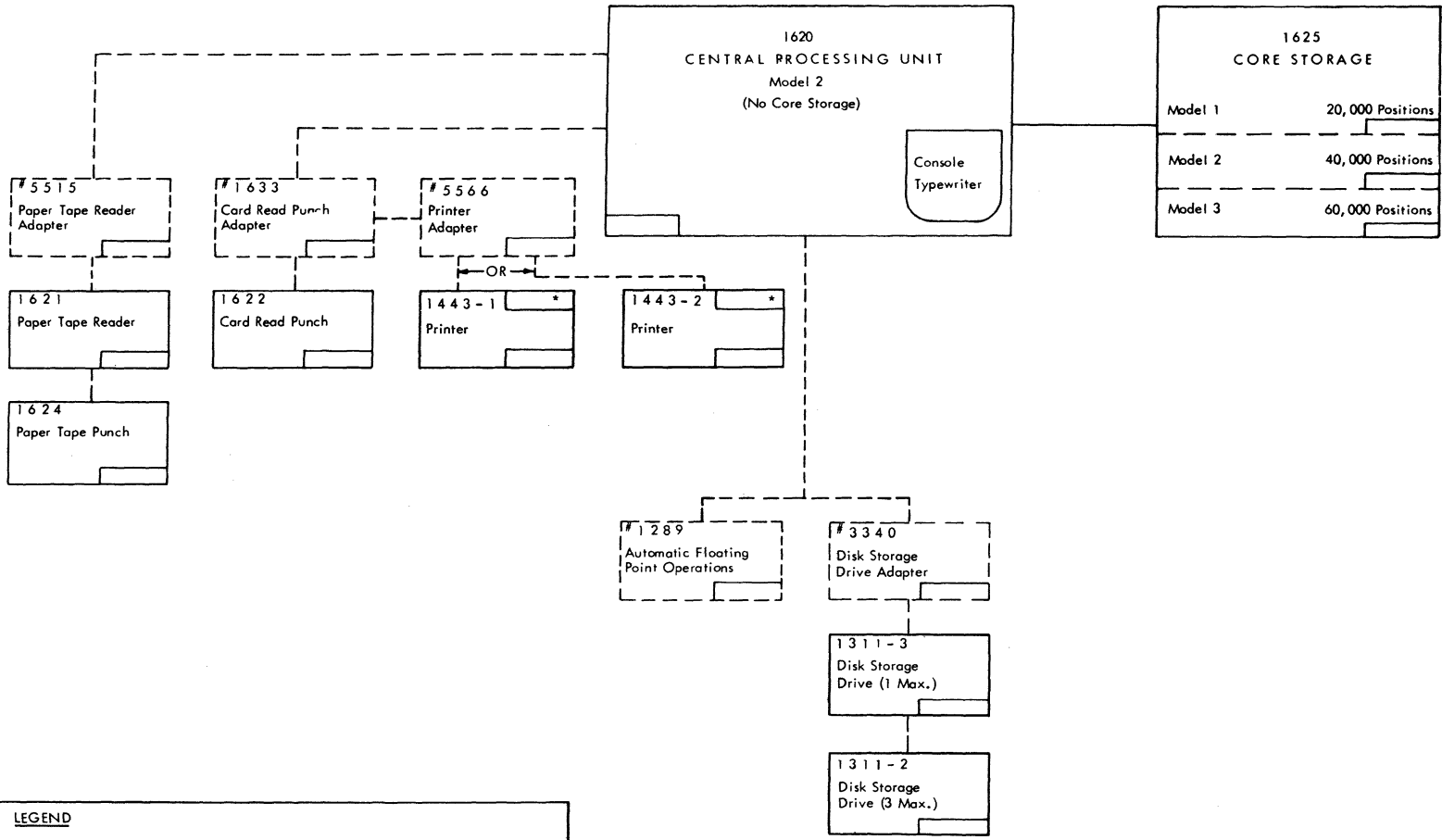
All machines and features are plant or field installed unless otherwise noted.
 Solid connecting lines indicate required features.
 Broken connecting lines indicate special features.
 Solid-line box is a machine.
 Broken-line box is a feature.
 Read up or across to CPU to determine prerequisite.



NOTES

- * 1623-1 cannot be field-converted to 1623-2.
- ** Requires #5568 Printer Control for 1620. Printer Control is plant installed only.

1620 Model 2 Configurator



LEGEND

All machines and features are plant or field installed unless otherwise noted.
 Solid connecting lines indicate required features.
 Broken connecting lines indicate special features.
 Solid-line box is a machine.
 Broken-line box is a feature.
 Read up or across to CPU to determine prerequisite.

Machine or Feature Number

Note

Description

Price

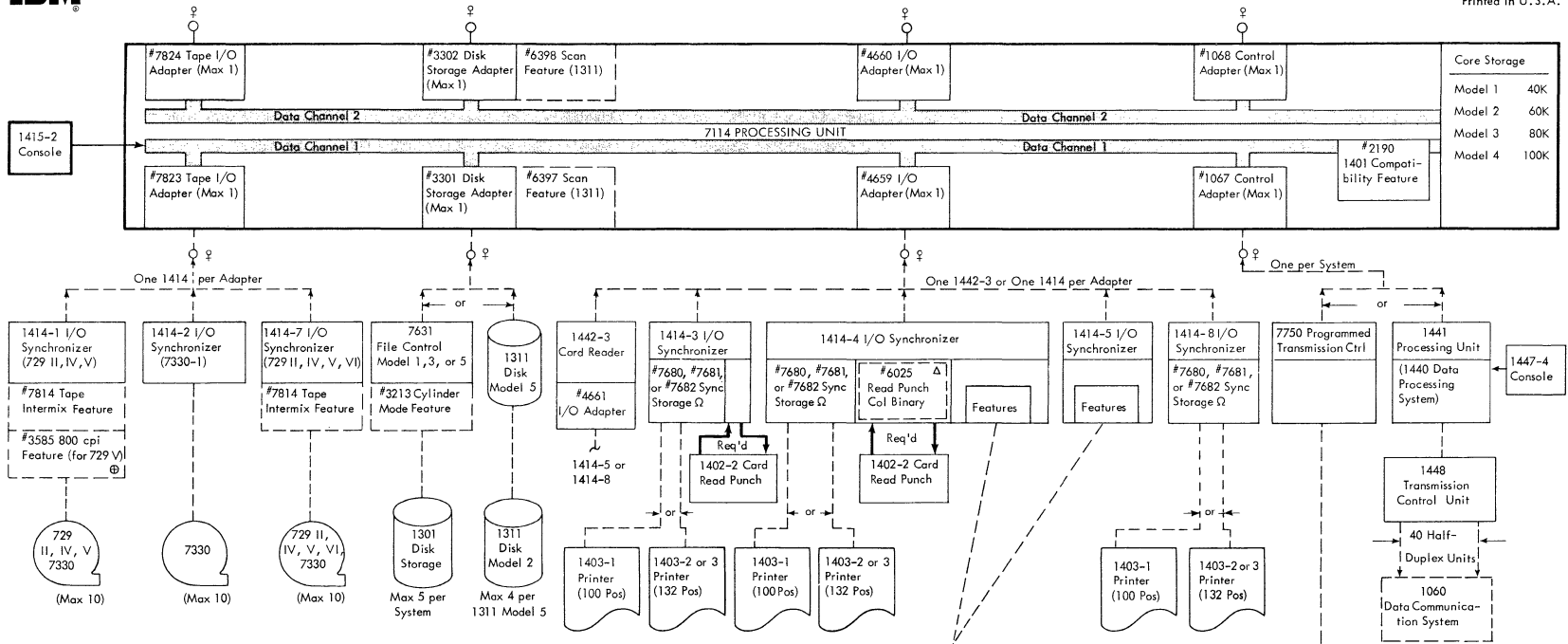
NOTES

* Requires #5568 Printer Control for 1620. Printer Control is plant installed only.



7010 Configuration

A22-6723-1
Printed in U. S. A.



BASE SYSTEM

1. 7114 Processing Unit.
2. 7010 Instruction Set with Priority and Processing Overlap.
3. Data Channel 1 and 1415 Model 2 Console.

OPTIONAL FEATURES

1. Expansion of core storage of 100,000 character positions.
2. Addition of second data channel.
3. 1401 Compatibility Feature.

♀ I/O devices and features shown attached to channel 1 adapters can be attached to the related channel 2 adapters. Channel and systems restrictions for channel 1 apply to channel 2.

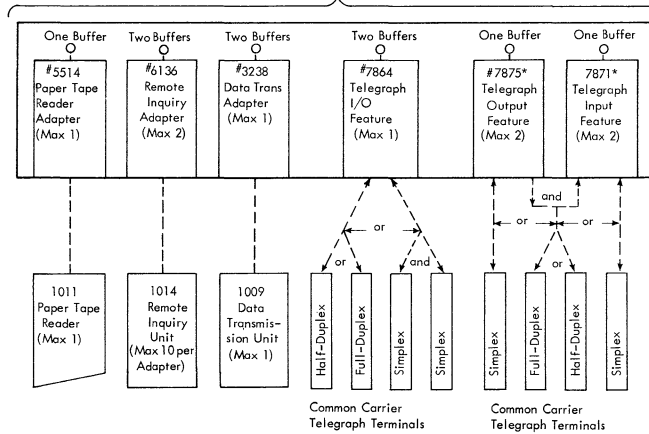
⊕ #3586 if on channel 2.

Note: Channel 2 is identical to channel 1 except that 1415 Model 2 Console and 1401 Compatibility Feature attach only to channel 1.

TO USE THIS 7010 CONFIGURATION:

1. Select devices needed.
2. Follow dotted lines from selected devices to the data channel. Optional features between the selected devices and the data channel are outlined in dotted lines.

Copies of this and other IBM publication can be obtained through IBM branch offices. Address comments concerning the content of this publication to: IBM Corporation, Customer Manuals, Dept B98, PO Box 390, Poughkeepsie, N. Y.



Notes:

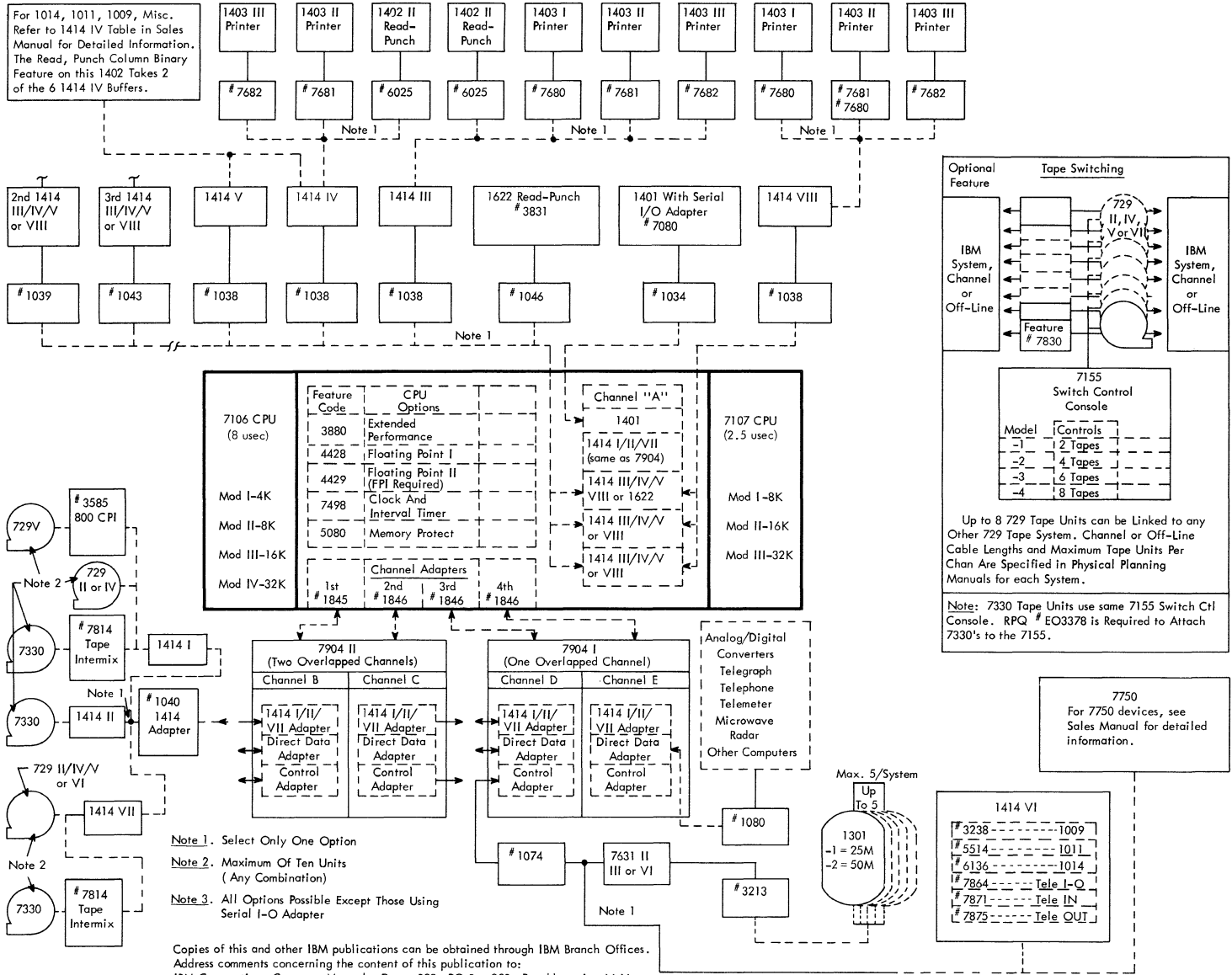
1. Maximum of six buffers available on 1414-4 and 1414-5.
- * Requires two 1414 buffers.
- ⊕ Feature 7864 is a prerequisite for Feature 7871 or Feature 7875.
- ⊖ For 1403-1, Feature 7680; for 1403-2, Features 7680 and 7681; for 1403-3, Feature 7682.

IBM

7040-7044 Configurator

A22-6685-2
Printed in U.S.A.

For 1014, 1011, 1009, Misc. Refer to 1414 IV Table in Sales Manual for Detailed Information. The Read, Punch Column Binary Feature on this 1402 Takes 2 of the 6 1414 IV Buffers.



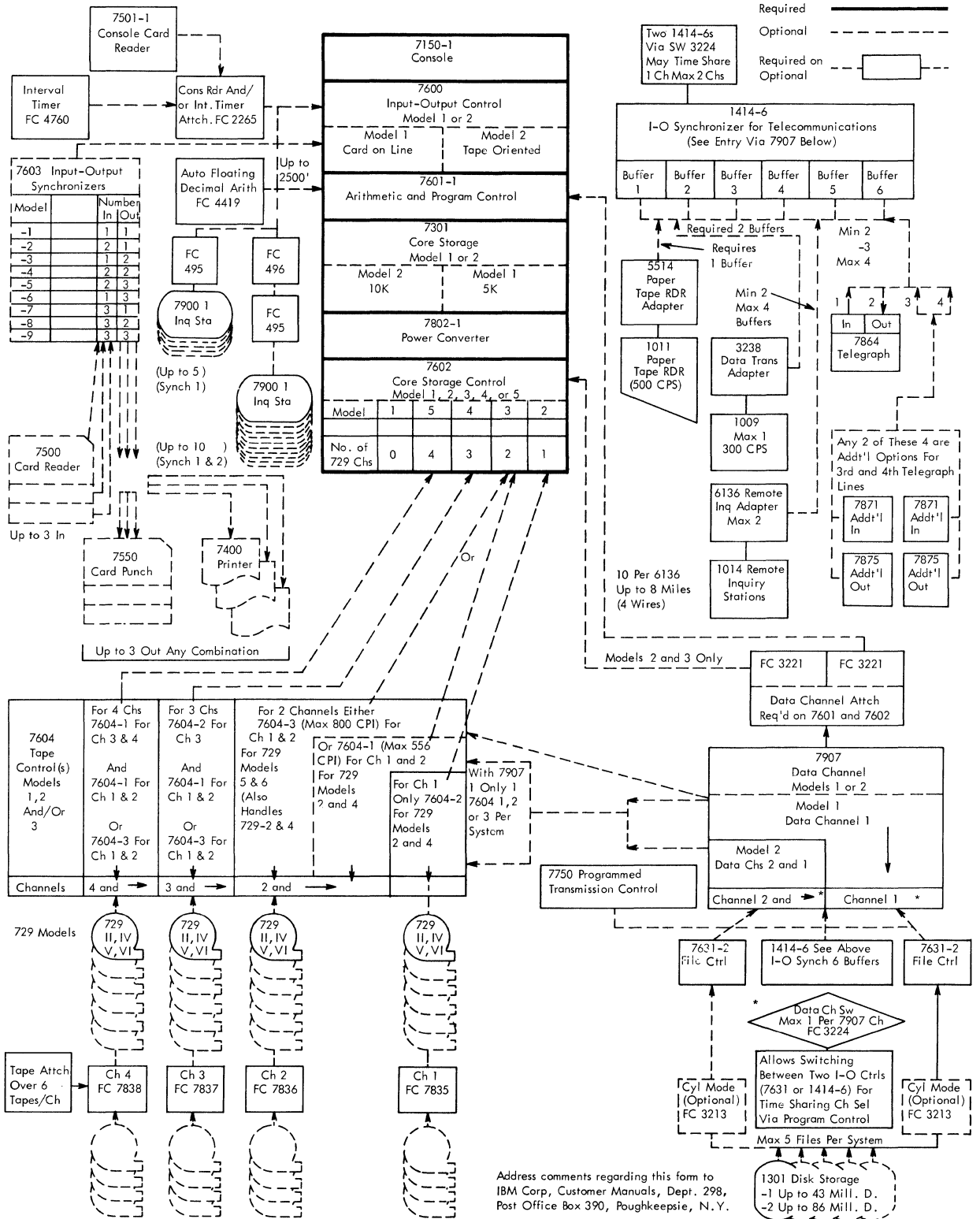
- Note 1.** Select Only One Option
- Note 2.** Maximum Of Ten Units (Any Combination)
- Note 3.** All Options Possible Except Those Using Serial I-O Adapter

Copies of this and other IBM publications can be obtained through IBM Branch Offices. Address comments concerning the content of this publication to: IBM Corporation, Customer Manuals, Dept. B98, PO Box 390, Poughkeepsie, N.Y.

IBM

7070 Data Processing System Configuration

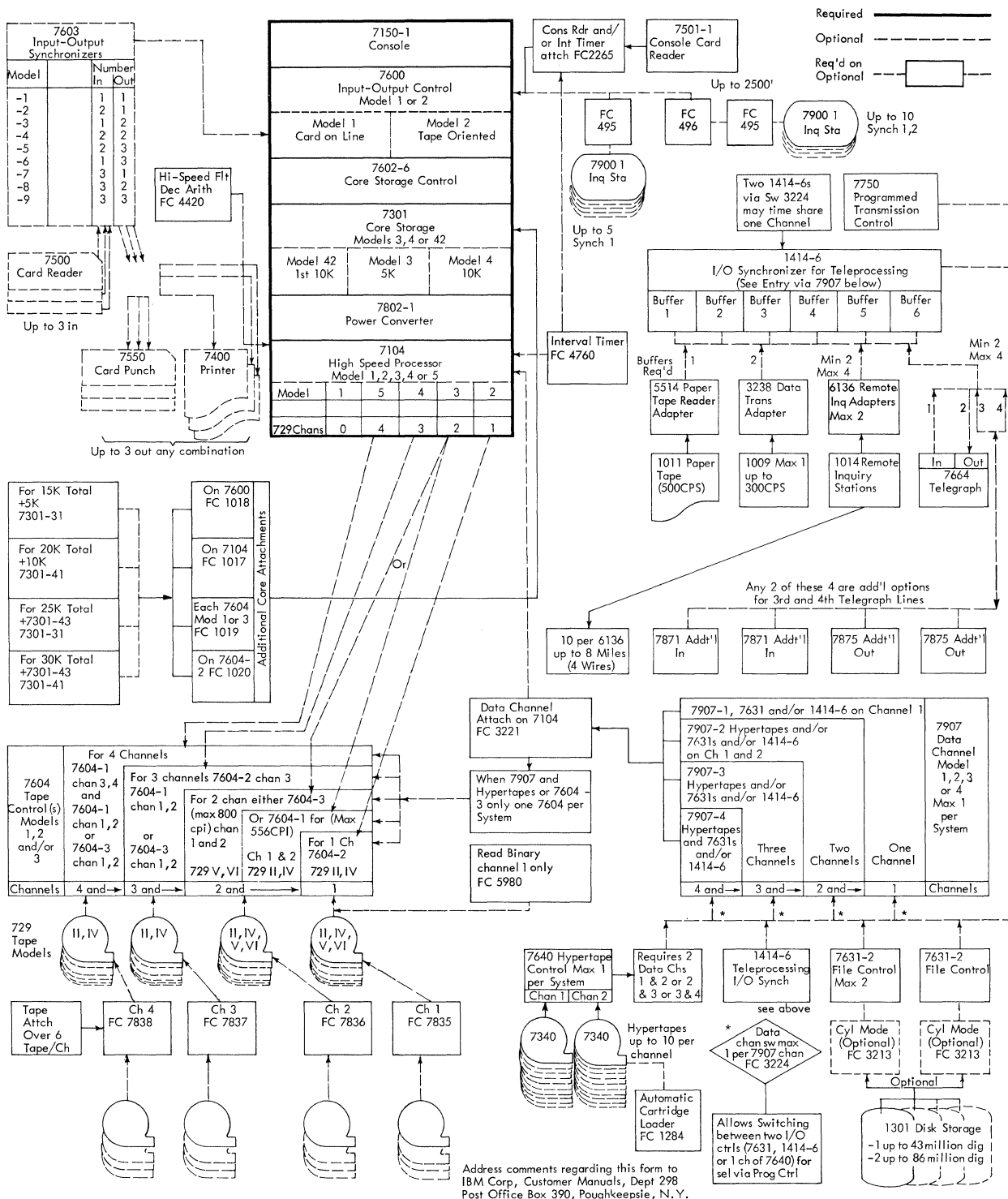
A22-6683
PRINTED IN U. S. A.





7074 Data Processing System Configuration

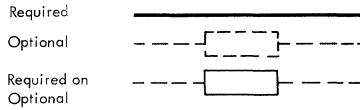
A22-6684-1
PRINTED IN U. S. A.



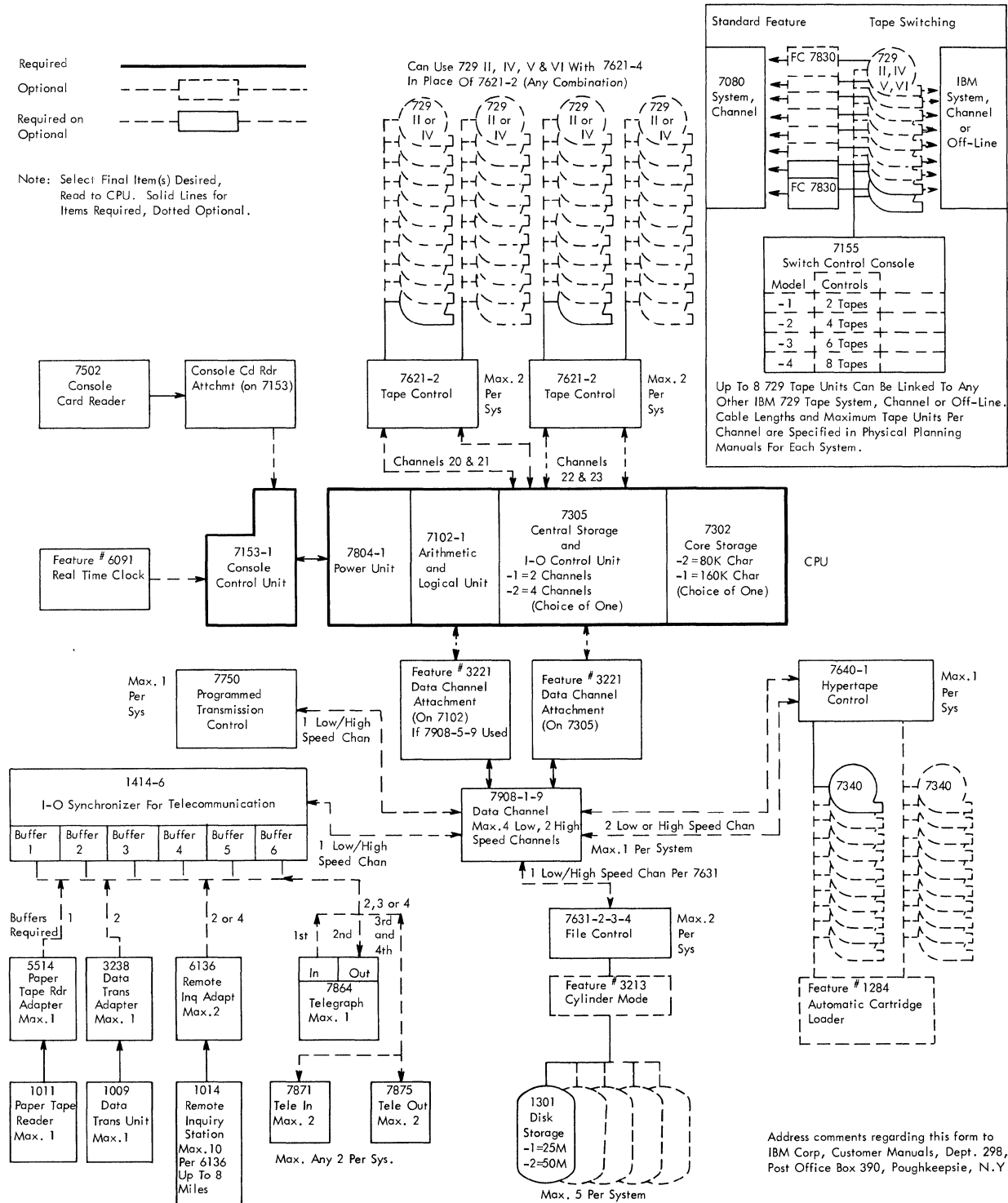
IBM

7080 Data Processing System Configuration

A22-6687
PRINTED IN U. S. A.



Note: Select Final Item(s) Desired, Read to CPU. Solid Lines for Items Required, Dotted Optional.



Address comments regarding this form to IBM Corp, Customer Manuals, Dept. 298, Post Office Box 390, Poughkeepsie, N.Y.

IBM 7090 Configurator

A22-6686-1
PRINTED IN U.S.A.

Required

Optional

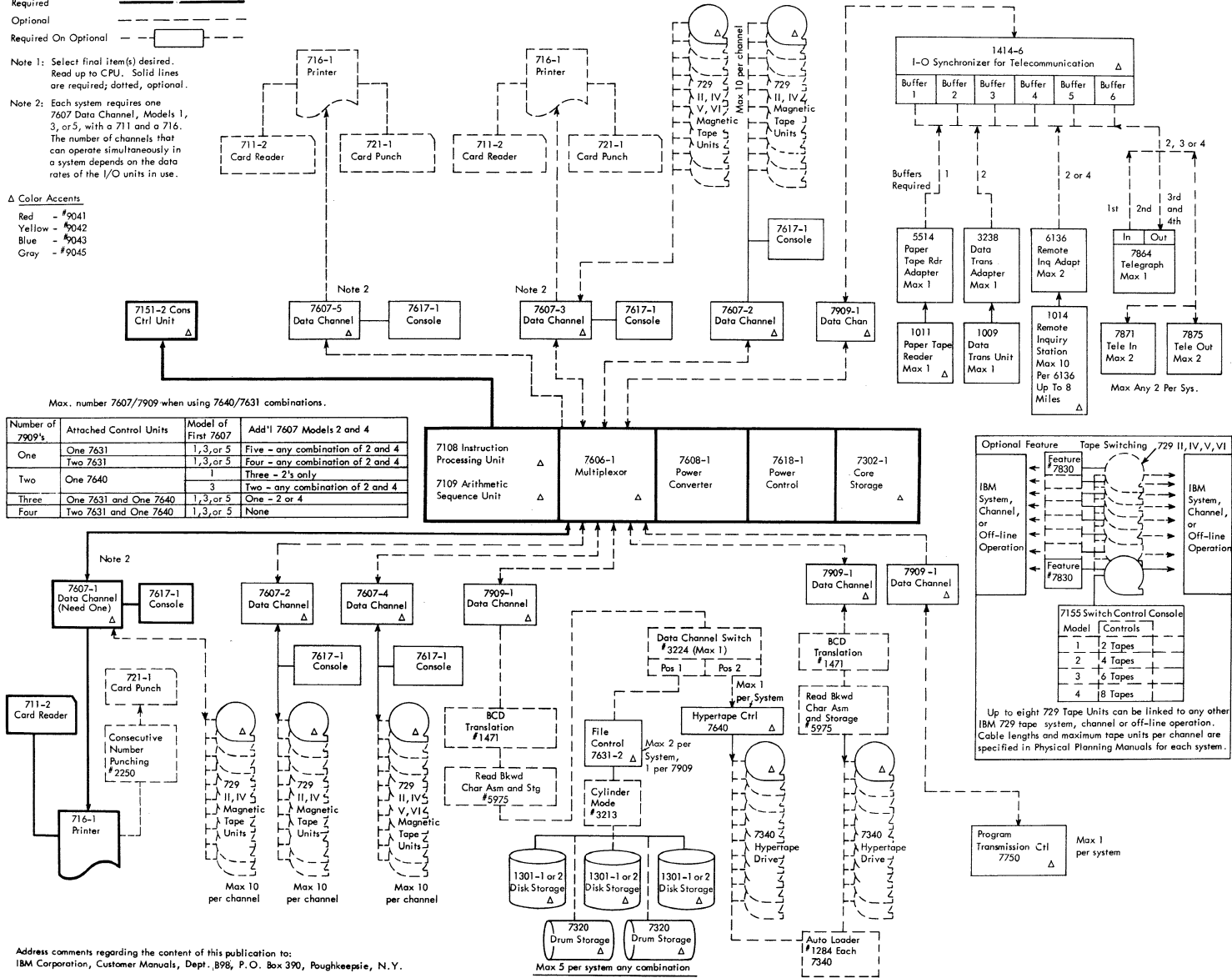
Required On Optional

Note 1: Select final item(s) desired. Read up to CPU. Solid lines are required; dotted, optional.

Note 2: Each system requires one 7607 Data Channel, Models 1, 3, or 5, with a 711 and a 716. The number of channels that can operate simultaneously in a system depends on the data rates of the I/O units in use.

Δ Color Accents

- Red - #9041
- Yellow - #9042
- Blue - #9043
- Gray - #9045



Address comments regarding the content of this publication to:
IBM Corporation, Customer Manuals, Dept. B98, P.O. Box 390, Poughkeepsie, N.Y.

09

11

16

IBM 7094 Configurator

A22-6689-2
PRINTED IN U. S. A.

Required _____
Optional - - - - -
Required On Optional - - - - -

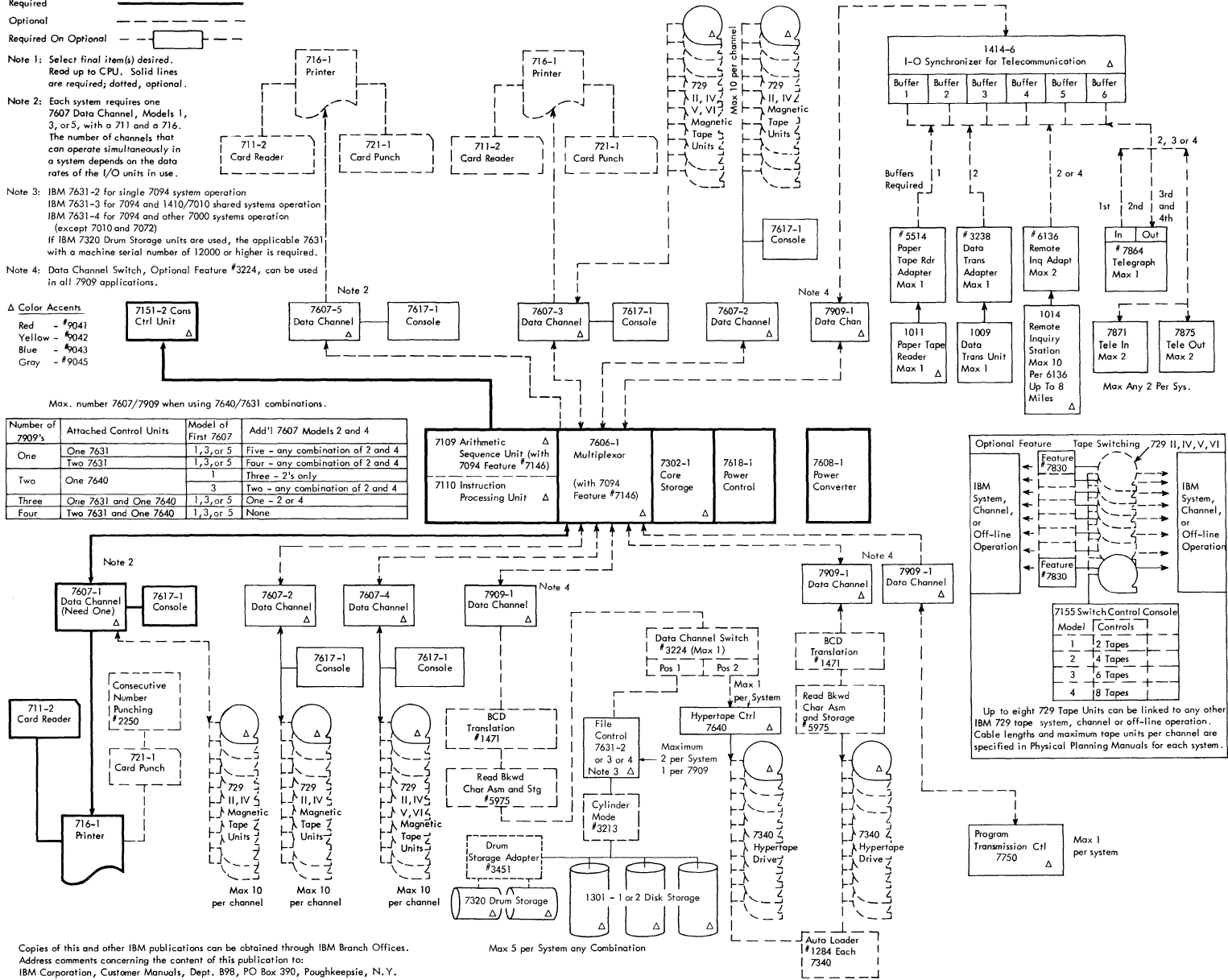
Note 1: Select final item(s) desired. Read up to CPU. Solid lines are required; dotted, optional.

Note 2: Each system requires one 7607 Data Channel, Models 1, 3, or 5, with a 711 and a 716. The number of channels that can operate simultaneously in a system depends on the data rates of the I/O units in use.

Note 3: IBM 7631-2 for single 7094 system operation
IBM 7631-3 for 7094 and 1410/7010 shared systems operation
IBM 7631-4 for 7094 and other 7000 systems operation (except 7010 and 7072)
If IBM 7320 Drum Storage units are used, the applicable 7631 with a machine serial number of 12000 or higher is required.

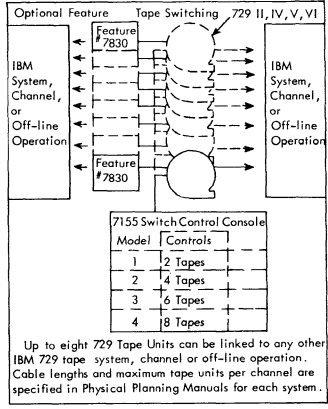
Note 4: Data Channel Switch, Optional Feature #3224, can be used in all 7909 applications.

Δ Color Accents
Red - #9041
Yellow - #9042
Blue - #9043
Gray - #9045



Max. number 7607/7909 when using 7640/7631 combinations.

Number of 7909's	Attached Control Units	Model of First 7607	Add'l 7607 Models 2 and 4
One	One 7631	1, 3, or 5	Five - any combination of 2 and 4
	Two 7631	1, 3, or 5	Four - any combination of 2 and 4
Two	One 7640	1	Three - 2's only
	One 7631 and One 7640	3	Two - any combination of 2 and 4
Three	One 7631 and One 7640	1, 3, or 5	One - 2 or 4
	Two 7631 and One 7640	1, 3, or 5	None



Copies of this and other IBM publications can be obtained through IBM Branch Offices. Address comments concerning the content of this publication to: IBM Corporation, Customer Manuals, Dept. B98, PO Box 390, Poughkeepsie, N.Y.

Max 5 per System any Combination



IBM SYSTEMS REFERENCE CARDS

One aspect of system installation is the conversion of the designed operating procedure into the programmed language of the system. The programmer who accomplishes this must have a great amount of information at his fingertips. Not only must he have a thorough understanding of what the procedure is to perform, but he must also be aware of the system configuration, have an intimate knowledge of the operations of the system components, and be able to accurately prepare estimates of the time required to operate the programmed procedure.

To aid the programmer in keeping straight the many detailed system operating facts, a printed reference card is available from Stationery Stores for major IBM systems. Reference cards are published and kept up to date by the respective manufacturing divisions and are designed to provide a ready summary of each system's prominent programming and timing characteristics. While these

characteristics may vary for different types of machines (internal binary versus binary coded decimal, for example), the basic contents of the reference cards are similar for all systems. They usually contain instruction format, instruction set mnemonics with corresponding binary or BCD code and card code, timing formulas, and internal binary or BCD-to-card-code conversion chart.

Reference cards are primarily for use by customer and IBM personnel who are actively engaged in the detailed programming of a specific computer system. They also serve as a reference chart for individuals with a previous thorough knowledge of the specific system programming features.

Note that the reference cards included in the handbook are, unlike other handbook material, arranged according to system number rather than assigned page number.



IBM 1401 Data Processing System Reference Card

SYSTEM TIMINGS

SYSTEM TIMINGS

Key to abbreviations used in formulas

- L_A = Length of the A-field
- L_B = Length of the B-field
- L_O = Length of Multiplicand field
- L_I = Length of Instruction
- L_{Mx} = Length of Multiplier field
- L_Q = Length of Quotient field
- L_{Kx} = Length of Divisor field
- L_S = Number of significant digits in Divisor (Excludes high-order 0's and blanks)
- L_W = Length of A- or B-field, whichever is shorter
- L_X = Number of characters to be cleared
- L_Y = Number of characters back to right-most "0" in control field
- L_Z = Number of 0's inserted in a field
- I/O = Timing for Input or Output cycle
- F_m = Forms movement times. Allow 20 ms for first space, plus 5 ms for each additional space
- T_m = Tape movement times
- Σ = Number of fields included in an operation

OPERATION	OP CODE	FORMULA
Punch a Card	4	.0115 ($L_I + 1$) + I/O
Read a Card	1	.0115 ($L_I + 1$) + I/O
Read and Punch	5	.0115 ($L_I + 1$) + I/O
Select Stacker	K	.0115 ($L_I + 1$)
Set Word Mark	9	.0115 ($L_I + 3$)
Start Punch Feed*	9	.0115 ($L_I + 1$)
Start Read Feed*	8	.0115 ($L_I + 1$)
Store A-address Register*	Q	.0115 ($L_I + 5$)
Store B-address Register*	H	.0115 ($L_I + 4$)
Subtract (no recomplement)	S	.0115 ($L_I + 3 + L_A + L_B$)
Subtract (recomplement)	S	.0115 ($L_I + 3 + L_A + 4 L_B$)
Write a Line	2	.0115 ($L_I + 1$) + I/O
Write and Punch	6	.0115 ($L_I + 1$) + I/O
Write and Read	3	.0115 ($L_I + 1$) + I/O
Write, Read and Punch	7	.0115 ($L_I + 1$) + I/O
Zero and Add	?	.0115 ($L_I + 1 + L_A + L_B$)
Zero and Subtract	I	.0115 ($L_I + 1 + L_A + L_B$)

INSTRUCTION FORMAT

The IBM 1401 Data Processing System uses a variable word-length concept; the length of an instruction can vary from one to eight characters.

OP CODE	A- or I-ADDRESS	B-ADDRESS	d-CHARACTER
X	XXX	XXX	X

Op Code: This is always a single character which defines the basic operation being performed. A word mark is always associated with the operation code position of an instruction.

A-Address: This always consists of three characters. It can identify the units position of the A-field, or it can be used to select a special unit or feature (tape unit, column binary feature, disk storage, inquiry, etc.).

I-Address: Instructions that can cause program branches use the I-address to specify the location of the next instruction to be executed if a branch occurs.

B-Address: This is a three-character storage address associated with the B-field. It usually addresses the units position of the B-field, but in some operations, such as tape or disk record read and write, it specifies the high-order position of a record storage area.

d-Character: The d-character is used to modify an operation code. It is a single alphabetic, numerical, or special character, positioned as the last character of an instruction. It can be used with instructions of any length.

OPERATION	OP CODE	FORMULA
Add (no recomplement)	A	.0115 ($L_I + 3 + L_A + L_B$)
Add (recomplement)	A	.0115 ($L_I + 3 + L_A + 4 L_B$)
Branch	B	.0115 ($L_I + 1$)
Branch if Bit Equal*	W	.0115 ($L_I + 2$)
Branch if Character Equal	B	.0115 ($L_I + 2$)
Branch if Indicator On	B	.0115 ($L_I + 1$)
Branch if Word Mark and/or Zone	V	.0115 ($L_I + 2$)
Clear Storage	/	.0115 ($L_I + 1 + L_X$)
Clear Word Mark	□	.0115 ($L_I + 3$)
Compare	C	.0115 ($L_I + 1 + L_A + L_B$)
Control Carriage	F	.0115 ($L_I + 1$) + F_m
Control Unit	U	.0115 ($L_I + 1$) + T_m
Divide (aver.)*	%	.0115 ($L_I + 2 + 7 L_K L_Q + 8 L_Q$)
Halt	*	.0115 ($L_I + 1$)
Load Characters to A	L	.0115 ($L_I + 1 + 2 L_A$)
Word Mark	#	.0115 ($L_I + 7$ or 9)
Modify Address*	#	.0115 ($L_I + 7$ or 9)
Move Characters to A or B Word Mark	M	.0115 ($L_I + 1 + 2 L_W$)
Move Characters and Edit	E	.0115 ($L_I + 1 + L_A + L_B + L_Y$)
Move Characters to Record or Word Mark*	P	.0115 ($L_I + 1 + 2 L_A$)
Move Characters and Suppress Zeros	Z	.0115 ($L_I + 1 + 3 L_A$)
Move and Insert Zeros*	X	.0115 ($L_I + 1 + 2 \Sigma L_A + \Sigma L_Z$)
Move Numeric	D	.0115 ($L_I + 3$)
Move Zone	Y	.0115 ($L_I + 3$)
Multiply (aver.)*	@	.0115 ($L_I + 3 + 2 L_C + 5 L_C L_M + 7 L_M$)
No Operation	N	.0115 ($L_I + 1$)

* Special Feature

TAPE OPERATIONS

T_m = Tape movement can be determined from the following:

- I = Number of Characters
- C = Character Rate
- 729 II at 200 cpi = .067 ms
- 729 IV at 200 cpi = .044 ms
- 730 at 556 cpi = .024 ms
- 730 at 556 cpi = .017 ms
- 730 at 556 cpi = .016 ms
- 730 at 200 cpi = .139 ms
- 730 at 556 cpi = .050 ms

729 Model II and V	Read	$10.7 + CN$ ms = TAU interlocked
	Write	$11.7 + CN$ ms = TAU interlocked
		$7.5 + CN$ ms = Processing interlocked
729 Model IV,	Read	$6.8 + CN$ ms = TAU interlocked
	Write	$7.8 + CN$ ms = TAU interlocked
		$5 + CN$ ms = Processing interlocked
7330	Read	$20.5 + CN$ ms = TAU interlocked
	Write	$7 + CN$ ms = Processing interlocked
		$20.3 + CN$ ms = TAU interlocked
		$5 + CN$ ms = Processing interlocked

- Rewind
- 729 Models II and V = 1.2 minutes/reel
- 729 Model IV = .9 minutes/reel
- 7330 (High Speed) = 2.2 minutes/reel
- 7330 = 103 ms

- Skip and Blank Tape (add to subsequent write time)
- 729 Models II and V = 40.5 ms
- 729 Model IV = 27 ms
- 7330 = 103 ms

Backspace (after Read)	Backspace (after Write)
729 Models II and V = $46 + CN$ ms	729 Models II and V = $52 + CN$ ms
729 Model IV = $33 + CN$ ms	729 Model IV = $37 + CN$ ms
7330 = $428 + CN$ ms	7330 = $435 + CN$ ms

PROCESSING OVERLAP

A-Address

The hundreds position of the A-address of a tape or input-output unit (not 1405 or 1407) instruction is changed from % to @. The symbol is used to signal an overlap operation with character reader, magnetic tape, paper tape, and data transmission units.

Overlap Mode

The following instructions are used when the system is in the overlap mode and card, printer or serial I/O operations are to be performed.

INSTRUCTION	FUNCTION
K\$	Overlap On
K()\$	Overlap On And Branch
K*	Overlap Off
K()*	Overlap Off And Branch
K□	Reset Overlap
K()□	Reset Overlap and Branch

1405 TIMING

TIMINGS (Model 2)	MAX.	AVG.	MIN.
Disk to Disk	800 ms	600 ms	450 ms
Track to Track	250 ms	175 ms	100 ms
Record to Record, same Track	50 ms	25 ms	

International Business Machines Corporation
Data Processing Division
112 East Post Road White Plains, N. Y.

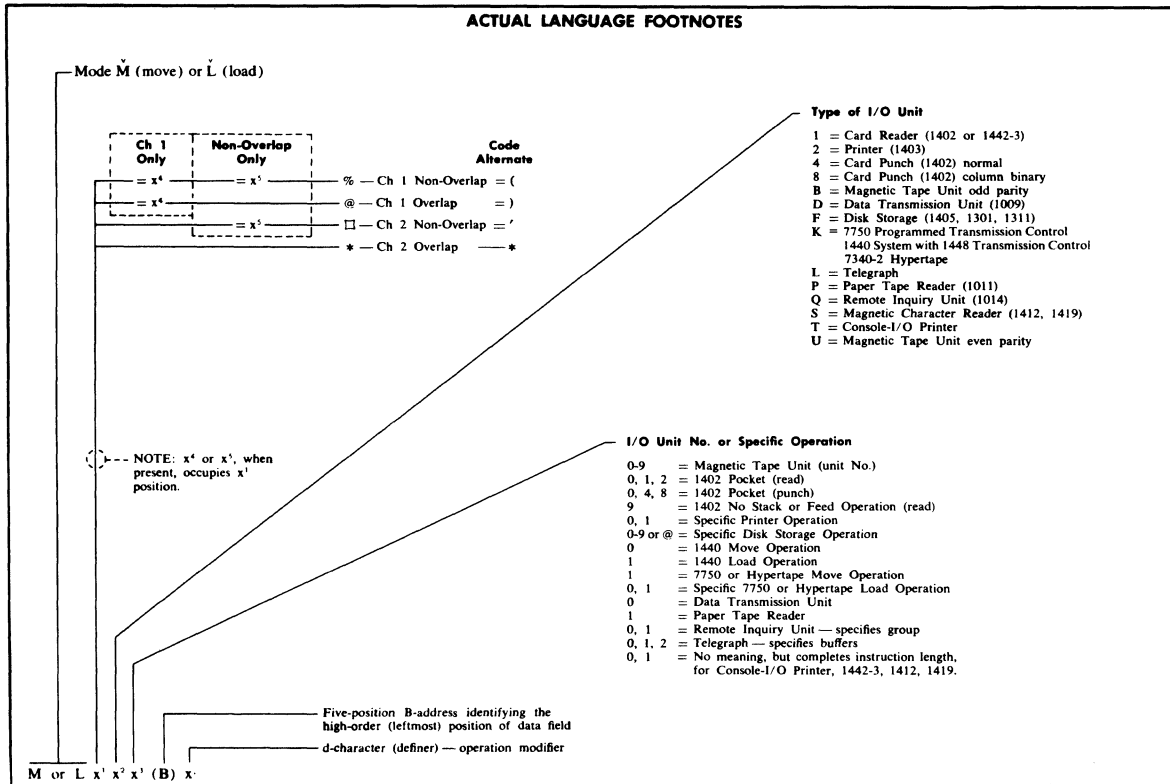
© 1959, 1960, 1961 by International Business Machines Corporation

Form X24-6447-6

Printed in U. S. A.

OPERATION CODE	FUNCTION	MNEMONIC	BCD CODE	CARD CODE	OPERATION CODE	FUNCTION	MNEMONIC	BCD CODE	CARD CODE	INSTRUCTION	FUNCTION	MNEMONIC	BCD CODE	CARD CODE
INPUT-OUTPUT CODES					MISCELLANEOUS OPERATION CODES					MAGNETIC TAPE %UX TAPE UNIT ADDRESS				
1	Read a Card	R	1	1	C	Compare	C	CBA21	12-3	L(%UX)(B)d	Read/Write Tape with Word Marks	LCA		d-modifier, R-Read Tape
2	Write a Line	W	2	2	E	Move Characters and Edit	MCE	CBA41	12-5	M(%UX)(B)d	Read/Write Tape	MCW		W-Write Tape
2 □	Write Word Marks			□ is modifier	F	Control Carriage	CC	CBA42	12-6	M(%CX)(B)R	Read Compressed Tape*			(%CX) is address of tape unit
3	Write-Read	WR	C21	3	H	Store B-Address Register*	SBR	BA8	12-8	P(A)(B)	Move Characters to Record or Group Mark*	MCM	CB421	11-7
4	Punch a Card	P	4	4	K	Select Stacker	SS	CB2	11-2	U(%UX)d	Control Unit	CU	CA4	0-4
4R	Read-Punch Feed*			R is modifier	N	No Operation	NOP	B41	11-5	X(A)(B)	Move and Insert Zeros*	MIZ	CA421	0-7
4(I)R	Read-Punch Feed and Branch*			R is modifier	Q	Store A-Address Register*	SAR	CB8	11-8	INSTRUCTION FUNCTION REMARKS				
5	Read-Punch	RP	C41	5	/	Clear Storage	CS	CA1	0-1	COLUMN BINARY				
6	Write-Punch	WP	C42	6	.	Halt	H	BA821	12-3-8	1C	Read Column Binary	C is Modifier		
6R	Write-Read Punch Feed*			R is modifier	#	Modify Address*	MA	821	3-8	4C	Punch Column Binary	C is Modifier		
6(I)R	Write-Read Punch Feed and Branch*			R is modifier	CHARACTER AT d FOR B(I)d BRANCH					M(A)(B)A	Move and Binary Decode	A is Modifier		
7	Write-Read-Punch	WRP	421	7	d	BRANCH ON	d	BRANCH ON		M(A)(B)B	Move Binary Code	B is Modifier		
8	Start Read Feed*	SRF	8	8	b1	Unconditional	R	Carriage Busy*		M(%BX)(A)R	Read Binary Tape	%BX is Address of tape unit		
9	Start Punch Feed*	SPF	CB1	9	9	Carr. Chan. #9	T	Low Compare B < A*		M(%BX)(A)W	Write Binary Tape			
ARITHMETIC CODES					A	"Last Card" Switch	U	High Compare B > A*		W(I)(B)d	Branch if Bit Equal	BBE is mnemonic		
A	Add	A	BA1	12-1	B	Sense Switch B*	Z	Overflow		DISK STORAGE %FX DISK OPERATION				
S	Subtract	S	CA2	0-2	C	Sense Switch C*	?	Reader Error if I/O Check Stop Switch OFF		M(%F0)(B)R	Seek Disk	B is Disk Address		
?	Zero and Add	ZA	CBA82	12-0	D	Sense Switch D*		Punch Error if I/O Check Stop Switch OFF		M(%FX)(B)R	Read Disk	X can be 1, 2, or 3		
!	Zero and Subtract	ZS	B82	11-0	E	Sense Switch E*	!	Punch Error if I/O Check Stop Switch OFF		M(%FX)(B)W	Write Disk	1 Specifies Single Record		
@	Multiply*	M	C84	4-8	F	Sense Switch F*	≠	Printer Error if I/O Check Stop Switch OFF		L(%FX)(B)R	Read Disk with Word Marks	2 Specifies Full Track		
%	Divide*	D	A84	0-4-8	G	Sense Switch G*	@	Carr. Chan. #12		L(%FX)(B)W	Write Disk with Word Marks	3 Specifies a Write Disk Check operation M(%F3)(B)W		
LOGIC OPERATION CODES					K	End of Reel*	%	Processing Check with Process Check Switch OFF		1407 INQUIRY %TO ADDRESS				
B(I)	Branch	B	BA2	12-2	L	Tape Error*	/	Unequal Compare B ≠ A		M(%T0)(B)R	Read Console Printer	Data from 1407 transferred to B-address		
B(I)d	Branch if Indicator ON			d is modifier	S	Equal Compare B = A*				M(%T0)(B)W	Write Console Printer	Data at B-address transferred to 1407		
B(I)(B)d	Branch if Character is Equal			Contents of B compared to d	CHARACTER AT d FOR MAGNETIC TAPE DISK STORAGE					L(%T0)(B)R	Read Console Printer with Word Marks	Data from 1407 transferred to B-address with Word Marks		
V(I)(B)d	Branch if WM and/or Zone	BWZ	A41	0-5	d	OPERATION	d	BRANCH ON		L(%T0)(B)W	Write Console Printer with Word Marks	Data at B-address transferred to 1407 with Word Marks		
MOVE AND LOAD CODES					B	Backspace Tape Record	N	Access Inoperable		M(%T0)(B)W	Line Space	B is address of a Group Mark with a Word Mark		
D	Move Numerical	MN	BA4	12-4	E	Skip and Blank Tape	V	Read/Write Parity Check or Read Back Check Error		* Special Feature				
L	Load Character to A Word Mark	LCA	B21	11-3	M	Write Tape Mark	W	Wrong-Length Record						
M	Move Characters to A or B Word Mark	MCW	CB4	11-4	R	Rewind Tape	X	Unequal Address Compare						
Y	Move Zone	MZ	CA8	0-8	U	Rewind Tape and Unload	Y	Any Disk Storage Error Condition						
Z	Move Characters and Suppress Zeros	MCS	A81	0-9	CHARACTER AT d FOR 1407 CONSOLE INQUIRY STATION									
?	Set Word Mark	SW	CA821	0-3-8	Q	Inquiry Request	*	Inquiry Clear						
□	Clear Word Mark	CW	CBA84	12-4-8										

IBM 1410 Instruction Card



AUTOCODER FOOTNOTES

Mnemonic Op-Code Suffixes

1 or 2 for Ch
 (#) 1 (or 2) for Ch, but the 1 may be omitted
 w W if WM (load mode)
 O if overlap
 (+I/O) Any I/O statement (prerequisite: priority feature)
 x Undefined and referenced

Operands

a A-address
 aⁱ A-address (AAR does not step)
 b B-address
 i I-address
 d d-character
 c 1 or 2 for Ch
 u I/O unit No.
 cu Ch and unit No.
 0/bl Either 0 or blank
 x X-control field
 (All other characters actual)

Examples:

P(#)w^c 0,b is written as P2WO 0,b to punch a card in the channel 2 1402, using the load and overlap modes, and stack in pocket 0. Equivalent actual-language instruction is \bar{L}^*40 (B) W.

P(#)w^c 0,b is written simply as P 0,b or P1 0,b to punch a card in the channel 1 1402, using the move and non-overlap modes, and stack in pocket 0. Equivalent actual-language instruction is \bar{M}^*40 (B) W.

INDEX TAGS

A, B, OR I ADDRESS				TAG REGISTER
B-BIT OVER HUNDREDS POSITION	A-BIT OVER HUNDREDS POSITION	B-BIT OVER TENS POSITION	A-BIT OVER TENS POSITION	
				NONE
			A	1
		B		2
		B	A	3
	A			4
	A		A	5
	A	B		6
	A	B	A	7
B				8
B			A	9
B		B		10
B		B	A	11
B	A			12
B	A		A	13
B	A	B		14
B	A	B	A	15

STANDARD BCD INTERCHANGE CODE

CHARACTER Report, Program	CARD CODE	BCD CODE (Core Storage)			
b	No Punches	C			
.	12-3-8	B	A	8	2 1
)	12-4-8	C	B	A	8 4 2 1
	12-5-8		B	A	8 4 2 1
<	12-6-8		B	A	8 4 2 1
#	12-7-8	C	B	A	8 4 2 1
+	12	C	B	A	
\$	11-3-8	C	B		8 2 1
*	11-4-8		B		8 4 1
	11-5-8	C	B		8 4 1
:	11-6-8	C	B		8 4 2 1
Δ	11-7-8		B		8 4 2 1
-	11		B		
/	0-1	C	A		
.	0-3-8	C	A	8	2 1
%	0-4-8		A	8	4 1
~	0-5-8	C	A	8	4 1
\	0-6-8	C	A	8	4 2 1
#	0-7-8		A	8	4 2 1
⊕	2-8		A		
# =	3-8			8	2 1
@	4-8	C		8	4 1
:	5-8			8	4 1
>	6-8			8	4 2 1
V	7-8	C		8	4 2 1
?	12-0	C	B	A	8 2 1
A	12-1		B	A	
B	12-2		B	A	2 1
C	12-3	C	B	A	2 1
D	12-4		B	A	4 2 1
E	12-5	C	B	A	4 2 1
F	12-6	C	B	A	4 2 1
G	12-7		B	A	4 2 1
H	12-8		B	A	8 2 1
I	12-9	C	B	A	8 2 1
!	11-0		B	8	2 1
J	11-1	C	B		2 1
K	11-2	C	B		2 1
L	11-3		B		2 1
M	11-4	C	B		4 2 1
N	11-5		B		4 2 1
O	11-6		B		4 2 1
P	11-7	C	B		4 2 1
Q	11-8	C	B		8 4 2 1
R	11-9		B	8	2 1
⊖	0-2-8		A	8	2 1
S	0-2	C	A		2 1
T	0-3		A		2 1
U	0-4	C	A		4 2 1
V	0-5		A		4 2 1
W	0-6		A		4 2 1
X	0-7	C	A		4 2 1
Y	0-8	C	A	8	2 1
Z	0-9		A	8	2 1
#	0	C		8	2 1
1	1			8	2 1
2	2			8	2 1
3	3	C		8	2 1
4	4			8	2 1
5	5	C		8	2 1
6	6	C		8	2 1
7	7			8	2 1
8	8			8	2 1
9	9	C		8	2 1

NOTE: Tape may use even parity.

INDEX REGISTER	STORAGE LOCATIONS
1	00025 through 00029
2	00030 through 00034
3	00035 through 00039
4	00040 through 00044
5	00045 through 00049
6	00050 through 00054
7	00055 through 00059
8	00060 through 00064
9	00065 through 00069
10	00070 through 00074
11	00075 through 00079
12	00080 through 00084
13	00085 through 00089
14	00090 through 00094
15	00095 through 00099

INDEXING EXAMPLE

Modify the A- and B-addresses of this instruction, by tags selecting the same index register:

```
Op Code   A-address   B-address
  V
  A         066F6     077G7
```

Both the A- and B-addresses are tagged by B and A bits over the tens position (F = BA42 and G = BA421). Index register 3 is selected. Because the index factor in IX3 is unsigned, it is added to both addresses.

```
A-address .....066F6..... 06666
IX 3 factor .....         ..... 00081
Effective A-address .....         ..... 06747
```

```
B-address .....077G7..... 07777
IX 3 factor .....         ..... 00081
Effective B-address .....         ..... 07858
```

Effective Instruction: A 06747 07858

VALID ADDRESSES ON 10K OR LARGER SYSTEMS

IBM 1410 Instruction Card

For use with 1410 Processor Operating Systems 1410-PR-108 and -134

File No. 1410-80
Form X22-6740
Printed in U.S.A.

Copies of this and other IBM publications can be obtained through IBM branch offices. Address comments concerning the content of this publication to IBM Corporation, Customer Manuals, Department B98, PO Box 390, Poughkeepsie, New York.

This card, Form X22-6740, obsoletes IBM 1410 Data Processing System Reference Card, Forms X24-6502-1, -2, and -3.

INSTRUCTION	AUTOCODER	ACTUAL
Add (One Field)	A a	A (A)
Add (Two Fields)	A a, b	A (A)(B)
Backspace Tape	BSP cu	U x'x'x' B
Branch if Arithmetic Overflow	BAV i	J (O) Z
Branch if Bit Equal (any bit in b matches a bit in d)	BBE i, b, d	W (I) (B) x
PRINTER		
Branch if Carriage Busy (Ch 1)	BPCB(#) i	J (O) R
Branch if Carriage Busy (Ch 2)	BPCB2 i	J (O) L
Branch if Carriage 9 (Ch 1)	BC9(#) i	J (O) 9
Branch if Carriage 9 (Ch 2)	BC92 i	J (O) !
Branch if Carriage Overflow, 12 (Ch 1)	BCV(#) i	J (O) @
Branch if Carriage Overflow, 12 (Ch 2)	BCV2 i	J (O) □
Branch if Column Binary Card (Ch 1)	BB1 i	J (O) M
Branch if Column Binary Card (Ch 2)	BB2 i	J (O) %
STATUS TEST		
Branch if Any I/O Channel Status Indicator On (Ch 1)	BA1 i	R (O) #
Branch if Any I/O Channel Status Indicator On (Ch 2)	BA2 i	X (O) #
Branch if I/O Unit Not Ready (Ch 1 or 2)	BNR# i	R or X (O) 1
Branch if I/O Unit Busy (Ch 1 or 2)	BCB# i	R or X (O) 2
Branch if I/O Unit Data Check (Ch 1 or 2)	BER# i	R or X (O) 4
Branch if I/O Unit Condition (Ch 1 or 2)	BEF# i	R or X (O) 8
Branch if I/O Wrong Length Record (Ch 1 or 2)	BWL# i	R or X (O) —
Branch if I/O Unit No Transfer (Ch 1 or 2)	BNT# i	R or X (O) b
Branch if Any On in Plural Indicator Test (by d-char of more than one bit less than all bits, such as #)	BEX# i, d	R or X (O) x
Branch if Character Equal (b = d)	BCE i, b, d	B (O) (B) x
Branch if Compare Equal	BE i	J (O) S
Branch if Compare High (B greater than A)	BH i	J (O) U
Branch if Compare Low (B less than A)	BL i	J (O) T
Branch if Compare Unequal	BU i	J (O) /
Branch if Divide Overflow	BDV i	J (O) W
Branch if Inquiry Request (Ch 1)	BNOQ(#) i	J (O) Q
Branch if Inquiry Request (Ch 2)	BNO2 i	J (O) *
Branch if Outquiry Request (Ch 1)	BOOQ(#) i	J (O) N
Branch if Outquiry Request (Ch 2)	BOO2 i	J (O) #
Branch if Overlap in Process (Ch 1)	BOL1 i	J (O) I
Branch if Overlap in Process (Ch 2)	BOL2 i	J (O) 2
PRIORITY		
Branch if Attention Priority Request (Ch 1)	BXPRI i	Y (O) A
Branch if Attention Priority Request (Ch 2)	BXPRI2 i	Y (O) B
Branch if Attention Signal (Ch 1) 1440	BXPRI i	Y (O) A
Branch if Attention Signal (Ch 2) 1440	BXPRI2 i	Y (O) B
Branch if Inquiry Priority Request (Ch 1)	BIPRI(#) i	Y (O) Q
Branch if Inquiry Priority Request (Ch 2)	BIPRI2 i	Y (O) *
Branch if I/O Unit Priority Request (Ch 1)	BUPRI(#) i	Y (O) U
Branch if I/O Unit Priority Request (Ch 2)	BUPRI2 i	Y (O) F
Branch if Outquiry Priority Request (Ch 1)	BOPRI(#) i	Y (O) N
Branch if Outquiry Priority Request (Ch 2)	BOPRI2 i	Y (O) #
Branch if Overlap Priority Request (Ch 1)	BOPRI i	Y (O) I
Branch if Overlap Priority Request (Ch 2)	BOPRI2 i	Y (O) 2
Branch if Priority Request Indicator On (Ch 1) — 7750 or Hypertape	BXPRI i	Y (O) A
Branch if Priority Request Indicator On (Ch 2) — 7750 or Hypertape	BXPRI2 i	Y (O) B
Branch if Seek Priority Request (Ch 1)	BSPRI i	Y (O) S
Branch if Seek Priority Request (Ch 2)	BSPRI2 i	Y (O) T
Branch Unconditionally and Enter Priority Alert Mode	BEPA i	Y (O) E
Branch Unconditionally and Exit From Priority Alert Mode	BXPA i	Y (O) X
Branch if Read Back Check (1405) Ch 1 or 2	BRC# i	R or X (O) @
Branch if Tape Indicator (for CE use)	-----	J (O) K
Branch if Zero Balance	BZ i	J (O) V
Branch if WM Present	BW i, b	V (O) (B) 1
Branch if WM Present, or Zone Bits Absent	BWZ i, b, A	V (O) (B) 3
Branch if WM Present, or Zone Equal A	BWZ i, b, AB	V (O) (B) C
Branch if WM Present, or Zone Equal B	BWZ i, b, B	V (O) (B) L
Branch if Zone Bits Absent	BZN i, b	V (O) (B) 2
Branch if Zone Equal A	BZN i, b, A	V (O) (B) S
Branch if Zone Equal AB	BZN i, b, AB	V (O) (B) B
Branch if Zone Equal B	BZN i, b, B	V (O) (B) K
MAGNETIC CHARACTER READER		
Branch if Late Read Indicator On (1412) Ch 1 or 2	BCLR# i	I or O (O) 1
Branch if Document To Be Read (1419) Ch 1 or 2	BCNR# i	I or O (O) 2
Branch if Document Under Read Head (1419) Ch 1 or 2	BCRC# i	I or O (O) 3
Branch if Read Check Indicator On (1412) Ch 1 or 2	BCAF# i	I or O (O) 4
Branch if Amount Field Valid (1419) Ch 1 or 2	BCPC# i	I or O (O) 5
Branch if Amount Field Indicator On (1412) Ch 1 or 2	BCAN# i	I or O (O) 6
Branch if Account-No. Field Valid (1419) Ch 1 or 2	BCAN# i	I or O (O) 6
Branch if Account-No. Field Indicator On (1412) Ch 1 or 2	BCAN# i	I or O (O) 6
Branch if Transit-Routing Field Valid (1419) Ch 1 or 2	BCTR# i	I or O (O) 7
Branch if Transit-Routing Field Indicator On (1412) Ch 1 or 2	BCTR# i	I or O (O) 7
Branch if Serial-No. Field Valid (1419) Ch 1 or 2	BCDC# i	I or O (O) 8
Branch if Docum't Spacing Chk Indic. On (1412) Ch 1 or 2	BCDC# i	I or O (O) 8
Branch if Incorrect Document Spacing (1419) Ch 1 or 2	BCDC# i	I or O (O) 8
Branch Unconditionally	B i	J (O) blank
PRINTER		
Carriage Control Immediate Skip to 1 (Ch 1 or 2)	CC(#) 1	F or 2 1
Carriage Control Immediate Skip to 2 (Ch 1 or 2)	CC(#) 2	F or 2 2
Carriage Control Immediate Skip to 3-10 (Ch 1 or 2)	CC(#) (3-10)	F or 2 3-0
Carriage Control Immediate Skip to 11 (Ch 1 or 2)	CC(#) #	F or 2 #
Carriage Control Immediate Skip to 12 (Ch 1 or 2)	CC(#) @	F or 2 @
Carriage Control Immediate 1 Space (Ch 1 or 2)	CC(#) J	F or 2 J
Carriage Control Immediate 2 Spaces (Ch 1 or 2)	CC(#) K	F or 2 K
Carriage Control Immediate 3 Spaces (Ch 1 or 2)	CC(#) L	F or 2 L
Carriage Control Skip After Print to 1-5 (Ch 1 or 2)	CC(#) (A-E)	F or 2 A-E
Carriage Control Skip After Print to 6-7 (Ch 1 or 2)	CC(#) (F-G)	F or 2 F-G
Carriage Control Skip After Print to 8-9 (Ch 1 or 2)	CC(#) (H-I)	F or 2 H-I
Carriage Control Skip After Print to 10 (Ch 1 or 2)	CC(#) ?	F or 2 ?
Carriage Control Skip After Print to 11 (Ch 1 or 2)	CC(#) /	F or 2 /
Carriage Control Skip After Print to 12 (Ch 1 or 2)	CC(#) □	F or 2 □
Carriage Control 1 Space After Print (Ch 1 or 2)	CC(#) /	F or 2 /
Carriage Control 2 Spaces After Print (Ch 1 or 2)	CC(#) S	F or 2 S
Carriage Control 3 Spaces After Print (Ch 1 or 2)	CC(#) T	F or 2 T

CLEAR

Clear Storage	CS b	/ (B)
Clear Storage and Branch	CS i, b	/ (O) (B)
Clear Word Mark (One Address)	CW a	□ (A) (B)
Clear Word Mark (Two Addresses)	CW a, b	□ (A) (B)
Compare (b to a)	C a, b	C (A) (B)
Control Unit (creates the unit-control instruction spec. by Autocoder operands, in object deck at compile time)	CU x, b, d	-----
Control 1440	SCM* c, b	M x'K0 (B) C
Control 1440 with WM	SCK* c, b	L x'K1 (B) C
Control Hypertape with WM (normal load)	SCK* c, b	L x'K1 (B) C
Control Hypertape with Actual WM Transfer	SCL* c, b	L x'K0 (B) C
Control Hypertape without WM	SCM* c, b	M x'K1 (B) C
Control 7750 with WM (normal load)	SCK* c, b	L x'K1 (B) C
Control 7750 with Actual WM Transfer	SCL* c, b	L x'K0 (B) C
Control 7750 without WM	SCM* c, b	M x'K1 (B) C
Data Move (see Move)	Mxxxx a, b	D (A) (B) x
Disengage Magnetic Character Reader (Ch 1 or 2)	DCR# b	Por Q D
Divide (a into b)	D a, b	% (A) (B)
Engage Magnetic Character Reader (Ch 1 or 2)	ECR# b	Por Q E
Erase Forward (Skip and Blank Tape)	SKP cu	U x'x'x' E
Halt	H	.
Halt and Branch	H i	.
I/O No Operation, Input Status (wo/w WM)	N(+)/O	M or L x'x'x' (B) Q
I/O No Operation, Output Status (wo/w WM)	N(+)/O	M or L x'x'x' (B) V
Load Unit (creates the load-mode I/O instruction spec. by Autocoder operands, in object deck at compile time)	LU x, b, d	-----

MOVE DATA RIGHT TO LEFT

Move Left Characters and WM Single Position	MLCWS a, b	D (A) (B) 7
Move Left Characters and WM thru 1st A-Field WM	MLCWA a, b	D (A) (B) X
Move Left Characters and WM thru 1st B-Field WM	MLCWB a, b	D (A) (B) P
Move Left Characters and WM thru 1st WM	MLCW a, b	D (A) (B) G
Move Left Characters Single Position	MLCS a, b	D (A) (B) 3
Move Left Characters thru 1st A-Field WM	MLCA a, b	D (A) (B) T
Move Left Characters thru 1st B-Field WM	MLCB a, b	D (A) (B) L
Move Left Characters thru 1st WM	MLC a, b	D (A) (B) C
Move Left Numeric and WM Single Position	MLNWS a, b	D (A) (B) 5
Move Left Numeric and WM thru 1st A-Field WM	MLNWA a, b	D (A) (B) V
Move Left Numeric and WM thru 1st B-Field WM	MLNWB a, b	D (A) (B) W
Move Left Numeric and WM thru 1st WM	MLNW a, b	D (A) (B) E
Move Left Numeric Single Position	MLNS a, b	D (A) (B) 1
Move Left Numeric thru 1st A-Field WM	MLNA a, b	D (A) (B) /
Move Left Numeric thru 1st B-Field WM	MLNB a, b	D (A) (B) J
Move Left Numeric thru 1st WM	MLN a, b	D (A) (B) A
Move Left WM Single Position	MLWS a, b	D (A) (B) 4
Move Left WM thru 1st A-Field WM	MLWA a, b	D (A) (B) U
Move Left WM thru 1st B-Field WM	MLWB a, b	D (A) (B) M
Move Left WM thru 1st WM	MLW a, b	D (A) (B) D
Move Left Zones and WM Single Position	MLZWS a, b	D (A) (B) 6
Move Left Zones and WM thru 1st A-Field WM	MLZWA a, b	D (A) (B) W
Move Left Zones and WM thru 1st B-Field WM	MLZWB a, b	D (A) (B) O
Move Left Zones and WM thru 1st WM	MLZW a, b	D (A) (B) F
Move Left Zones Single Position	MLZS a, b	D (A) (B) 2
Move Left Zones thru 1st A-Field WM	MLZA a, b	D (A) (B) S
Move Left Zones thru 1st B-Field WM	MLZB a, b	D (A) (B) K
Move Left Zones thru 1st WM	MLZ a, b	D (A) (B) B

MOVE DATA LEFT TO RIGHT

Move Right Characters and WM thru 1st A-Field GM-WM	MRCWG a, b	D (A) (B) Δ
Move Right Characters and WM thru 1st A-Field #	MRCWR a, b	D (A) (B) #
Move Right Char. and WM thru 1st A-Field # or GM-WM	MRCWM a, b	D (A) (B) #
Move Right Characters and WM thru 1st WM	MRCW a, b	D (A) (B) Y
Move Right Characters thru 1st A-Field GM-WM	MRCG a, b	D (A) (B) \$
Move Right Characters thru 1st A-Field #	MRCR a, b	D (A) (B) .
Move Right Characters thru 1st A-Field # or GM-WM	MRCM a, b	D (A) (B) .
Move Right Characters thru 1st WM	MRC a, b	D (A) (B) #
Move Right Numeric and WM thru 1st A-Field GM-WM	MRNWG a, b	D (A) (B) J
Move Right Numeric and WM thru 1st A-Field #	MRNWR a, b	D (A) (B) ∞
Move Right Num. and WM thru 1st A-Field # or GM-WM	MRNWM a, b	D (A) (B) J
Move Right Numeric and WM thru 1st WM	MRNW a, b	D (A) (B) V
Move Right Numeric thru 1st A-Field GM-WM	MRNG a, b	D (A) (B) R
Move Right Numeric thru 1st A-Field #	MRNR a, b	D (A) (B) Z
Move Right Numeric thru 1st A-Field # or GM-WM	MRNM a, b	D (A) (B) I
Move Right Numeric thru 1st WM	MRN a, b	D (A) (B) 9
Move Right WM thru 1st A-Field GM-WM	MRWG a, b	D (A) (B) *
Move Right WM thru 1st A-Field #	MRWR a, b	D (A) (B) %
Move Right WM thru 1st A-Field # or GM-WM	MRWM a, b	D (A) (B) □
Move Right WM thru 1st WM	MRW a, b	D (A) (B) @
Move Right Zones and WM thru 1st A-Field GM-WM	MRZWG a, b	D (A) (B) ;
Move Right Zones and WM thru 1st A-Field #	MRZWR a, b	D (A) (B) \
Move Right Zones and WM thru 1st A-Field # or GM-WM	MRZWM a, b	D (A) (B) <
Move Right Zones and WM thru 1st WM	MRZW a, b	D (A) (B) >
Move Right Zones thru 1st A-Field GM-WM	MRZG a, b	D (A) (B) !
Move Right Zones thru 1st A-Field #	MRZR a, b	D (A) (B) #
Move Right Zones thru 1st A-Field # or GM-WM	MRZM a, b	D (A) (B) ?
Move Right Zones thru 1st WM	MRZ a, b	D (A) (B) 0

Move Unit (creates move-mode I/O instruction spec. by Autocoder operands, in object deck at compile time)	MU x, b, d	-----
Move Tape One Record Without Reading (for CE use)	-----	U x'x'x' A
Multiply	M a, b	@ (A) (B)
No Operation	NOP	N
No Operation Word Mark (no-op switch Autocoder pseudo-op — omits WM over next op code)	NOPWM	N
Typical next op	B i	J (I) blank
Prevent Seek Complete (1301 with 7631-3 or -5)	PSC* c, b	M x'F4 (B) or W
Priority Test and Branch (see Branch if Priority)	Bxxxx i	Y (O) x
Punch a Card, Stack in Pocket 0 (wo/w WM)	P(#)* 0, b	M or L x'40 (B) W
Punch a Card, Stack in Pocket 4 (wo/w WM)	P(#)* 4, b	M or L x'44 (B) W
Punch a Card, Stack in Pocket 8 (wo/w WM)	P(#)* 8, b	M or L x'48 (B) W
Punch a Column Binary Card, Stack in Pocket 0	PB#* 0, b	M x'80 (B) W
Punch a Column Binary Card, Stack in Pocket 4	PB#* 4, b	M x'84 (B) W
Punch a Column Binary Card, Stack in Pocket 8	PB#* 8, b	M x'88 (B) W
Read a Card, Stack in Pocket 0 (1402) wo/w WM	R(#)* 0, b	M or L x'10 (B) R
Read a Card, Stack in Pocket 1 (1402) wo/w WM	R(#)* 1, b	M or L x'11 (B) R
Read a Card, Stack in Pocket 2 (1402) wo/w WM	R(#)* 2, b	M or L x'12 (B) R
Read a Card, No Stack or Feed Operation (1402) wo/w WM	R(#)* 9, b	M or L x'19 (B) R
Read and Feed a Card (1442-3) wo/w WM	R(#)* 0, b	M or L x'10 (B) R
Read Console Printer (wo/w WM)	RCPw* b	M or L x'T0 (B) R

TABLE
Look Equal LE a, b
Look Equal or High LEH a, b
Look High LH a, b
Look Low LL a, b
Look Low or Equal LLE a, b
Look Low or High LLH a, b
Look to Any T (A) (B) 7
Look to End T (A) (B) blank
Move Characters and Edit MCE a, b
Move Characters and Suppress Zeros MCS a, b

READ FILE
Read Cylinder (1301) wo/w WM RCYw* c, b
Read Cylinder to End of Cylinder (1301) wo/w WM RCYGw* c, b
Read Disk Sector (1311) wo/w WM RDw* c, b
Read Disk Sector to End of Core or Sector Count (1311) wo/w WM RDGW* c, b
Read Disk Sector Count Overlay (1311) wo/w WM RDVw* c, b
Read Disk Sector Count Overlay to End of Track (1311) wo/w WM RDVGw* c, b
Read Full Track (1405) wo/w WM RDTw* c, b
Read Full Track to End of Core (1405) wo/w WM RDTGW* c, b
Read Full Track Sectors with Addresses (1311) wo/w WM RFTw* c, b
Read Full Track Sectors with Addresses to End of Core or Track (1311) wo/w WM RFTGW* c, b
Read Full Track with Home Address (1301) wo/w WM RHAW* c, b
Read Full Track with Home Address to End of Track (1301) wo/w WM RHAGw* c, b
Read Full Track with Record Addresses (1301) wo/w WM RFTw* c, b
Read Full Track with Record Addresses to End of Track (1301) wo/w WM RFTGW* c, b
Read Full Track without Rec Addresses (1301) wo/w WM RDTw* c, b
Read Full Track without Record Addresses to End of Track (1301) wo/w WM RDTGW* c, b
Read Single Record (1405, 1301) wo/w WM RDw* c, b
Read Single Record to End of Core (1405, 1301) wo/w WM RDGW* c, b
Read Track Record (1311) wo/w WM RDTw* c, b
Read Track Rec to End of Core or Track (1311) wo/w WM RDTGW* c, b
Read Track Record with Address (1311) wo/w WM RTRw* c, b
Read Track Record with Address to End of Track (1311) wo/w WM RTRGW* c, b

REMOTE
Read Data Transmission Unit (wo/w WM) RTDw* c, b
Read Inquiry (1014 unit) wo/w WM ROW* c0, b
Read Inquiry (1014 in 2nd group) wo/w WM RQW* c1, b
Read Paper Tape (wo/w WM) RPTw* c, b
Read Telegraph RL* c0, b
Read Telegraph (1st additional in/out/both buffers) RL* c1, b
Read Telegraph (2nd additional in or out buffer) RL* c2, b
Read Magnetic Character Reader RCR* c, b
Read Magnetic Character Reader and Place WM Over Special Symbols RCRW* c, b

TAPE
Read Tape (wo/w WM) RTw* cu, b
Read Tape to IRG or End of Core (wo/w WM) RTGW* cu, b
Read Tape Binary (odd Parity) wo/w WM RTBW* cu, b
Read Tape Binary to IRG or End of Core (wo/w WM) RTBGW* cu, b
Read 1440 SRM* c, b
Read 1440 with WM SRK* c, b
Read Hypertape with WM (normal load) SRK* c, b
Read Hypertape with Actual WM Transfer SRL* c, b
Read Hypertape without WM SRM* c, b
Read Hypertape to IRG or End of Core with WM (nor. load) SRKG* c, b
Read Hypertape to IRG or End of Core with Actual WM Transfer SRLG* c, b
Read Hypertape to IRG or End of Core without WM SRMG* c, b
Read 7750 with WM (normal load) SRL* c, b
Read 7750 with Actual WM Transfer SRM* c, b
Read 7750 without WM SRKG* c, b
Read 7750 to End of Core with WM (normal load) SRLG* c, b
Read 7750 to End of Core with Actual WM Transfer SRMG* c, b
Read 7750 to End of Core without WM REL* c, b
Rewind RWD cu
Rewind and Unload RWU cu

WRITE FILE
Write Address (1405, for CE use) M x'F4 (B) W
Write Cylinder (1301) wo/w WM WCYw* c, b
Write Cylinder to End of Cylinder (1301) wo/w WM WCYEW* c, b
Write Disk Check (1405, 1301, 1311) wo/w WM WDCw* c, b
Write Disk Check to End of Core (1405, 1301, 1311) wo/w WM WDCEw* c, b
Write Disk Sector (1311) wo/w WM WDw* c, b
Write Disk Sectr to End of Core or Sectr Ct (1311) wo/w WM WDEw* c, b
Write Disk Sector Count Overlay to End of Track (1311) wo/w WM WDVw* c, b
Write Format Track (1301) WDVew* c, b
Write Format Track to End of Core (1301) WFO* c, b
Write Full Track (1405) wo/w WM WFOE* c, b
Write Full Track to End of Core (1405) wo/w WM WDTw* c, b
Write Full Track Sectors With Addresses (1311) wo/w WM WDTGW* c, b
Write Full Track Sectors With Addresses to End of Core or Track (1311) wo/w WM WFTw* c, b
Write Full Track with Home Address (1301) wo/w WM WFTGW* c, b
Write Full Track with Home Address to End of Core (1301) wo/w WM WHAW* c, b
Write Full Track with Record Addresses (1301) wo/w WM WHAGw* c, b
Write Full Track with Record Addresses to End of Core (1301) wo/w WM WFTw* c, b
Write Full Track with Rec Addresses (1301) wo/w WM WFTGW* c, b
Write Full Track without Record Addresses to End of Core (1301) wo/w WM WDTw* c, b
Write Single Record (1405, 1301) wo/w WM WDTGW* c, b
Write Single Record to End of Core (1405, 1301) wo/w WM WDEw* c, b
Write Track Record (1311) wo/w WM WDTw* c, b
Write Track Rec to End of Core or Track (1311) wo/w WM WDTGW* c, b
Write Track Record with Address (1311) wo/w WM WTRw* c, b
Write Track Record with Address to End of Track (1311) wo/w WM WTRGW* c, b

SCAN DATA
Scan Left Single Position SCNLS a, b
Scan Left thru 1st A-Field WM SCNLA a, b
Scan Left thru 1st B-Field WM SCNLB a, b
Scan Left thru 1st WM SCNL a, b
Scan Right thru 1st A-Field GM-WM SCNRG a, b
Scan Right thru 1st A-Field or GM-WM SCNRR a, b
Scan Right thru 1st WM SCNR a, b
Scan Right thru 1st WM SCNR a, b
Scan Disk Equal (1311) wo/w WM SDEw* c, b
Scan Disk High or Equal (1311) wo/w WM SDHw* c, b
Scan Disk Low or Equal (1311) wo/w WM SDLw* c, b
Seek Disk (1405, 1301, 1311) SD* c, b
Select Stacker 0 and Feed (Ch 1 or 2) SSF(0) 0/b1
Select Stacker 1 and Feed (Ch 1 or 2) SSF(1) 1
Select Stacker 2 and Feed (Ch 1 or 2) SSF(2) 2
Select Stacker A Magnetic Character Reader (Ch 1 or 2) SS# A
Select Stacker B Magnetic Character Reader (Ch 1 or 2) SS# B
Select Stacker (0-9) Magnetic Char. Reader (Ch 1 or 2) SS# 0-9
Select Stacker R Magnetic Character Reader (Ch 1 or 2) SS# R
Sense 1440 SSM* c, b
Sense 1440 with WM SSK* c, b
Sense Hypertape with WM (normal load) SSK* c, b
Sense Hypertape with Actual WM Transfer SSL* c, b
Sense Hypertape without WM SSM* c, b
Sense 7750 with WM (normal load) SSK* c, b
Sense 7750 with Actual WM Transfer SSL* c, b
Sense 7750 without WM SSM* c, b
Set Access Inoperative (1301) SAI* c, b
Set Word Mark (One Address) SW a
Set Word Mark (Two Addresses) SW a, b
Skip and Blank Tape SKP cu
Store A-address Register SAR a'
Store B-address Register SBR a'
Store E-address Register SER a'
Store F-address Register SFR a'
Store File Address (1311) wo/w WM SFAw* c, b
Store Time Clock STC a'
Subtract (One Field) S a
Subtract (Two Fields) S a, b
Table Lookup (see Lookup) Lxx a, b
Test and Branch (see Branch) Bxx i
Unit Control (see Backspace, Skip, Write V, Rewind) xxx cu
Write a Line (1403) W(1403) b
Write a Line, WM Create Blanks in Printing W(#)W* b
Write Console Printer (wo/w WM) WCPw* b

REMOTE
Write Data Transmission Unit (wo/w WM) WTDw* c, b
Write Inquiry Response (1014 unit) wo/w WM WQW* c0, b
Write Inquiry Response (1014 in 2nd group) wo/w WM WQW* c1, b
Write Telegraph WL* c0, b
Write Telegraph (1st additional in/out/both buffers) WL* c1, b
Write Telegraph (2nd additional in or out buffer) WL* c2, b

TAPE
Write Tape (wo/w WM) WTW* cu, b
Write Tape to End of Core (wo/w WM) WTEw* cu, b
Write Tape Binary (odd parity) wo/w WM WTBw* cu, b
Write Tape Binary (odd parity) to End of Core (wo/w WM) WTBew* cu, b
Write Tape Mark WTM cu
Write Printer (see Write a Line) W(1403) b
Write Word Marks As 1's (1403) W(#)' b
Write 1440 SWM* c, b
Write 1440 with WM SWK* c, b
Write Hypertape with WM (normal load) SWK* c, b
Write Hypertape with Actual WM Transfer SWL* c, b
Write Hypertape without WM SWM* c, b
Write Hypertape to End of Core with WM (normal load) SWKE* c, b
Write Hypertape to End of Core with Actual WM Transfer SWLE* c, b
Write Hypertape to End of Core without WM SWME* c, b
Write 7750 with WM (normal load) SWK* c, b
Write 7750 with Actual WM Transfer SWL* c, b
Write 7750 without WM SWM* c, b
Write 7750 to End of Core with WM (normal load) SWKE* c, b
Write 7750 to End of Core with Actual WM Transfer SWLE* c, b
Write 7750 to End of Core without WM SWME* c, b
Zero and Add (One Field) ZA a
Zero and Add (Two Fields) ZA a, b
Zero and Subtract (One Field) ZS a
Zero and Subtract (Two Fields) ZS a, b

IBM 1440 Data Processing System Reference Card

DECLARATIVE OPERATIONS table with columns Mnemonic, Description, Mnemonic, Description. Includes DA, DC, DCW, CTL, END, EX, JOB.

INDEXING table with columns Index Number, Autocoder Notation, Storage Locations, Storage Address, Tens Position, Zone Punch. Includes X1, X2, X3.

Definitions for L1, L11, L1:, L1r, L1w, BI, A, B, A1, B1, dbb, N1, L1.

INPUT-OUTPUT OPERATIONS (OTHER THAN DISK) table with columns Mnemonic Op Code, Description, Op Code, A/I-Address, B-Address, d-character, WM's Req., Registers A, B, Remarks.

ARITHMETIC OPERATIONS table with columns A, S, ZA, ZS, D, M and their respective descriptions and register effects.

DATA CONTROL OPERATIONS table with columns MCE, MCS, MLC, MLCWA, MLNS, MLZS, MRCM and their descriptions.

LOGIC OPERATIONS table with columns B, BAV, BBE, BCE, BCV, BC9, BE, BH, BL, BLC, BPB, BSS, BU, BW, BWZ, BIN and their descriptions.

Mnemonic Op Code	Description	Op Code	A/I-Address	B-Address	d-character	WM's Req.	Registers		Remarks
							A	B	
BIN	Branch if Access Inoperable	B	XXX		N†		BI	dbb, Blank or (NSI)**	Disk in Not Ready Status
BIN	Branch if Disk Error	B	XXX		V†				Read/Write Parity or, WDC Error
BIN	Branch if Wrong-Length Record (Disk)	B	XXX		W†				000 Sector Count and $\frac{1}{2}$ Do Not Correspond
BIN	Branch if Access Busy	B	XXX		\†				Disk Drive in Busy Status
BIN	Branch if Unequal Address Compare (Disk)	B	XXX		X†				Sector Address on Disk \neq Sector Address in Core
BIN	Branch if Reader Error	B	XXX		?†				I/O Check Stop Switch Must Be Off
BIN	Branch if Punch Error	B	XXX		I†				I/O Check Stop Switch Must Be Off
BIN	Branch if Printer Error	B	XXX		‡†				I/O Check Stop Switch Must Be Off
BIN	Branch if Processing Check	B	XXX		%†				Process Check Stop Switch Must Be Off
C	Compare	C	XXX	XXX		B - L ₆ + 1	A - L ₆	B - L ₆	B-Field Word Mark Determines Length of Compare
MISCELLANEOUS OPERATIONS									
CC	Control Carriage	F			d†		dbb	dbb	If Carriage Already in Motion, Program Stops
CS	Clear Storage (and Branch)	/	XXX	(XXX)			A or (BI)	X00-1 or (Blank or NSI)**	Clears Left from A- or B-Address to Nearest Hundreds Position
CW	Clear Word Mark	□	XXX	(XXX)			A - 1	A - 1 (B - 1)	WM's Cleared at A- and B-Addresses
H	Halt (and Branch)	.	(XXX)	(XXX)			A _n or (BI)	B _n or (Blank or NSI)**	Press Start Key to Resume. I-Address Gives Auto Branch
MA	Modify Address*	#	XXX	(XXX)			A - 3	B - 1 or B - 3	Three Character Field on Systems > Than 4K
NOP	No Operation	N	(XXX)	(XXX)			A _p	B _p	Program Operation Resumes at Next Op Code
SAR	Store A-Address* Register	Q	XXX				A - 3	A _p	Stores Contents of A-AR in A-Field
SBR	Store B-Address* Register	H	XXX	(XXX)			A - 3	B _p (B)	Store Contents of B-AR in A-Field
SS	Select Stacker*	K			2†		dbb	dbb	Give before Inst. that Ejects Card. Standard on Model 2 1442
SW	Set Word Mark	,	XXX	(XXX)			A - 1	A - 1 (B - 1)	WM's Set at A- and B-Addresses
DISK OPERATIONS									
SDE	Scan Disk Equal*	M	%F8	XXX	W		B + 6	B + 11 + L _F	Records Must Be in Sector Format. Test Result with BE, BH, BL, Instruction. $\frac{1}{2}$ Must Be Set at Right of Search Argument.
SDEW	Scan Disk Equal With Word Marks*	L	%F8	XXX	W		B + 6	B + 11 + L _F	
SDH	Scan Disk High,* Equal	M	%F9	XXX	W		B + 6	B + 11 + L _F	
SDHW	Scan Disk High,* Equal with Word Marks	L	%F9	XXX	W		B + 6	B + 11 + L _F	
SDL	Scan Disk Low, Equal*	M	%F7	XXX	W		B + 6	B + 11 + L _F	
SDLW	Scan Disk Low, Equal* with Word Marks	L	%F7	XXX	W		B + 6	B + 11 + L _F	
RD	Read Disk Sector(s)	M	%F1	XXX	R		B + 6	B + 11 + N _s L _s	Complete When SC = 000
RDCO	Read Disk with Sector Count Overlay	M	%F5	XXX	R		B + 6	B + 8 + N _s L _s	Mult. Sect. Count Field in First Record Read Determines # Sectors Read
RDCOW	Read Disk with Sector Count Overlay with Word Marks	L	%F5	XXX	R		B + 6	B + 8 + N _s L _s	See RDCO
RDT	Read Disk Track Sectors with Addresses	M	%F6	XXX	R		B + 9	B + 11 + 2120	Reads 2120 chars. and Sector Addresses
RDTA	Read Disk Track Record with Address*	M	%F@	XXX	R		B + 9	B + 11 + 2986	Reads 2986 Chars.
RDTAW	Read Disk Track Record with Address and Word Marks*	L	%F@	XXX	R		B + 9	B + 11 + 2688	Reads 2688 Chars.
RDTR	Read Disk Track Record*	M	%F2	XXX	R		B + 6	B + 11 + 2980	Reads 2980 Chars.
RDTRW	Read Disk Track Record with Word Marks*	L	%F2	XXX	R		B + 6	B + 11 + 2682	Reads 2682 Chars.
RDTW	Read Disk Track Sectors with Addresses and Word Marks	L	%F6	XXX	R		B + 9	B + 11 + 1920	Reads 1920 Chars. and Sect. Addresses
RDW	Read Disk Sector(s) with Word Marks	L	%F1	XXX	R		B + 6	B + 11 + N _s L _s	See RD
SD	Seek Disk	M	%F0	XXX	R		B + 6	B + 7	Seeks Specified Cylinder
WD	Write Disk Sector(s)	M	%F1	XXX	W		B + 6	B + 11 + N _s L _s	Complete when SC = 000
WDC	Write Disk Check	M	%F3	XXX	W		Depends on Previous Opn.		Data in Core Comp. with Data Written on Disk
WDCO	Write Disk with Sector Count Overlay	M	%F5	XXX	W		B + 6	B + 8 + N _s L _s	Mult. Sect. Count Field in First Record Determines # Sectors Written
WDCOW	Write Disk with Sector Count Overlay with Word Marks	L	%F5	XXX	W		B + 6	B + 8 + N _s L _s	See WDCO
WDCW	Write Disk Check with Word Marks	L	%F3	XXX	W		Depends on Previous Opn.		See WDC
WDT	Write Disk Track Sectors with Addresses	M	%F6	XXX	W		B + 9	B + 11 + 2120	Writes 2120 Chars. and Sect. Addresses
WDTA	Write Disk Track Record with Address*	M	%F@	XXX	W		B + 9	B + 11 + 2986	Writes 2986 Chars.
WDTAW	Write Disk Track Record with Address and Word Marks*	L	%F@	XXX	W		B + 9	B + 11 + 2688	Writes 2688 Chars.
WDTR	Write Disk Track Record*	M	%F2	XXX	W		B + 6	B + 11 + 2980	Writes 2980 Chars.
WDTRW	Write Disk Track Record with Word Marks*	L	%F2	XXX	W		B + 6	B + 11 + 2682	Writes 2682 Chars.
WDTW	Write Disk Track Sectors with Address and Word Marks	L	%F6	XXX	W		B + 9	B + 11 + 1920	Writes 1920 Chars. and Sect. Addresses
WDW	Write Disk Sector(s) with Word Marks	L	%F1	XXX	W		B + 6	B + 11 + N _s L _s	See WD

* Special feature
 ** If store-address-register function is installed
 † d-character must be coded in operand portion of instruction
 (XXX) Address not required for some formats of instruction

SWITCH AND INDICATOR CODES

Q	Q ₁	Indicator
0	1	Program Switch 1
0	2	Program Switch 2
0	3	Program Switch 3
0	4	Program Switch 4
0	6	Read Check Indicator*
0	7	Write Check Indicator*
0	9	Last Card Indicator
1	1	High-Positive Indicator
1	2	Equal-Zero Indicator
1	3	High-Positive or Equal-Zero Indicator
1	4	Overflow Check Indicator
1	6	Man-Even Check Indicator*
1	7	Man-Odd Check Indicator*
1	9	Any Data Check

* Will Cause 19 to be ON.

COMPARE RESULTS

Condition (Algebraic)	High/Positive	Equal/Zero
P Greater Than Q	ON	OFF
P Less Than Q	OFF	OFF
P Equal to Q	OFF	ON

P = Data in Field at P Address
Q = Data in Field at Q Address

INPUT/OUTPUT DEVICE CODES

Q	Q ₁	Device
0	1	Typewriter — 10 char/sec
0	2	Tape Punch — 15 char/sec
0	3	Paper Tape Reader — 180 char/sec
0	4	Card Punch — 125 cpm
0	5	Card Reader — 250 cpm

CORE STORAGE TABLE AREAS

Address	Area
0000-0099	Control Area
0000-0099	Product Area
0010-0099	Multiply Table
0000-0099	Add Table

TYPEWRITER CONTROL CODES

Q ₁	Space
1	Space
2	Return Carriage
8	Tabulate

BIT CODE, DECIMAL DIGITS 0-9

	C	F	8	4	2	0
0	1	0	0	0	0	0
1	0	0	0	0	0	1
2	0	0	0	0	1	0
3	1	0	0	0	1	1
4	0	0	0	1	0	0
5	1	0	0	1	0	1
6	1	0	0	1	1	0
7	0	0	0	1	1	1
8	0	0	1	0	0	0
9	1	0	1	0	0	1

SIGN CONTROL CHART

Sign of P Field	ADD			SUBTRACT		
	True	Comp	True	True	Comp	True
Sign of Q Field	-	-	-	-	-	-
Stored P Field Sign	-	-	-	-	-	-
True or Complement Add Q Field	True	Comp	True	True	Comp	True
Recomplement Answer if Q Field Value is Greater Than P Field Value	X	X		X		X
Resulting Sign of P Field (Change on Recomplement)	-	-	-	-	-	-

IBM

International Business Machines Corporation
Data Processing Division
112 East Post Road, White Plains, New York

Form 526-6872-3
Printed in U.S.A.

IBM Reference Card | 1620 Data Processing System

Mnemonic	Code	Instruction	Operation	Time
A	21	Add	F _p + F _q replaces F _p	160 + 80 D ₁ , basic time 80 D ₁ , recomp. time ¹
AM	11	Add (I)	F _p + Q replaces F _p	160 + 80 D ₁ , basic time 80 D ₁ , recomp. time ¹
B	49	Branch	Do I _p	200
BB	42	Branch Back	Do I _p	200
BD	43	Branch on Digit	If d ₀ not zero, do I _p	200 No Branch 240 Branch
BI	46	Branch Indicator	If i ₀ on, do I _p	160 No Branch 200 Branch
BNF	44	Branch No Flag	If no i ₀ , do I _p	200 No Branch 240 Branch
BNI	47	Branch No Indicator	If i ₀ off, do I _p	160 No Branch 200 Branch
BNR	45	Branch No Record Mark	If no r ₀ , do I _p	200 No Branch 240 Branch
BT	27	Branch & Transmit	Save A _p , F _p to L _p - 1, do I _p	200 + 40D ₁
BTM	17	Branch & Transmit (I)	Save A _p , Q to L _p - 1, do I _p	200 + 40D ₁
C	24	Compare	F _p compared with F _q	200 + 80D ₁ , Unlike signs 160 + 80D ₁ , Like signs
CF	33	Clear Flag	Remove f from L _p	200
CM	14	Compare (I)	F _p compared with Q	200 + 80D ₁ , Unlike signs 160 + 80D ₁ , Like signs
D	29	Divide	Product Area (00000-00099) = F _q	160 + 520D ₁ , Q ₁ + 740Q ₁ Average quotient digit 4.5
DM	19	Divide (I)	Product Area (00000-00099) = Q	160 + 520D ₁ , Q ₁ + 740Q ₁ Average quotient digit 4.5
DN	35	Dump Numerically	1/Q ₀ writes from L _p to 999, 39, 999 or 59, 999	§
H	48	Halt	Stop	160
K	34	Control	Do Q ₁ on i/o ₀	f
LDD	28	Load Dividend	F _q to L _p	460 + 40D ₁
LDM	18	Load Dividend (I)	Q to L _p	460 + 40D ₁
M	23	Multiply	F _p x F _q (result at 00099)	560 + 40D ₁ + 168D ₁ , D ₀
MF	71	Move Flag	i ₀ to L _p	240
MM	13	Multiply (I)	Q x F _p (result at 00099)	560 + 40D ₁ + 168D ₁ , D ₀
NOP	41	No Operation	Go to A _n	160
RA	37	Read Alphanumerically	1/Q ₀ reads at L _p - 1	§ Except Card i/o (3.4 ms)
RN	36	Read Numerically	1/Q ₀ reads at L _p	§ Except Card i/o (3.4 ms)
S	22	Subtract	F _p - F _q replaces F _p	160 + 80D ₁ , basic time 80 D ₁ , recomp. time ¹
SF	32	Set Flag	Place f at L _p	200
SM	12	Subtract (I)	F _p - Q replaces F _p	160 + 80D ₁ , basic time 80 D ₁ , recomp. time ¹
TD	25	Transmit Digit	d ₀ to L _p	200
TDM	15	Transmit Digit (I)	Q ₁ to L _p	200
TF	26	Transmit Field	F _p to L _p	160 + 40D ₁
TFM	16	Transmit Field (I)	Q to L _p	160 + 40D ₁
TNF	73	Transfer Numerical Fill	F _q to F _p	160 + 40 D ₁
TNS	72	Transfer Numerical Strip	F _p to F _q	160 + 40 D ₁
TR	31	Transmit Record	R ₀ to L _p	160 + 40D ₁
WA	38	Write Alphanumerically	1/Q ₀ writes from L _p - 1	§ Except Card i/o (3.4 ms)
WN	38	Write Numerically	1/Q ₀ writes from L _p	§ Except Card i/o (3.4 ms)

Symbols and Definitions for "Operation" Column

P	P part of instruction	R ₀	Record defined by Q
Q	Q part of instruction	1/Q ₀	1/Q ₀ defined by Q ₀
F _p	Field defined by P	d ₀	Digit at L ₀
F _q	Field defined by Q	i ₀	Flag bit at L ₀
I _p	Instruction defined by P	f	Flag bit
i _p	Saved instruction	r ₀	Record mark at L ₀
L _p	Location defined by P	i ₀	Indicator defined by Q ₀ , Q ₁
L _q	Location defined by Q	A ₀	Address of next seq. instr.

Symbols and Definitions for "Time" Column

D ₁	Number of digits, including high-order zero, in the field at P.
D ₂	Number of digits, including high-order zero, in the field at Q.
D ₃	Number of digits, including high-order zero, in the Q part of the instruction.
D ₄	Number of digits, including high-order zero, in the dividend.
Q ₁	Number of digits, including high-order zero, in the quotient.
D ₅	Number of digits, including high-order zero, in the divisor.
D ₆	Number of positions compared until a digit other than zero is detected in either field.

All times are in microseconds (1 microsecond = 1/1,000,000 second).

1 Immediate.
 † If signs initially unlike and numerical value of Q data greater than P data.
 ‡ Depends on speed of i/o device and number of characters involved.
 § Depends on control function and speed of i/o device.

1620 CHARACTER CODING

Alphanumeric Character	Input		Core Storage		Output	
	Typewriter	Type	Alpha	Num	Typewriter	Type
(Blank)	(Space)	C	(Blank)	C	(Space)	C
(Period)	.	X0821 12, 3, 8	.	C	.	X0821 12, 3, 8
1	1	X0C84 12, 4, 8	1	C	1	X0C84 12, 4, 8
+	+	X0C 12	+	C	+	X0C 12
\$	\$	X0821 11, 3, 8	1	3	\$	X0821 11, 3, 8
*	*	X84 11, 8, 4	1	4	*	X84 11, 8, 4
- (Hyphen)	-	X 11	2	C	-	X 11
/	/	OC1 0, 1	2	F	/	OC1 0, 1
(Comma)	(OC821 0, 3, 8	2	3	(OC821 0, 3, 8
((084 0, 4, 8	2	4	(084 0, 4, 8
=	=	821 3, 8	3	3	=	821 3, 8
0	0	C84 4, 8	3	4	0	C84 4, 8
A-1	A-1	X0, 1-9 12, 1-9	4	1-9	A-1	X0, 1-9 12, 1-9
0 (-)	(None)	(None)	11, 0	5	C	X 11, 0
J-R	J-R	X, 1-9 11, 1-9	5	1-9	J-R	X, 1-9 11, 1-9
1-9 (-)	J-R	X, 1-9 11, 1-9	5	1-9	J-R	X, 1-9 11, 1-9
S-Z	S-Z	0, 2-9 0, 2-9	6	2-9	S-Z	0, 2-9 0, 2-9
0 (+)	0	0 or 12, 0	7	C	0	0
1-9 (+)	1-9	1-9 1-9	7	1-9	1-9	1-9
1	1	082 0, 2, 8	7	C	082 (Stop)	EOL 0, 2, 8
Numerical Character						
(Blank)	(Space)	C	(Blank)	C	0	0
0 (+)	0	0	0	0	0	0
0 (-)	0	X, X0C 11, 0	F	0	X	11, 0
1-9 (+)	1-9	1-9 1-9	1-9	1-9	1-9	1-9
1-9 (-)	1-9	X, 1-9 11, 1-9	1-9	1-9	X, 1-9 11, 1-9	1-9
1	1	082 0, 2, 8	C82	1	EOL (W/N) 0, 2, 8	082 (DN)
Num Blank +	+	C84 4, 8	C84	0	C84 (Blank)	(Blank)

* For Card Format Use Only

ALLOWABLE INDIRECT ADDRESSING

Arithmetic Instructions:	MNEMONIC	CODE	P & Q	F
Add Immediate	AM	11	X	X
Subtract Immediate	SM	12	X	X
Multiply Immediate	MM	13	X	X
Load Dividend	LD	28	X	X
Load Dividend Immediate	DM	18	X	X
Divide	D	19	X	X
Divide Immediate	DM	19	X	X
Internal Data Transmission Instructions:				
Transmit Digit	TD	25	X	X
Transmit Digit Immediate	TDM	15	X	X
Transmit Field	TF	26	X	X
Transmit Field Immediate	TFM	16	X	X
Transmit Record	TR	31	X	X
Transfer Numerical Strip	TNS	72	X	X
Transfer Numerical Fill	TNF	73	X	X
Logic (Compare and Branch) Instructions:				
Compare	C	24	X	X
Compare Immediate	CM	14	X	X
Branch	B	49	X	X
Branch No Flag	BNF	44	X	X
Branch No Record Mark	BNR	45	X	X
Branch On Digit	BD	43	X	X
Branch Indicator	BI	46	X	X
Branch No Indicator	BNI	47	X	X
Branch and Transmit	BT	27	X	X
Branch and Transmit Immediate	BTM	17	X	X
Branch Back	BB	42		
Input-Output Instructions:				
Read Numerically	RN	36	X	X
Write Numerically	WN	38	X	X
Dump Numerically	DN	35	X	X
Read Alphanumerically	RA	37	X	X
Write Alphanumerically	WA	39	X	X
Control	K	34		
Program Control Instructions:				
Set Flag	SF	32	X	X
Clear Flag	CF	33	X	X
Move Flag	MF	71	X	X
Halt	H	48		
No Operation	NOP	41		

SIGNIFICANCE OF P & Q ADDRESS

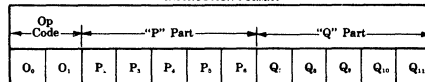
Operation Code	P Address	Q Address
11—Add (1)*	Memory address of units position of Augend.	Q ₀ of instruction is units position of Addend.
21—Add	Same as Code 11.	Memory address of units position of Addend.
12—Subtract (1)*	Memory address of units position of Minuend.	Q ₀ of instruction is units position of Subtrahend.
22—Subtract	Same as Code 12.	Memory address of units position of Subtrahend.
13—Multiply (1)*	Memory address of units position of Multiplier.	Q ₀ of instruction is units position of Multiplier.
23—Multiply	Same as Code 13.	Memory address of units position of Multiplier.
14—Compare (1)*	Memory address of units position of the field to which another field is to be compared.	Q ₀ of instruction is units position of the field to be compared with the field at the P address.
24—Compare	Same as Code 14.	Memory address of units position of the field to be compared with the field at the P address.
15—Transmit Digit (1)*	Memory address to which single digit is to be transmitted.	Q ₀ of instruction is the single digit to be transmitted.
25—Transmit Digit	Same as Code 15.	Memory address of single digit to be transmitted.
16—Transmit Field (1)*	Memory address to which units position of field is to be transmitted.	Q ₀ of instruction is the units position of the field to be transmitted.
26—Transmit Field	Same as Code 16.	Memory address of units position of the field to be transmitted.
17—Branch and Transmit (1)*	"P minus one" is the memory address to which the units position of the Q field is to be transmitted. "P" is the memory address of the high-order digit of the next instruction to be interpreted and executed.	Q ₀ of instruction is the units position of the field to be transmitted.
18—Load Dividend (1)*	Memory address in Product Area to which units position of field (Dividend) is to be transmitted.	Q ₀ of instruction is the units position of the field (Dividend) to be transmitted.
28—Load Dividend	Same as Code 18.	Memory address of the units position of the field (Dividend) to be transmitted.
19—Divide (1)*	Memory address at which first subtraction of the Divisor is to occur.	Q ₀ of instruction is the units position of the Divisor.
29—Divide	Same as Code 19.	Memory address of units position of Divisor.
31—Transmit Record	Memory address to which high-order position of record is to be transmitted.	Memory address of high-order position of the record to be transmitted.
32—Set Flag	Memory address at which flag bit is to be placed.	Not used.
33—Clear Flag	Memory address from which flag bit is to be cleared.	Not used.
34—Control	Not used.	Q ₀ and Q ₁ specify input-output device. Q ₁ specifies control function to be performed.
35—Dump Numerically	Memory address from which first numerical character is to be written.	Q ₀ and Q ₁ specify output device.
36—Read Numerically	Memory address at which first numerical character is to be stored.	Q ₀ and Q ₁ specify input device.
37—Read Alphanumerically	Memory address at which numerical digit of first character is to be stored. (Zone digit of first character will be stored at "P minus one.")	Same as Code 36.
38—Write Numerically	Memory address from which first numerical character is to be written.	Same as Code 35.
39—Write Alphanumerically	Memory address for numerical digit of first character to be written. (Zone digit of first character is at "P minus one.")	Same as Code 35.
41—No Op	Not used.	Not used.
42—Branch Back	Not used.	Not used.
43—Branch On Digit	Memory address of the high-order digit of the next instruction to be interpreted and executed, if Branch occurs. (Not used if Branch does not occur.)	Memory address to be interrogated for the presence of a significant digit (not a zero).
44—Branch No Flag	Same as Code 43.	Memory address to be interrogated for the presence of a flag bit.
45—Branch No Record Mark	Same as Code 43.	Memory address to be interrogated for the presence of a Record Mark character.
46—Branch Indicator	Same as Code 43.	Q ₀ and Q ₁ digits specify program switch or indicator to be interrogated for status.
47—Branch No Indicator	Same as Code 43.	Same as Code 46.
48—Halt	Not used.	Not used.
49—Branch	Memory address of the high-order digit of the next instruction to be interpreted and executed.	Not used.
71—Move Flag	Memory address of flag to be moved.	Memory address to which flag is to be moved.
72—Transfer Numerical Strip	Memory address of the units position of the alphanumeric field.	Memory address of the units position of the numerical field.
73—Transfer Numerical Fill	Same as Code 72.	Same as Code 72.

* Immediate.

1620 STORAGE REGISTERS

Register	Function
IB-1	Contains address of next instruction if machine is stopped with stop key or halt instruction.
IB-2	Saves return address when BT and BTM instructions are executed.
OB-1	Contains Q address after 1-cycle of an instruction.
OB-2	Contains P address after 1-cycle of an instruction.
OB-3	Retains address of low-order multiplier digit during multiplication.
PR-1	Saves return address when a save key operation occurs.
PR-2	Decrementer for each new multiply digit during multiply.
PR-3	Decrementer for each new multiplicand digit during multiply.
PR-4	Used to add partial product to each multiply cycle result.
MSR	Addresses core storage.
MSR	Receives digits entering or leaving core storage.
MSR	Receives addressed digit entering or leaving core storage.
Digit	Stores partial product during multiplication.
OP	Contains OP code of instruction just executed if machine is stopped with stop key or halt instruction.
Multiplier	Contains multiplier digits during multiply operation.
Sense & Branch	Contains I/O device code during input/output operations. Units positions used to develop each quotient digit during divide operation.
Digit & Branch	On some machines, combines functions of Digit, Sense & Branch Registers.

INSTRUCTION FORMAT



IBM Reference Card 7040-7044 Codes

Form X22-6696-1
Printed in U. S. A.

CHARACTER CODING AND TRANSLATION

CORE STORAGE BYTE	TAPE	OTHER	TYPEWRITER		TYPEWRITER	
			REPORT WRITING GRAPHICS		PROGRAMMING LANGUAGES GRAPHICS	
			BCD	BINARY	BCD	BINARY
000 000	0 001 010	1 001 010	Ø	Blank	Ø	Blank
000 001	1 000 001	0 000 001	1	1	1	1
000 010	1 000 010	0 000 010	2	2	2	2
000 011	0 000 011	1 000 011	3	3	3	3
000 100	1 000 100	0 000 100	4	4	4	4
000 101	0 000 101	1 000 101	5	5	5	5
000 110	0 000 110	1 000 110	6	6	6	6
000 111	1 000 111	0 000 111	7	7	7	7
001 000	1 001 000	0 001 000	8	8	8	8
001 001	0 001 001	1 001 001	9	9	9	9
001 010	0 001 010	0 010 000	Ø	Ø	Ø	Ø
001 011	1 001 011	0 001 011	#	#	=	=
001 100	0 001 100	1 001 100	@	@	!	!
001 101	1 001 101	0 001 101	:	:	:	:
001 110	1 001 110	0 001 110	>	>	>	>
001 111	0 001 111	1 001 111	✓ TM	✓	✓	✓
010 000	0 110 000	1 110 000	£	£	+	+
010 001	1 110 001	0 110 001	A	/	A	/
010 010	1 110 010	0 110 010	B	S	B	S
010 011	0 110 011	1 110 011	C	T	C	T
010 100	1 110 100	0 110 100	D	U	D	U
010 101	0 110 101	1 110 101	E	V	E	V
010 110	0 110 110	1 110 110	F	W	F	W
010 111	1 110 111	0 110 111	G	X	G	X
011 000	1 111 000	0 111 000	H	Y	H	Y
011 001	0 111 001	1 111 001	I	Z	I	Z
011 010	0 111 010	1 111 010	?	?	?	?
011 011	1 111 011	0 111 011
011 100	0 111 100	1 111 100	□	%)	(
011 101	1 111 101	0 111 101	⊂	⊂	⊂	⊂
011 110	1 111 110	0 111 110	<	<	<	<
011 111	0 111 111	1 111 111	‡ GM	‡	‡	‡
100 000	1 100 000	0 100 000	-	-	-	-
100 001	0 100 001	1 100 001	J	J	J	J
100 010	0 100 010	1 100 010	K	K	K	K
100 011	1 100 011	0 100 011	L	L	L	L
100 100	0 100 100	1 100 100	M	M	M	M
100 101	1 100 101	0 100 101	N	N	N	N
100 110	1 100 110	0 100 110	O	O	O	O
100 111	0 100 111	1 100 111	P	P	P	P
101 000	0 101 000	1 101 000	Q	Q	Q	Q
101 001	1 101 001	0 101 001	R	R	R	R
101 010	1 101 010	0 101 010	I	I	I	I
101 011	0 101 011	1 101 011	\$	\$	\$	\$
101 100	1 101 100	0 101 100	•	•	•	•
101 101	0 101 101	1 101 101	⌋	⌋	⌋	⌋
101 110	0 101 110	1 101 110	:	:	:	:
101 111	1 101 111	0 101 111	Δ	Δ	Δ	Δ
110 000	1 010 000	1 000 000	Blank	&	Blank	+
110 001	0 010 001	1 010 001	/	A	/	A
110 010	0 010 010	1 010 010	S	B	S	B
110 011	1 010 011	0 010 011	T	C	T	C
110 100	0 010 100	1 010 100	U	D	U	D
110 101	1 010 101	0 010 101	V	E	V	E
110 110	1 010 110	0 010 110	W	F	W	F
110 111	0 010 111	1 010 111	X	G	X	G
111 000	0 011 000	1 011 000	Y	H	Y	H
111 001	1 011 001	0 011 001	Z	I	Z	I
111 010	1 011 010	0 011 010	‡ RM	‡	‡	‡
111 011	0 011 011	1 011 011
111 100	1 011 100	0 011 100	%	⊂	()
111 101	0 011 101	1 011 101	⊂	⊂	⊂	⊂
111 110	0 011 110	1 011 110	<	<	<	<
111 111	1 011 111	0 011 111	‡	‡	‡	‡

* CORE STORAGE = 000 000 on READ

PRINTER		PRINTER		READ - PUNCH	
"A" CHAIN		"H" CHAIN		IBM CARD CODE	
BCD	BINARY	BCD	BINARY	BCD	BINARY
Ø	Blank	Ø	Blank	Ø	Blank
1	1	1	1	1	1
2	2	2	2	2	2
3	3	3	3	3	3
4	4	4	4	4	4
5	5	5	5	5	5
6	6	6	6	6	6
7	7	7	7	7	7
8	8	8	8	8	8
9	9	9	9	9	9
±	Ø	±	Ø	8-2	0
#	#	=	=	8-3	8-3
@	@	!	!	8-4	8-4
Blank	Blank	Blank	Blank	8-5	8-5
Blank	Blank	Blank	Blank	8-6	8-6
Blank	Blank	Blank	Blank	8-7	8-7
&	≠	+	≠	12	8-2
A	/	A	/	12-1	0-1
B	S	B	S	12-2	0-2
C	T	C	T	12-3	0-3
D	U	D	U	12-4	0-4
E	V	E	V	12-5	0-5
F	W	F	W	12-6	0-6
G	X	G	X	12-7	0-7
H	Y	H	Y	12-8	0-8
I	Z	I	Z	12-9	0-9
&	±	+	±	12-0	0-8-2
.	.	.	.	12-8-3	0-8-6
□	%)	(12-8-4	0-8-4
Blank	Blank	Blank	Blank	12-8-5	0-8-5
Blank	Blank	Blank	Blank	12-8-6	0-8-3
Blank	Blank	Blank	Blank	12-8-7	0-8-7
J	J	J	J	11	11
K	K	K	K	11-1	11-1
L	L	L	L	11-2	11-2
M	M	M	M	11-3	11-3
N	N	N	N	11-4	11-4
O	O	O	O	11-5	11-5
P	P	P	P	11-6	11-6
Q	Q	Q	Q	11-7	11-7
R	R	R	R	11-8	11-8
-	-	-	-	11-9	11-9
-	-	-	-	11-0	11-0
\$	\$	\$	\$	11-8-3	11-8-3
•	•	•	•	11-8-4	11-8-4
Blank	Blank	Blank	Blank	11-8-5	11-8-5
Blank	Blank	Blank	Blank	11-8-6	11-8-6
Blank	Blank	Blank	Blank	11-8-7	11-8-7
Blank	Blank	Blank	Blank	12	12
/	A	/	A	0-1	12-1
S	B	S	B	0-2	12-2
T	C	T	C	0-3	12-3
U	D	U	D	0-4	12-4
V	E	V	E	0-5	12-5
W	F	W	F	0-6	12-6
X	G	X	G	0-7	12-7
Y	H	Y	H	0-8	12-8
Z	I	Z	I	0-9	12-9
‡	&	‡	‡	0-8-2	12-0
.	.	.	.	0-8-3	12-8-3
%	⊂	()	0-8-4	12-8-4
Blank	Blank	Blank	Blank	0-8-5	12-8-5
Blank	Blank	Blank	Blank	0-8-6	12-8-6
Blank	Blank	Blank	Blank	0-8-7	12-8-7

"9"=709, 7090, 7094
Storage Code
"5"=705, 7080 and
BCD Magnetic
Tape Code
"14"=1401, 1410, 1414
Storage and
Buffer Code
"H"=Standard 64
Character IBM
Card Code

INSTRUCTION LISTING-NUMERIC
(All Options)

SYM	NUMERIC	NAME	F	T
TRA	0020	TRANSFER ON REDUNDANCY CHECK, CHANNEL A	X	X
TRCA	0022	TRANSFER ON REDUNDANCY CHECK, CHANNEL A	X	X
TRCB	0023	TRANSFER ON REDUNDANCY CHECK, CHANNEL B	X	X
TRCC	0024	TRANSFER ON REDUNDANCY CHECK, CHANNEL C	X	X
TRCD	0024	TRANSFER ON REDUNDANCY CHECK, CHANNEL D	X	X
TRCE	0026	TRANSFER ON REDUNDANCY CHECK, CHANNEL E	X	X
TEFA	0030	TRANSFER ON END OF FILE, CHANNEL A	X	X
TEFB	0030	TRANSFER ON END OF FILE, CHANNEL B	X	X
TEFC	0031	TRANSFER ON END OF FILE, CHANNEL C	X	X
TEFD	0031	TRANSFER ON END OF FILE, CHANNEL D	X	X
TEFE	0032	TRANSFER ON END OF FILE, CHANNEL E	X	X
TCCA	0060	TRANSFER ON CHANNEL A IN OPERATION	X	X
TCCB	0061	TRANSFER ON CHANNEL B IN OPERATION	X	X
TCCD	0062	TRANSFER ON CHANNEL C IN OPERATION	X	X
TCCD	0063	TRANSFER ON CHANNEL D IN OPERATION	X	X
TCEE	0064	TRANSFER ON CHANNEL E IN OPERATION	X	X
TSX	0074	TRANSFER AND SET INDEX	X	X
TZE	0100	TRANSFER ON ZERO	X	X
TNZ	-0100	TRANSFER ON NO ZERO	X	X
TPL	0120	TRANSFER ON PLUS	X	X
TMI	-0120	TRANSFER ON MINUS	X	X
TOV	0140	TRANSFER ON OVERFLOW	X	X
MPY	0200	MULTIPLY	X	X
VLM	0204	VARIABLE LENGTH MULTIPLY	X	X
DVP	0221	DIVIDE OR PROCEED	X	X
VDP	0225	VARIABLE LENGTH DIVIDE OR PROCEED	X	X
FDP	0241	FLOATING DIVIDE OR PROCEED	X	X
DFDP	0241	DOUBLE PREC FLO DIVIDE OR PROCEED	X	X
FMP	0260	FLOATING MULTIPLY	X	X
UFM	-0260	UNNORMALIZED FLOATING MULTIPLY	X	X
DFMP	0261	DOUBLE PRECISION FLOATING MULTIPLY	X	X
FAD	0300	FLOATING ADD	X	X
UFA	-0300	UNNORMALIZED FLOATING ADD	X	X
DFAD	0301	DOUBLE PRECISION FLOATING ADD	X	X
FSB	0302	FLOATING SUBTRACT	X	X
UFS	-0302	UNNORMALIZED FLOATING SUBTRACT	X	X
DFSB	0303	DOUBLE PRECISION FLOATING SUBTRACT	X	X
ANA	-0320	AND TO ACCUMULATOR	X	X
CAS	0340	COMPARE ACCUMULATOR WITH STORAGE	X	X
LAS	-0340	LOGICAL COMPARE ACCUM WITH STORAGE	X	X
ACL	0361	ADD AND CARRY LOGICAL WORD	X	X
ADD	0400	ADD	X	X
SUB	0420	SUBTRACT	X	X
HPR	0420	HALT AND PROCEED	X	X
CLA	0500	CLEAR AND ADD	X	X
CAL	-0500	CLEAR AND ADD LOGICAL WORD	X	X
ORA	-0501	OR TO ACCUMULATOR	X	X
CLS	0502	CLEAR AND SUBTRACT	X	X
XEC	0522	EXECUTE	X	X
LXA	0534	LOAD INDEX FROM ADDRESS	X	X
LXD	-0534	LOAD INDEX FROM DECREMENT	X	X
LAC	0535	LOAD COMPLEMENT OF ADDRESS IN INDEX	X	X
LDC	-0535	LOAD COMPLEMENT OF DECREMENT IN INDEX	X	X
RCHA	0540	RESET AND LOAD CHANNEL A	X	X
RCHB	0540	RESET AND LOAD CHANNEL B	X	X
DLB	0540.1	LOAD DATA REGISTER AND LOOP, CHANNEL B	X	X
RCHC	0541	RESET AND LOAD CHANNEL C	X	X
DLCC	0541.1	LOAD DATA REGISTER AND LOOP, CHANNEL C	X	X
RCHD	0541	RESET AND LOAD CHANNEL D	X	X
DLDD	0541.1	LOAD DATA REGISTER AND LOOP, CHANNEL D	X	X
RCHE	0542	RESET AND LOAD CHANNEL E	X	X
DLDE	0542.1	LOAD DATA REGISTER AND LOOP, CHANNEL E	X	X
LDO	0560	LOAD M0	X	X
ENL	0564	ENABLE FROM Y	X	X
STO	0600	STORE M0	X	X
STO	0601	STORE LOGICAL WORD	X	X
SLW	0602	STORE ADDRESS	X	X
SFA	0622	STORE DECREMENT	X	X
STL	-0625	STORE INSTRUCTION LOCATION COUNTER	X	X
SXA	0634	STORE INDEX IN ADDRESS	X	X
SXD	-0634	STORE INDEX IN DECREMENT	X	X
SCHB	0640	STORE CHANNEL B	X	X
SCHB	0640.1	STORE DATA REGISTER, CHANNEL B	X	X
SCHC	0641	STORE CHANNEL C	X	X
SCHC	0641.1	STORE DATA REGISTER, CHANNEL C	X	X
SCHD	0641	STORE CHANNEL D	X	X
SCHD	0641.1	STORE DATA REGISTER, CHANNEL D	X	X
SCHD	0642	STORE CHANNEL E	X	X

INSTRUCTION LISTING-NUMERIC
(All Options)

SYM	NUMERIC	NAME	F	T
SDRE	0642.1	STORE DATA REGISTER, CHANNEL E	X	X
SSLB	-0660	STORE SENSE LINES, CHANNEL B	X	X
SSLC	0661	STORE SENSE LINES, CHANNEL C	X	X
SSLD	-0661	STORE SENSE LINES, CHANNEL D	X	X
SSLE	0662	STORE SENSE LINES, CHANNEL E	X	X
PSLB	-0664	PRESENT SENSE LINE, CHANNEL B	X	X
PSLC	0665	PRESENT SENSE LINES, CHANNEL C	X	X
PSLD	-0665	PRESENT SENSE LINES, CHANNEL D	X	X
PSLE	0666	PRESENT SENSE LINES, CHANNEL E	X	X
PAX	0734	PLACE ADDRESS IN INDEX	X	X
PDX	-0734	PLACE DECREMENT IN INDEX	X	X
PAC	0737	PLACE COMPLEMENT OF ADDRESS IN INDEX	X	X
PDC	-0737	PLACE COMPLEMENT OF DECREMENT IN INDEX	X	X
PXA	0794	PLACE INDEX IN ADDRESS	X	X
PXD	-0794	PLACE INDEX IN DECREMENT	X	X
SBT	0760.00001	LOW ORDER BIT TEST	X	X
PBT	-0760.00001	P BIT TEST	X	X
CHS	0760.00002	CHANGE SIGN	X	X
SSP	0760.00003	SET SIGN PLUS	X	X
ENK	0760.00004	ENTER KEYS	X	X
IOI	0760.00009	INPUT/OUTPUT CHECK TEST	X	X
COM	0760.00006	COMPLEMENT MAGNITUDE	X	X
DCT	0760.00012	DIVIDE CHECK TEST	X	X
TRC	0760.00014	RESTORE CHANNEL TRAPS	X	X
SWT	0760.0016N	SENSE SWITCH TEST	X	X
ETTA	-0760.01000	END OF TAPE TEST, CHANNEL A	X	X
ETTB	0760.01000	RESET DATA CHANNEL A	X	X
ETTB	-0760.02000	END OF TAPE TEST, CHANNEL B	X	X
RDCB	0760.02352	RESET DATA CHANNEL B	X	X
ETTC	-0760.03000	END OF TAPE TEST, CHANNEL C	X	X
RDCD	0760.03352	RESET DATA CHANNEL C	X	X
ETTD	-0760.04000	END OF TAPE TEST, CHANNEL D	X	X
RDCD	0760.04352	RESET DATA CHANNEL D	X	X
ETTE	-0760.05000	END OF TAPE TEST, CHANNEL E	X	X
RDEE	0760.05352	RESET DATA CHANNEL E	X	X
RDS	0762	READ SELECT	X	X
LLS	0763	LONG LEFT SHIFT	X	X
LGL	-0763	LOGICAL LEFT SHIFT	X	X
BSR	0764	BACKSPACE RECORD	X	X
LRS	0765	LONG RIGHT SHIFT	X	X
LGR	-0765	LOGICAL RIGHT SHIFT	X	X
WBT	0766	WRITE BLANK TAPE	X	X
ALS	0767	WRITE SELECT	X	X
WEL	0770	WRITE END OF FILE	X	X
ARL	0771	ACCUMULATOR RIGHT SHIFT	X	X
REW	0772	REWIND	X	X
RUN	-0772	REWIND AND UNLOAD	X	X
ROL	-0773	ROTATE M0 LEFT	X	X
AXT	0774	ADDRESS TO INDEX TRUE	X	X
STR	-1000	STORE LOCATION AND TRAP	X	X
TXI	1000	TRANSFER WITH INDEX INCREMENTED	X	X
RPM	-1004	RELEASE PROTECT MODE	X	X
TDOA	-1060	TRANSFER ON CHANNEL A DEVICE IN OP	X	X
SPM	-1160	SET PROTECT MODE	X	X
TRT	-1164	TRANSFER AND RESTORE TRAPS	X	X
TRP	-1165	TRANSFER AND RESTORE PARITY AND TRAPS	X	X
VMA	-1204	VARIABLE LENGTH MULTIPLY AND ACCUM	X	X
CCS	-1341	COMPARE CHARACTER WITH STORAGE	X	X
MIT	-1341.6	STORAGE MINUS TEST	X	X
PLT	-1341.7	STORAGE PLUS TEST	X	X
PDS	-1505	PLACE CHARACTER FROM STORAGE	X	X
SAP	-1610	STORE AND ADD LOGICAL WORD WITH PARITY	X	X
SLP	-1612	STORE LOGICAL WORD WITH PARITY	X	X
SAC	-1623	STORE ACCUMULATOR CHARACTER	X	X
MSM	-1623.6	MAKE STORAGE SIGN MINUS	X	X
MSP	-1623.7	MAKE STORAGE SIGN PLUS	X	X
TSL	-1627	TRANS AND STORE INSTR LOCATION COUNTER	X	X
TMT	-1704	TRANSMIT	X	X
ICT	-1760.00014	INHIBIT CHANNEL TRAPS	X	X
SLNA	-1760.01541	STATUS LINE ON, CHANNEL A	X	X
PRD	-1762	PREPARE TO READ	X	X
SEN	-1762	SENSE SELECT XREAD	X	X
CTR	-1766	CONTROL SELECT XWRITE	X	X
PWR	-1766	PREPARE TO WRITE	X	X
TIX	2000	TRANSFER ON INDEX	X	X
TNX	-2000	TRANSFER ON NO INDEX	X	X
TXH	3000	TRANSFER ON INDEX HIGH	X	X
TXL	-3000	TRANSFER ON INDEX LOW OR EQUAL	X	X

NOTES
F - Represents an indirectly addressable instruction
T - Represents an indexable instruction
N - Represents a number 1 to 6

IBM
International Business Machines Corporation
Data Processing Division
112 East Post Road, White Plains, New York

IBM 7070-7074 Data Processing System Reference Card

CORE STORAGE LOCATIONS WITH SPECIAL FUNCTIONS

0001-0099 Indexing words 01-99
 0097 Priority address word
 0098 Table lookup indexing value and found address
 0099 Address of 729 tape priority final-status word
 0100 Priority indicator storage word
 0101-0103 Electronic switches
 0104 Unit-record priority A branch address
 0105 Unit-record priority B branch address
 0106 Inquiry-control 1 priority branch address
 0107 Inquiry-control 2 priority branch address
 01CU 0110-0149; 729 tape final status words
 C=7604 channel (1-4); U=729 tape unit (0-9)
 0150-0159 729 tape priority branch addresses
 01CU+50 0160-0199; 729 tape initial status words
 0301-0304 7907 data channel final status words
 0351-0354 7907 data channel initial status words
 0311-0314 7907 normal stacking latch branch addresses
 0321-0324 7907 attention stacking latch branch addresses

INITIAL STATUS WORD

S Sign of 729 tape instruction
 0 Always 8
 1 Operation—from pos. 5 of instruction
 2345 Location + 1 of instruction
 6789 Address in instruction

FINAL STATUS WORD

S Always +
 0 Units digit of priority branch address
 1 Condition Code:
 0. TWE
 1. ERR
 2. CLR
 3. SLR
 4. LLR
 5. EOF
 6. EOS
 7. SCLR
 2345 Last core storage location used by RDW
 6789 Location + 1 of last RDW



International Business Machines Corporation
 Data Processing Division
 112 East Post Road, White Plains, New York

Printed in U.S.A.

Form X22-6467-3

PRIORITY INDICATOR STORAGE WORD, 0100

0 Sign Change:
 0. Off, sense/stop switch at stop
 5. Off, sense/stop switch at sense
 1 Field Overflow:
 0. Off, sense/stop switch at stop
 5. Off, sense/stop switch at sense
 9. On, sense/stop switch at sense
 2 Floating Point Underflow: 0-Off; 1-On
 3 Floating Point Overflow: 0-Off; 1-On
 4 High (compare): 0-Off; 1-On
 5 Equal (compare): 0-Off; 1-On
 6 Low (compare): 0-Off; 1-On
 7 Accumulator 1 Overflow: 0-Off; 1-On
 8 Accumulator 2 Overflow: 0-Off; 1-On
 9 Accumulator 3 Overflow: 0-Off; 1-On

PRIORITY MASK WORD

0 (0—allow priority; 1—mask priority)
 1 729 tape channel 4, units 0-9
 2 729 tape channel 3, units 0-9
 3 Inquiry control group 2, stations 1-5
 4 Inquiry control group 1, stations 1-5
 5 729 tape channel 2, units 0-9
 6 729 tape channel 1, units 0-9
 7 Unit record 8
 8 Unit record A
 9 Unit record B

STORE AND ADD-TO-STORAGE CODES

All store and add-to-storage codes can turn on the field-overflow indicator:
 -11, -21, -31 Zero Storage and Store Accumulator # ZST1, ZST2, ZST3
 +12, +22, +32 Store Accumulator # ST1, ST2, ST3
 -12, -22, -32 Store Digits from Accumulator # and Ignore Sign STD1, STD2, STD3
 +18, +28, +38 Add to Storage from Accumulator # ASI, AS2, AS3
 -18, -28, -38 Subtract Accumulator # from Storage SS1, SS2, SS3
 +19, +29, +39 Add to Absolute Storage from Accumulator # AAS1, AAS2, AAS3
 These codes can turn on the sign-change indicator:
 +12, +22, +32 Store Accumulator # ST1, ST2, ST3
 +18, +28, +38 Add to Storage from Accumulator # ASI, AS2, AS3
 -18, -28, -38 Subtract Accumulator # from Storage SS1, SS2, SS3
 *Only if less than a full word is field-defined
 These codes cannot turn on the sign-change indicator:
 -11, -21, -31 Zero Storage and Store Accumulator # ZST1, ZST2, ZST3
 -12, -22, -32 Store Digits from Accumulator # and Ignore Sign STD1, STD2, STD3
 +19, +29, +39 Add to Absolute Storage from Accumulator # AAS1, AAS2, AAS3

DIGIT AND SIGN CODING

Bit Position	Digit Value										Sign Value		
	0	1	2	3	4	5	6	7	8	9	A	-	+
0	X	X	X	X	X	X	X	X	X	X			
1	X	X	X	X	X	X	X	X	X	X			
2	X	X	X	X	X	X	X	X	X	X			
3			X	X	X	X	X	X	X	X			
6					X	X	X	X	X	X			

7070-7074 CODING SYSTEMS

Character	Core Storage Alphabetical Code	Core Storage 1-Digit Numerical Code	Card Code	Magnetic Tape BCD Code	Notes
Blank	00		12-3-8	CA	8
15			12-4-8	CBA821	
□ or	16		12-5-8	BA84	
17			12-6-8	CBA841	1
18			12-7-8	CBA842	1
GM #	19		12-7-8	BA8421	1
& or +	20		12	BA	
\$	25		11-3-8	B821	
26			11-4-8	CB84	
27			11-5-8	B841	1
28			11-6-8	B842	1
30			11	CB	7
/	31		0-1	A1	
35			0-3-8	AB21	
% or (36		0-4-8	CB84	
37			0-5-8	AB41	1
38			0-6-8	AB42	1
SM	39		0-7-8	CA8421	1, 5
# or =	45		3-8	CB21	
46			4-8	B4	
@ or !	47		5-8	CB41	1
48			6-8	CB42	1
49			7-8	8421	1, 6
+0	60		12-0	BA82	2
A	61		12-1	CB41	
B	62		12-2	CB42	
C	63		12-3	BA21	
D	64		12-4	CB44	
E	65		12-5	BA41	
F	66		12-6	BA42	
G	67		12-7	CBA421	
H	68		12-8	CB48	
I	69		12-9	BA81	
-0	70		11-0	CB82	2
J	71		11-1	B1	
K	72		11-2	B2	
L	73		11-3	CB21	
M	74		11-4	B4	
N	75		11-5	CB41	
O	76		11-6	CB42	
P	77		11-7	B421	
Q	78		11-8	B8	
R	79		11-9	CB81	
RM #	80		0-2-8	CB82	3
S	81		0-2	A2	
T	83		0-3	CA21	
U	84		0-4	A4	
V	85		0-5	CA41	
W	86		0-6	CA42	
X	87		0-7	AA21	
Y	88		0-8	A8	
Z	89		0-9	CA81	
0	90	0	0	B2	7
1	91	1	1	C1	7
2	92	2	2	C2	7
3	93	3	3	21	7
4	94	4	4	C4	7
5	95	5	5	41	7
6	96	6	6	42	7
7	97	7	7	CA21	7
8	98	8	8	C8	7
9	99	9	9	B1	7
Delta Δ			11-7-8	CB8421	4
Alpha α			12-11		7

NOTES:

- Cannot be read into the IBM 7500 Card Reader, nor are they translated on output to the 7500 Card Punch, 7400 Printer, 7900 Inquiry Station, or the Console typewriter.
- Cannot be read by the 7500 Card Reader unless they are wired as the units position of numeric words. Similarly, on punching or printing operations, these codes (60, 70) require special wiring.
- This code cannot be wired to read or punch on the unit record equipment.
- Generated by the system controls on write operations, and not translated on read operations. This card code (11-7-8) cannot be entered through the 7500 Card Reader.
- The tape segment mark Write instruction (TSM) writes a single-character tape record that causes a condition six tape final status word (end of segment—EOS) when it is read; it is not placed in core storage. However, the tape segment mark can be read into core storage as a character within a tape record having a length of five or more characters. Also, the tape segment mark can be written from core storage as part of a tape record.
- The tape mark is handled the same as the tape segment mark.
- Only the numerals 0 through 9 and special characters +, -, and alpha can be read in the 7501 Console Card Reader. The 7500 Card Reader does not read the 12-11 alpha punch.
- Note that no invalid (nontranslatable) 2-digit combinations are included. Any invalid 2-digit combination in core storage would be written on tape as a check (C) bit.

IBM 7070-7074 Data Processing System Reference Card

Operation Codes

MNEMONIC SEQUENCE

A# + 14, 24, 34 AA + 17 AAS# + 19, 29, 39 AS# + 18, 28, 38

B + 01 BAL + 60 BAS, BCB + 51 BCX - 43 BDCA, BDCL + 60 BDX - 49
 BE - 41 BES + 61, 62, 63 BFV + 41 BH - 40 BIX + 49 BL + 40
 BLX + 02 BM# - 10, 20, 30 BQL + 60 BSC - 03 BSP, BSN + 61, 62, 63
 BTL, BUL + 60 BV# + 11, 21, 31 BXM - 44 BXN + 44 BZ# + 10, 20, 30

C# + 15, 25, 35 CA - 15 CD + 03 CSA, CSM, CSP - 03

D - 53 DCAF, DCLF - 62 DCAN, DCLN - 61 DCP, DCUA, DCUR, DCPR, DCU - 93, 94, 96, 97

EAN - 57 ENA + 56 ENB + 57 ENS - 56 ESF, ESN + 61, 62, 63

FA + 74 FAA + 77 FAD + 76 FADS - 76 FBU - 70 FBV + 70
 FD - 73 FDD - 75 FM + 73 FR + 71 FS - 74 FSA - 77 FZA + 75

HB + 00 HMFV + 41 HMSC - 03 HP - 00

ITS, ITZ + 69

LE + 67 LEH + 68 LL + 66

M + 53 MSA, MSM, MSP - 03

NOP - 01

PC, PC1 + 55 PDCP, PDCUA, PDCUR, PDCPR, PDCU + 93, 94, 96, 97 PR + 64
 PTM, PTR, PTRR, PTRR, PTSE, PTSE, PTSM, PTW, PTWC, PTWR, PTWZ + 81, 82, 83, 84 PTRN + 88

QLF - 62 QLN - 61 QR, QW + 54

RG - 65 RS + 65

S# - 14, 24, 34 SA - 17 SL - 50 SL# + 50 SLC - 50 SLC# + 50
 SLS - 50 SMFV + 41 SMSC - 03 SR - 50 SR# + 50 SRR - 50
 SRR# + 50 SRS - 50 SS# - 18, 28, 38 ST# + 12, 22, 32 STD# - 12, 22, 32

TEF + 81, 82, 83, 84 TLF - 62 TLN - 61 TM, TR, TRA - 81, 82, 83, 84 TRB
 ± 81, 82, 83, 84 TRN - 88 TRR - 81, 82, 83, 84 TRU, TRW ± 81, 82, 83, 84
 TSB - 81, 82, 83, 84 TSEL ± 81, 82, 83, 84 TSF - 81, 82, 83, 84 TSHD,
 TSK, TSLD ± 81, 82, 83, 84 TSM, TW, TWC, TWR, TWZ - 81, 82, 83, 84 TYP + 69

ULF - 62 ULN - 61 UP, UPIV, UR, US, UW, UWIV + 69

XA + 47 XL + 45 XLIN - 48 XS - 47 XSN + 48 XU - 45 XZA + 46
 XZS - 46

ZA# + 13, 23, 33 ZAA + 16 ZS# - 13, 23, 33 ZSA - 16 ZST# - 11, 21, 31

NUMERIC SEQUENCE

+00	HB	+01	B	+02	BLX	+03	CD	+04		+05		+06		+07		+08		+09	
-00	HP	-01	NOP	-02		-03	(1)	-04	(16)	-05		-06		-07		-08		-09	
+10	BZ1	+11	BV1	+12	ST1	+13	ZA1	+14	A1	+15	C1	+16	ZAA	+17	AA	+18	AS1	+19	AAS1
-10	BM1	-11	ZST1	-12	STD1	-13	ZS1	-14	S1	-15	CA	-16	ZSA	-17	SA	-18	SS1	-19	
+20	BZ2	+21	BV2	+22	ST2	+23	ZA2	+24	A2	+25	C2	+26		+27		+28	AS2	+29	AAS2
-20	BM2	-21	ZST2	-22	STD2	-23	ZS2	-24	S2	-25		-26		-27		-28	SS2	-29	
+30	BZ3	+31	BV3	+32	ST3	+33	ZA3	+34	A3	+35	C3	+36		+37		+38	AS3	+39	AAS3
-30	BM3	-31	ZST3	-32	STD3	-33	ZS3	-34	S3	-35		-36		-37		-38	SS3	-39	
+40	BL	+41	(2)	+42		+43		+44	BXN	+45	XL	+46	XZA	+47	XA	+48	XSN	+49	BIX
-40	BH	-41	BE	-42		-43	BCX	-44	BXM	-45	XU	-46	XZS	-47	XS	-48	XLIN	-49	BDX
+50	(3)	+51	(5)	+52		+53	M	+54	(6)	+55	PC	+56	ENA	+57	ENB	+58		+59	
-50	(4)	-51		-52		-53	D	-54		-55		-56	ENS	-57	EAN	-58		-59	
+60	(7)	+61	(8)	+62	(8)	+63	(8)	+64	PR	+65	RS	+66	LL	+67	LE	+68	LEH	+69	(11)
-60		-61	(9)	-62	(10)	-63		-64		-65	RG	-66		-67		-68		-69	
+70	FBV	+71	FR	+72		+73	FM	+74	FA	+75	FZA	+76	FAD	+77	FAA	+78		+79	
-70	FBU	-71		-72		-73	FD	-74	FS	-75	FDD	-76	FADS	-77	FSA	-78		-79	
+80		+81	(12)	+82	(12)	+83	(12)	+84	(12)	+85		+86		+87		+88	PTRN	+89	
-80		-81	(13)	-82	(13)	-83	(13)	-84	(13)	-85		-86		-87		-88	TRN	-89	
+90		+91		+92		+93	(14)	+94	(14)	+95	PDS	+96	(1A)	+97	(1A)	+98		+99	
-90		-91		-92		-93	(15)	-94	(15)	-95	DAR	-96	(15)	-97	(15)	-98		-99	

AUGMENTED CODES

① - 03, Sign Control	④ - 50, Coupled Shift Control	⑧ + 61, + 62, + 63 Electronic Switch Control	⑪ + 69 Unit Record Control	- 8cxxx7xxxx TSF
- 03xxx30xxxx CSA	- 50xxxx00xxx SR	+ 61xx0xxxx BES	+ 69xx0xxxx US	- 8cxxx8xxxx TSB
- 03xxx0xxxx CSM	- 50xxxx01xxx SRR	+ 61xx1xxxx ESN	+ 69xx1xxxx UR	- 8cxxx9xxxx TRA
- 03xxx90xxxx CSP	- 50xxxx02xxx SL	+ 61xx2xxxx ESN	+ 69xx2xxxx UW-UP	- 8cxxx0xxxx0 TSEL
- 03xx31xxxx MSA	- 50xxxx03xxx SLC	+ 61xx3xxxx ESP	+ 69xx3xxxx UWIV-UIPV	- 8cxxx1xxxx1 TM
- 03xx01xxxx MSM	- 50xxxxp4xxx SRS	+ 61xx4xxxx BSN	+ 69xx4xxxx TYP	- 8cxxx2xxxx2 TRW
- 03xx91xxxx MSP	- 50xxxxp5xxx SLS	+ 61xx5xxxx BSF	+ 69xx41xxxx UR-CONSOLE	- 8cxxx3xxxx3 TRU
- 03xxx2xxxx SMSC	- 50xxxxp6xxx SRS		+ 69xx90xxxx ITZ	- 8cxxx4xxxx4 TRB
- 03xxx3xxxx HMSC	- 50xxxxp7xxx SLS		+ 69xx91xxxx ITS	- 8cxxx5xxxx5 TSM
- 03xxx4xxxx BSC				- 8cxxx6xxxx6 TSK
				- 8cxxx7xxxx7 TEF
				- 8cxxx8xxxx8 TSLD
				- 8cxxx9xxxx9 TSHD
② + 41, Field Overflow Control	⑤ + 51, Branch on Alteration Switch or Channel Busy	⑨ - 61 Stacking Latch Set	⑫ + 81, + 82, + 83, + 84 Priority Tape Control	⑭ + 93, + 94, + 96, + 97 Priority 7907 Control
+ 41xxx0xxxx BFV	+ 51xxx0xxxx BAS	- 61xx01xxxx ULN	+ 8cxt1xxxx PTR	+ 93xxx1xxxx PDCP
+ 41xxx1xxxx SMFV	+ 51xxx1xxxx BCB	- 61xx02xxxx QLN	+ 8cxt2xxxx PTRR	+ 93xxx2xxxx PDCUA
+ 41xxx2xxxx HMFV	+ 51xxx2xxxx BDCB	- 61xx03xxxx TLN	+ 8cxt3xxxx PTW	+ 93xxx3xxxx PDCUR
		- 61xx10xxxx to	+ 8cxt4xxxx PTWF	+ 93xxx4xxxx PDCU
		- 61xx49xxxx to	+ 8cxt6xxxx PTWC	
		- 61xx8dxxxx DCAN	+ 8cxt7xxxx PTF	
		- 61xx9dxxxx DCAN	+ 8cxt8xxxx PTF	
			+ 8cxt9xxxx PTF	
			+ 8cxt10xxxx PTF	
			+ 8cxt11xxxx PTF	
			+ 8cxt12xxxx PTF	
			+ 8cxt13xxxx PTF	
			+ 8cxt14xxxx PTF	
			+ 8cxt15xxxx PTF	
			+ 8cxt16xxxx PTF	
			+ 8cxt17xxxx PTF	
			+ 8cxt18xxxx PTF	
			+ 8cxt19xxxx PTF	
			+ 8cxt20xxxx PTF	
			+ 8cxt21xxxx PTF	
			+ 8cxt22xxxx PTF	
			+ 8cxt23xxxx PTF	
			+ 8cxt24xxxx PTF	
			+ 8cxt25xxxx PTF	
			+ 8cxt26xxxx PTF	
			+ 8cxt27xxxx PTF	
			+ 8cxt28xxxx PTF	
			+ 8cxt29xxxx PTF	
			+ 8cxt30xxxx PTF	
			+ 8cxt31xxxx PTF	
			+ 8cxt32xxxx PTF	
			+ 8cxt33xxxx PTF	
			+ 8cxt34xxxx PTF	
			+ 8cxt35xxxx PTF	
			+ 8cxt36xxxx PTF	
			+ 8cxt37xxxx PTF	
			+ 8cxt38xxxx PTF	
			+ 8cxt39xxxx PTF	
			+ 8cxt40xxxx PTF	
			+ 8cxt41xxxx PTF	
			+ 8cxt42xxxx PTF	
			+ 8cxt43xxxx PTF	
			+ 8cxt44xxxx PTF	
			+ 8cxt45xxxx PTF	
			+ 8cxt46xxxx PTF	
			+ 8cxt47xxxx PTF	
			+ 8cxt48xxxx PTF	
			+ 8cxt49xxxx PTF	
			+ 8cxt50xxxx PTF	
			+ 8cxt51xxxx PTF	
			+ 8cxt52xxxx PTF	
			+ 8cxt53xxxx PTF	
			+ 8cxt54xxxx PTF	
			+ 8cxt55xxxx PTF	
			+ 8cxt56xxxx PTF	
			+ 8cxt57xxxx PTF	
			+ 8cxt58xxxx PTF	
			+ 8cxt59xxxx PTF	
			+ 8cxt60xxxx PTF	
			+ 8cxt61xxxx PTF	
			+ 8cxt62xxxx PTF	
			+ 8cxt63xxxx PTF	
			+ 8cxt64xxxx PTF	
			+ 8cxt65xxxx PTF	
			+ 8cxt66xxxx PTF	
			+ 8cxt67xxxx PTF	
			+ 8cxt68xxxx PTF	
			+ 8cxt69xxxx PTF	
			+ 8cxt70xxxx PTF	
			+ 8cxt71xxxx PTF	
			+ 8cxt72xxxx PTF	
			+ 8cxt73xxxx PTF	
			+ 8cxt74xxxx PTF	
			+ 8cxt75xxxx PTF	
			+ 8cxt76xxxx PTF	
			+ 8cxt77xxxx PTF	
			+ 8cxt78xxxx PTF	
			+ 8cxt79xxxx PTF	
			+ 8cxt80xxxx PTF	
			+ 8cxt81xxxx PTF	
			+ 8cxt82xxxx PTF	
			+ 8cxt83xxxx PTF	
			+ 8cxt84xxxx PTF	
			+ 8cxt85xxxx PTF	
			+ 8cxt86xxxx PTF	
			+ 8cxt87xxxx PTF	
			+ 8cxt88xxxx PTF	
			+ 8cxt89xxxx PTF	
			+ 8cxt90xxxx PTF	
			+ 8cxt91xxxx PTF	
			+ 8cxt92xxxx PTF	
			+ 8cxt93xxxx PTF	
			+ 8cxt94xxxx PTF	
			+ 8cxt95xxxx PTF	
			+ 8cxt96xxxx PTF	
			+ 8cxt97xxxx PTF	
			+ 8cxt98xxxx PTF	
			+ 8cxt99xxxx PTF	

LEGEND
 a = alteration switch (1-4) d = 7907 channel (1-4) h = 7907 channel; 3 = ch1
 c = 729 tape channel (1-4) e = electronic switch (0-9) 4 = 2, 6 = 3, 7 = 4
 g = electronic switch group (1-3) p = digit position (0-9) i = inquiry control group (1-2)
 j = unit-record synchronizer (1-3) t = tape unit (0-9)

IBM 7080 Data Processing System Reference Card

Printed in U.S.A. X22-6561-1

INSTRUCTION	MNEMONIC	CODE		
		OP	BIT	CARD
Add	ADD	G	111 0111	12-7
Add Addr to Mem	AAM	@	000 1100	4-8
Add to Mem	ADM	6	000 0110	6
Blank Memory BLM 00	BLM	\$	010 1011	11-3-8
Blank Memory Serial BLM 01	BLMS	\$	010 1011	11-3-8
Comma 00 Set Start Point Count	SPC	.	001 1011	0-3-8
Comma 02 Load Four Char	LFC	.	001 1011	0-3-8
Comma 03 Unload Four Char	UFC	.	001 1011	0-3-8
Comma 04 Load Stor Bank	LSB	.	001 1011	0-3-8
Comma 05 Unload Stor Bank	USB	.	001 1011	0-3-8
Comma 06 Enter Interrupt Mode	EIM	.	001 1011	0-3-8
Comma 07 Leave Interrupt Mode	LIM	.	001 1011	0-3-8
Comma 08 Ten Character Transmit	TCT	.	001 1011	0-3-8
Comma 10 Enable Indirect Address	EIA	.	001 1011	0-3-8
Comma 14 Transfer to Interrupt Prog.	TIP	.	001 1011	0-3-8
Comma 15 Leave Interrupt Prog.	LIP	.	001 1011	0-3-8
Compare	CMP	4	100 0100	4
Control 0000 Turn off I-O Ind.	IOF	3	000 0011	3
Control 0001 Write Tape Mark	WTM	3	000 0011	3
Control 0002 Rewind	RWD	3	000 0011	3
Control 0002 (01) Rewind and Unload	RUN	3	000 0011	3
Control 0003 Turn on I-O Ind.	ION	3	000 0011	3
Control 0004 Backspace	BSP	3	000 0011	3
Control 0004 (01) Backspace File	BSF	3	000 0011	3
Control 0005 Supp print or punch	SUP	3	000 0011	3
Control 0009 Skip Tape	SKP	3	000 0011	3
Control xxxx (12) Enable Compare Backward	ECB	3	000 0011	3
Control xxxx (13) Channel Reset	CHR	3	000 0011	3
Control xxxx (14) Enter 7080 Mode	EEM	3	000 0011	3
Control xxxx (15) Leave 7080 Mode	LEM	3	000 0011	3
Control 0037 Set Density Low	SDL	3	000 0011	3
Control 0038 Set Density High	SDH	3	000 0011	3
Divide	DIV	W	101 0110	0-6
Lengthen	LNG	D	111 0100	12-4
Load	LOD	8	100 1000	8
Load Address	LDA	#	100 1011	3-8
Multiply	MPY	V	101 0101	0-5
No Operation	NOP	A	111 0001	12-1
No Operation, Comma (11)	CNO	.	001 1011	0-3-8
Norm and Tr	NTR	X	001 0111	0-7
Read 00	RD	Y	001 1000	0-8
Read 01 Forward Space	FSP	Y	001 1000	0-8
Read 02 Read Memory Address	RMA	Y	001 1000	0-8
Read 03 Sense Status Trigger	SST	Y	001 1000	0-8
Read 04 Control Read	CRD	Y	001 1000	0-8
Read 05 Read Memory Block	RMB	Y	001 1000	0-8
Read while Writing	RWW	S	001 0010	0-2
Receive	RCV	U	001 0100	0-4
Reset Add	RAD	H	111 1000	12-8
Reset Subtract	RSU	Q	010 1000	11-8
Round	RND	E	011 0101	12-5

SPECIAL CHARACTERS	CODE	
	BIT	CARD
&	0 11 0000	12
+	0 01 1011	0-3-8
⊥	0 11 1100	12-4-8
0	0 00 1010	0
△	1 11 1110	None
Blank	1 01 0000	None
Drum Mark or Storage Mark (in storage)	0 00 0000	None
Card Reader Storage Mark	None	12-1-4-7
Group Mark	0 11 1111	12-5-8; 12-7-8
Record Mark	1 01 1010	0-2-8
+	0 11 1010	12-0
0	1 10 1010	11-0

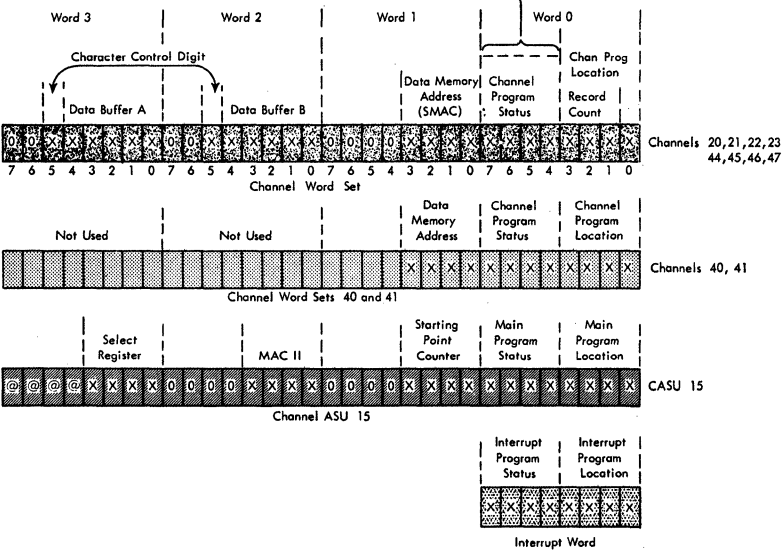
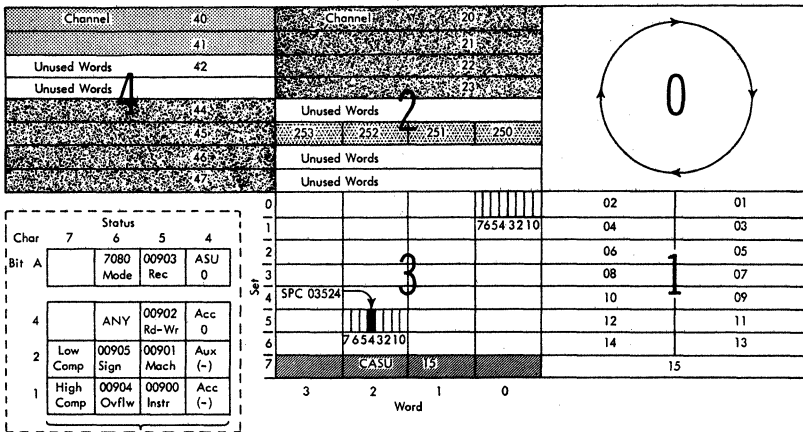
Mode	705 I-II Switch	40 K Switch	Program Wrap Around
705 I	ON	OFF	20 K
705 II	ON	ON	40 K
705 III	OFF	ON	40 K
705 III	OFF	OFF	80 K

INSTRUCTION	MNEMONIC	CODE		
		OP	BIT	CARD
Select	SEL	2	100 0010	2
Select (Sel 6001 Reset WTC Trigger)		2	100 0010	2
Select (Sel 6002 Reset SAR 8 Trigger)		2	100 0010	2
Send	SND	/	001 0001	0-1
Set Bit Alternate SB 07	SBA	%	101 1100	0-4-3
Set Bit 1 SB09-14	SBN	%	101 1100	0-4-8
Set Bit Redundant SB08	SBR	%	101 1100	0-4-8
Set Bit 0 SB 01-06	SBZ	%	101 1100	0-4-8
Set Left	SET	B	111 0010	12-2
Shorten	SHR	C	011 0011	12-3
Sign	SGN	T	101 0011	0-3
Stop	HLT	J	010 0001	11-1
Store	ST	F	011 0110	12-6
Store for Print	SPR	5	000 0101	5
Subtract	SUB	P	010 0111	11-7
Transfer	TR	1	100 0001	1
Transfer Auto Restart (TRS 09)	TAR	0	110 0110	11-6
Tr Sw A On 0911 (TRA 01)	TAA	I	011 1001	12-9
Tr Sw B On 0912 (TRA 02)	TAB	I	011 1001	12-9
Tr Sw C On 0913 (TRA 03)	TAC	I	011 1001	12-9
Tr Sw D On 0914 (TRA 04)	TAD	I	011 1001	12-9
Tr Sw E On 0915 (TRA 05)	TAE	I	011 1001	12-9
Tr Sw F On 0916 (TRA 06)	TAF	I	011 1001	12-9
Tr Non-stop Sw (TRA 07)	TNS	I	011 1001	12-9
Tr and Store Location Ctr (Tr 01)	TSL	I	100 0001	1
Tr Any	TRA	I	011 1001	12-9
Tr Read-Write Check (TR 12)	TRC	O	110 0110	11-6
Tr Equal	TRE	L	110 0011	11-3
Tr High	TRH	K	010 0010	11-2
Tr Instr Check (TRS 10)	TIC	O	110 0110	11-6
Tr Machine Check (TRS 11)	TMC	O	110 0110	11-6
Tr O'flow Check (TRS 14)	TOC	O	110 0110	11-6
Tr Plus	TRP	M	010 0100	11-4
Tr Ready (TRS 01)	TRR	O	110 0110	11-6
Tr Echo Check (TRS 13)	TEC	O	110 0110	11-6
Tr Sign Check (TRS 15)	TRS	O	110 0110	11-6
Tr Signal	TRS	O	110 0110	11-6
Tr Sync Any (TRS 03)	TSA	O	110 0110	11-6
Tr Transmission Check (TRS 02)	TTC	O	110 0110	11-6
Tr Zero	TRZ	N	110 0101	11-5
Tr Zero Bit	TZB	.	111 1011	12-3-8
Transmit	TMT	9	000 1001	9
Transmit Serial (TMT 01-15)	TMTS	9	000 1001	9
Unload	UNL	7	100 0111	7
Unload Address	ULA	*	110 1100	11-4-8
Write 00	WR	R	110 1001	11-9
Write 01 (Dump Memory)	DMP	R	110 1001	11-9
Write 02 (Set Record Counter)	SRC	R	110 1001	11-9
Write 03 (Set Control Condition)	SRC	R	110 1001	11-9
Write 04 (Control Write)	CWR	R	110 1001	11-9
Write 05 (Write Multiple Control)	WMC	R	110 1001	11-9
Write Erase 00	WRE	Z	101 1001	0-9
Write Erase 01	WRE	Z	101 1001	0-9

Card Readers	0100-0199
Console Card	
Reader	0100-0109
754, 760, and Tape Units	0200-0299
Card Punches	0300-0399
Printers 717	0400-0499
720-730	02x4
Typewriter	0500
Real Time Clock	0501
TRC Tape Units	0600-0699
Alteration Switches	0911-0916
Channel Tape Units	2000-2999
	or 0200-0299

Instruction	0900
Machine	0901
Read/Write	0902
Record Check	0903
Overflow	0904
Sign	0905

7080 CENTRAL STORAGE



IBM
International Business Machines Corporation
Data Processing Division
112 East Post Road, White Plains, New York

Orders	Mnemonic Code	Numeric Code
No Operation	DNOP	00
Release	DREL	04
Eight-Bit Mode	DEBM	08
Six-Bit Mode	DSBM	09
Seek	DSEK	80
Prepare to Verify—Single Record	DVSR	82
Prepare to Write Format	DWRF	83
Prepare to Verify—Track with No Addresses	DVTN	84
Prepare to Verify—Cylinder Operation (Optional Feature)	DVCY	85
Prepare to Write Check	DWRC	86
Set Access Inoperative	DSAI	87
Prepare to Verify—Track Operation	DVTA	88
Prepare to Verify—Home Address Operation	DVHA	89

Orders	Mnemonic Code	Numeric Code
No Operation	HNOP	00
End of Sequence	HEOS	01
Reserved Light Off	HRLF	02
Reserved Light On	HRLN	03
Check Light On	HCLN	05
Select	HSEL	06
Select for Backward Reading	HSBR	07
Change Cartridge and Rewind	HCCR	28
Rewind	HRWD	30
Rewind and Unload Cartridge	HRUN	31
Erase Long Gap	HERG	32
Write Tape Mark	HWTM	33
Backspace	HBSR	34
Backspace File	HBSF	35
Space	HSKR	36
Space File	HSKF	37
Change Cartridge	HCHC	38
Unload Cartridge	HUNL	39
File Protect On	HFPN	42

IBM Reference Card

7090 Codes

X22-6682
Printed in U.S.A.

Notes:
1. F = Indirectly Addressable Instruction
2. T = Indexable Instruction
3. c = 01 to 10 or A to H
4. u = Unit Address
5. V = Variable
6. n = A number from 1 to 6

INSTRUCTION LISTING - ALPHABETIC

Alpha Codes	Octal Codes	F	T	CPU Cycles	Alpha Codes	Octal Codes	F	T	CPU Cycles		
ACL	0361	x	x	2	LLS	0763			x	v	
ADD	0400	x	x	2	LNT	-0056			x	3	
ADM	0401	x	x	2	LRS	0765			x	v	
ALS	0767	x	x	v	LSNM	-076000010			x	2	
ANA	-0320	x	x	3	LTM	-076000007			x	2	
ANS	0320	x	x	4	LXA	0534			x	2	
ARS	0771	x	x	v	LXD	-0534			x	2	
AXC	-0774			1	MPR	-0200	x	x	2,14		
AXT	0774			1	MPY	0200	x	x	2,14		
BSFc	-0764ccc2uu			x	2	MSE	-07600			x	2
BSRC	0764ccc2uu			x	2	ONT	0446			x	4
BTTC	0760ccc000			x	2	ORA	-0501			x	x
CAL	-0500	x	x	2	2	OAI	0043			x	1
CAQ	-0114			2,8	OFT	0444	x	x	4		
CAS	0340	x	x	3	ONT	0446			x	4	
CHS	0760000002			x	2	ORA	-0501			x	x
CLA	0500	x	x	2	ORS	-0602	x	x	2		
CLM	0760000000			x	2	OS1	0442			x	x
CLS	0502	x	x	2	PAC	0737			x	2	
COM	0760000006			x	2	PAI	0044			x	1
CRQ	-0154			2,8	PAX	0734			x	1	
CVR	0114			2,8	PBT	-076000001			x	2	
DCT	0760000012			x	2	PDC	-0737			x	1
DVH	0220	x	x	3,14	PDX	-0734			x	1	
DVP	0221	x	x	3,14	PIA	-0046			x	1	
ECTM	-0760000006			x	2	PSE	07600			x	2
EFTM	-0760000002			x	2	PXA	0754			x	1
ENB	0544	x	x	2	PKD	-0754			x	1	
ENK	0760000004			x	2	RCDC	0762ccc321			x	2
ERA	0322	x	x	3	RCHA	0540			x	3	
ESMT	-0021	x	x	1	RCHB	-0540			x	3	
ESTM	-0760000005			x	2	RCHC	0541			x	3
ETM	0760000007			x	2	RCHD	-0541			x	3
ETTC	-0760ccc000			x	2	RCHE	0542			x	3
FAD	0300	x	x	6,15	RCHF	-0542			x	3	
FAM	0304	x	x	6,15	RCHG	0543			x	3	
FDH	0240	x	x	3,14	RCHM	-0543			x	3	
FDP	0241	x	x	3,14	RCT	0760000014			x	2	
FMP	0260	x	x	2,13	RDCC	0760ccc352			x	2	
FBN	0760000011			x	2	RDS	07620			x	2
FSB	0302	x	x	6,15	RENC	0772ccuuu			x	2	
FSM	0306	x	x	6,15	RFT	0054			x	3	
HPR	0420			2	RIA	-0042			x	1	
HTR	0000	x	x	2	RICC	0760ccc350			x	2	
I1A	0041			1	RIL	-0057			x	1	
I1L	-0051			1	RIR	0057			x	1	
I1R	0051			1	RIS	0445			x	x	
I1S	0440	x	x	2	RND	0760000010			x	2	
IOT	0760000005			x	2	RNT	0056			x	3
LAC	0535			2	RPRC	0762ccc361			x	2	
LAS	-0340	x	x	3	RQL	-0773			x	v	
LBT	0760000001			x	2	RSCA	0540			x	3
LCHA	0544	x	x	v	RSCB	-0540			x	3	
LCHB	-0544	x	x	v	RSCC	0541			x	3	
LCHC	0545	x	x	v	RSCD	-0541			x	3	
LCHD	-0545	x	x	v	RSCD	0542			x	3	
LCHE	0546	x	x	v	RSCF	-0542			x	3	
LCHF	-0546	x	x	v	RSCG	0543			x	3	
LCHG	0547	x	x	v	RSCM	-0543			x	3	
LCHH	-0547	x	x	v	RTBC	0762ccc2uu			x	2	
LDC	-0535			2	RTCC	0762ccc2uu			x	2	
LDI	0441	x	x	2	RUNC	-0772ccuuu			x	2	
LDO	0560	x	x	2	SBM	-0400			x	2	
LFT	-0054			3	SCHA	0640			x	2	
LFTM	-0760000004			x	2	SCHB	-0640			x	2
LGL	-0763			x	v	SCHC	0641			x	2
LGR	-0765			x	v	SCHD	-0641			x	2

INSTRUCTION LISTING - ALPHABETIC

Alpha Codes	Octal Codes	F	T	CPU Cycles	Alpha Codes	Octal Codes	F	T	CPU Cycles		
SCHE	0642	x	x	2	TCOH	0067	x	x	2		
SCHF	-0642	x	x	2	TEFA	0030	x	x	2		
SCHG	0643	x	x	2	TEFB	-0030	x	x	2		
SCHM	-0643	x	x	2	TEFC	0031	x	x	2		
SDHC	0776ccc2uu			2	TEFD	-0031	x	x	2		
SDLC	0776ccc2uu			2	TEFE	0032	x	x	2		
SIL	-0055			1	TEFF	-0032	x	x	2		
SIR	0055			1	TEFG	0033	x	x	2		
SLF	0760000140			x	2	TEFH	-0033	x	x	2	
SLNM	076000014n			x	2	TIF	0046	x	x	2	
SLQ	-0620	x	x	2	TIO	0042	x	x	2		
S LTC	-076000014n			x	2	TIQ	0040	x	x	2	
SLW	0602	x	x	2	TIK	2000	x	x	2		
SPRN	0760ccc3nn			x	2	TIL	0040	x	x	2	
SPTC	0760ccc360			x	2	TMI	-0120	x	x	1	
SPUN	0760ccc34n			x	2	TNO	-0140	x	x	1	
SSM	-0760000003			x	2	TNK	-2000			x	2
SSP	0760000003			x	2	TNZ	-0100	x	x	2	
STA	0621	x	x	2	TOV	0140	x	x	2		
STCA	0544			v	TPL	0120	x	x	1		
STCB	-0544			v	TGO	0161	x	x	1		
STCC	0545			v	TGP	0162	x	x	1		
STCD	-0545			v	TRA	0020	x	x	1		
STCE	0546			v	TRCA	0022	x	x	2		
STCF	-0546			v	TRCB	-0022	x	x	2		
STCG	0547			v	TRCC	0024	x	x	2		
STCH	-0547			v	TRCD	-0024	x	x	2		
STD	0622	x	x	2	TRCE	0026	x	x	2		
STI	0604	x	x	2	TRCF	-0026	x	x	2		
STL	-0625	x	x	2	TRCG	0027	x	x	2		
STO	0601	x	x	2	TRCH	-0027	x	x	2		
STP	0630	x	x	2	TSX	0074	x	x	2		
STQ	-0600	x	x	2	TTR	0021	x	x	1		
STR	-1000			2	TXH	3000			x	2	
STT	0625	x	x	2	TXI	1000			x	2	
STZ	0600	x	x	2	TXL	-3000			x	2	
SUB	0402	x	x	2	TZE	0100	x	x	2		
SWTn	076000016n			x	2	UAM	-0304	x	x	5,10	
SKA	0634			2	UFA	-0300	x	x	5,10		
SKD	-0634			2	UFM	-0260	x	x	2,13		
TCNA	-0060	x	x	2	UFS	-0302	x	x	5,9		
TCNB	-0061	x	x	2	USM	-0306	x	x	5,11		
TCNC	-0062	x	x	2	VDH	0224	x	x	2,14		
TCND	-0063	x	x	2	VDP	0225	x	x	2,14		
TCNE	-0064	x	x	2	VLM	0204	x	x	2,14		
TCNF	-0065	x	x	2	WFC	0770ccc22u			x	2	
TCNG	-0066	x	x	2	WPBC	0766ccc362			x	2	
TCNH	-0067	x	x	2	WPDC	0766ccc361			x	2	
TCOA	0060	x	x	2	WPUC	0766ccc341			x	2	
TCOB	0061	x	x	2	WRS	07660			x	2	
TCOC	0062	x	x	2	WTBC	0766ccc2uu			x	2	
TCOD	0063	x	x	2	WTDc	0766ccc2uu			x	2	
TCOE	0064	x	x	2	XCA	0131			x	1	
TCOF	0065	x	x	2	XCL	-0130			x	1	
TCOG	0066	x	x	2	XEC	0522			x	1	
					ZET	0520			x	2	

INPUT-OUTPUT DEVICE ADDRESSES

Channel	Magnetic Tape Units	BCD	Binary	Card Readers	Channel	Octal
A	01201	TO	01212 01221 TO 01232	A	01321	
B	02201	TO	02212 02221 TO 02232	B	02321	
C	03201	TO	03212 03221 TO 03232	C	03321	
D	04201	TO	04212 04221 TO 04232	D	04321	
E	05201	TO	05212 05221 TO 05232	E	05321	
F	06201	TO	06212 06221 TO 06232	F	06321	
G	07201	TO	07212 07221 TO 07232	G	07321	
H	10201	TO	10212 10221 TO 10232	H	10321	

Printers	Channel	Octal
A	01361	01362
B	02361	02362
C	03361	03362
D	04361	04362
E	05361	05362
F	06361	06362
G	07361	07362
H	10361	10362

INSTRUCTION LISTING - NUMERIC

Table with columns: Octal Codes, Alpha Codes, F, T, CPU Cycles. Contains instruction listings for codes 0000 to 0120.

INSTRUCTION LISTING - NUMERIC

Table with columns: Octal Codes, Alpha Codes, F, T, CPU Cycles. Contains instruction listings for codes 0541 to 0754.

IBM 7607/7909 DATA CHANNEL COMMANDS

Table with columns: Binary Codes, Alpha Codes, F, T. Lists data channel commands.

IBM 7631 FILE CONTROL ORDERS

Table with columns: Binary Positions, Alpha Codes, Num Code. Lists file control orders.

IBM 7640 HYPERTAPE CONTROL ORDERS

Table with columns: Binary Positions, Alpha Codes, Num Code. Lists hypertape control orders.

7904 DATA CHANNEL DIAGNOSTIC INSTRUCTIONS

Table with columns: SYM, NUMERIC, NAME, F, T. Lists diagnostic instructions for code 7904.

STORAGE PROTECTION INSTRUCTIONS

Table with columns: SYM, NUMERIC, NAME, F, T. Lists storage protection instructions.

DOUBLE PRECISION FLOATING POINT INSTRUCTIONS

Table with columns: SYM, NUMERIC, NAME, F, T. Lists double precision floating point instructions.

SINGLE PRECISION FLOATING POINT INSTRUCTIONS

Table with columns: SYM, NUMERIC, NAME, F, T. Lists single precision floating point instructions.

BASIC INSTRUCTION SET - ALPHABETIC

INPUT-OUTPUT DEVICE ADDRESS

SYM	NUMERIC	NAME	F	T
ACL	0361	ADD AND CARRY LOGICAL WORD	X	X
ADD	0400	ADD	X	X
ALS	0767	ACCUMULATOR LEFT SHIFT	X	X
ANA	-0320	AND TO ACCUMULATOR	X	X
ARS	0771	ACCUMULATOR RIGHT SHIFT	X	X
BSR	0764	BACKSPACE RECORD	X	X
CAD	-0500	CLEAR AND ADD LOGICAL WORD	X	X
CAL	-1510	CLEAR AND ADD LOGICAL WORD WITH PARITY	X	X
CAS	0340	COMPARE ACCUMULATOR WITH STORAGE	X	X
CHS	0760,00002	CHANGE SIGN	X	X
CLA	0500	CLEAR AND ADD	X	X
CLS	0502	CLEAR AND SUBTRACT	X	X
CLR	0760,00006	COMPLEMENT MAGNITUDE	X	X
CTR	-1766	CONTROL SELECT (WRITE)	X	X
DCT	0760,00012	DIVIDE CHECK TEST	X	X
DVP	0221	DIVIDE OR PROCEED	X	X
ENB	0564	ENABLE FROM Y	X	X
ENK	0760,00004	ENTER KEYS	X	X
ETTA	-0760,01000	END OF TAPE TEST, CHANNEL A	X	X
ETTB	-0760,02000	END OF TAPE TEST, CHANNEL B	X	X
ETTC	-0760,03000	END OF TAPE TEST, CHANNEL C	X	X
ETTD	-0760,04000	END OF TAPE TEST, CHANNEL D	X	X
ETTE	-0760,05000	END OF TAPE TEST, CHANNEL E	X	X
HPR	0420	HALT AND PROCEED	X	X
ICT	-1760,00014	INHIBIT CHANNEL TRAPS	X	X
IOT	0760,00005	INPUT/OUTPUT CHECK TEST	X	X
LAS	-0340	LOGICAL COMPARE ACCUMULATOR WITH STOR	X	X
LBT	0760,00001	LOW ORDER BIT TEST	X	X
LDO	0560	LOAD MQ	X	X
LGI	-0763	LOGICAL LEFT SHIFT	X	X
LGR	-0765	LOGICAL RIGHT SHIFT	X	X
LNG	0760	LONG LEFT SHIFT	X	X
LRS	0763	LONG RIGHT SHIFT	X	X
MPY	0200	MULTIPLY	X	X
ORA	-0901	OR TO ACCUMULATOR	X	X
PBT	-0760,00001	P BIT TEST	X	X
PRD	-1762	PREPARE TO READ	X	X
PWR	-1764	PREPARE TO WRITE	X	X
RCHA	0540	RESET AND LOAD CHANNEL A	X	X
RCHB	0540	RESET AND LOAD CHANNEL B	X	X
RCHC	0541	RESET AND LOAD CHANNEL C	X	X
RCHD	0541	RESET AND LOAD CHANNEL D	X	X
RCHE	0542	RESET AND LOAD CHANNEL E	X	X
RCL	0760,00014	RESTORE CHANNEL TRAPS	X	X
RDCA	0760,01352	RESET DATA CHANNEL A	X	X
RDCB	0760,02352	RESET DATA CHANNEL B	X	X
RDCC	0760,03352	RESET DATA CHANNEL C	X	X
RDCD	0760,04352	RESET DATA CHANNEL D	X	X
RDCE	0760,05352	RESET DATA CHANNEL E	X	X
RDS	0762	READ SELECT	X	X
REW	-0772	REWIND	X	X
ROL	-0773	ROTATE MQ LEFT	X	X
RUN	-0772	REWIND AND UNLOAD	X	X
SCHA	0640	STORE CHANNEL A	X	X
SCHB	0640	STORE CHANNEL B	X	X
SCHC	0641	STORE CHANNEL C	X	X
SCHD	0641	STORE CHANNEL D	X	X
SCH E	0642	STORE CHANNEL E	X	X
SEN	-1762	SENSE SELECT (READ)	X	X
SLC	1612	SENSE LOGICAL WORD WITH PARITY	X	X
SLW	0602	STORE LOGICAL WORD	X	X
SSP	0760,00003	SET SIGN PLUS	X	X
STA	0621	STORE ADDRESS	X	X
STD	0622	STORE DECREMENT	X	X
STI	-0623	STORE INSTRUCTION LOCATION COUNTER	X	X
STO	0601	STORE	X	X
STQ	-0600	STORE MQ	X	X
STR	-1000	STORE LOCATION AND TRAP	X	X
STZ	0600	STORE ZERO	X	X
SUB	0402	SUBTRACT	X	X
SWT	0760,0016N	SENSE SWITCH TEST	X	X
TCOA	0060	TRANSFER ON CHANNEL A IN OPERATION	X	X
TCOB	0061	TRANSFER ON CHANNEL B IN OPERATION	X	X
TCOC	0062	TRANSFER ON CHANNEL C IN OPERATION	X	X
TCOD	0063	TRANSFER ON CHANNEL D IN OPERATION	X	X
TCOE	0064	TRANSFER ON CHANNEL E IN OPERATION	X	X
TDAA	-1060	TRANSFER ON CHANNEL A DEVICE IN OF	X	X
TEFA	0030	TRANSFER ON END OF FILE, CHANNEL A	X	X
TEFB	-0030	TRANSFER ON END OF FILE, CHANNEL B	X	X
TEFC	0031	TRANSFER ON END OF FILE, CHANNEL C	X	X
TEFD	-0031	TRANSFER ON END OF FILE, CHANNEL D	X	X
TEFE	0032	TRANSFER ON END OF FILE, CHANNEL E	X	X
TMI	-0120	TRANSFER ON MINUS	X	X
TNZ	-0100	TRANSFER ON NO ZERO	X	X
TOV	0140	TRANSFER ON OVERFLOW	X	X
TPL	0120	TRANSFER ON PLUS	X	X
TRA	0020	TRANSFER	X	X
TRCA	0022	TRANSFER ON REDUNDANCY CHECK, CHANNEL A	X	X
TRCB	-0022	TRANSFER ON REDUNDANCY CHECK, CHANNEL B	X	X
TRCC	0024	TRANSFER ON REDUNDANCY CHECK, CHANNEL C	X	X
TRCD	-0024	TRANSFER ON REDUNDANCY CHECK, CHANNEL D	X	X
TRCE	0026	TRANSFER ON REDUNDANCY CHECK, CHANNEL E	X	X
TRP	-1165	TRANSFER AND RESTORE PARITY AND TRAPS	X	X
TRT	-1164	TRANSFER AND RESTORE TRAPS	X	X
TSL	-1627	TRANSFER ON ZERO	X	X
TZE	0100	TRANSFER AND STORE INSTR LOCATION CTR	X	X
VDP	0225	VARIABLE LENGTH DIVIDE OR PROCEED	X	X
VLM	0204	VARIABLE LENGTH MULTIPLY	X	X
VMA	-1204	VARIABLE LENGTH MULTIPLY AND ACCUM	X	X
WBT	0766	WRITE BLANK TAPE	X	X
WEF	0770	WRITE END OF FILE	X	X
WRS	0766	WRITE SELECT	X	X
XEC	0522	EXECUTE	X	X

DEVICE	CHAN	ADAPT	BCD ADDRESSES	BINARY ADDRESSES
MAGNETIC TAPE	A	0	01202/01212	01221/01232
	B		02201/02212	02221/02232
	C		03201/03212	03221/03232
	D		04201/04212	04221/04232
	E		05201/05212	05221/05232
DIRECT DATA CONNECTION	B		02240	02260
	C		03240	03260
	D		04240	04260
	E		05240	05260
1622 CARD READER	A	3	01210	01230
1622 CARD PUNCH	A	3	01211	01231
1402 CARD READER	A	3	01210	01230
1402 CARD PUNCH	A	3	01211	01231
1403 PRINTER	A	3	01212	01232
TYPEWRITER	A	4	01000	01020
1401 DATA PROC SYSTEM	A	5	01201/01212	01221/01232
1011 PAPER TAPE READER	A	3	01601	01621
1009 DATA TRANSMIT UNIT	A	3	01301	01321
1014 REMOTE INQUIRY UNIT	A	3	01701/01702	01721/01722
TELEGRAPH TYPE UNITS	A	3	01401/01404	01421/01424

NOTES
 F - Represents an indirectly addressable instruction
 N - Represents a number from 1 to 6

EXTENDED PERFORMANCE INSTRUCTIONS

SYM	NUMERIC	NAME	F	T
AXT	0774	ADDRESS TO INDEX TRUE		
CCS	-1341	COMPARE CHARACTER WITH STORAGE	X	X
LAC	-0535	LOAD COMPLEMENT OF ADDRESS IN INDEX		
LDC	-0535	LOAD COMPLEMENT OF DECREMENT IN INDEX		
LXA	0534	LOAD INDEX FROM ADDRESS		
LXD	-0534	LOAD INDEX FROM DECREMENT		
HIT	-1341,6	STORAGE MINUS TEST	X	X
MSM	-1623,6	MAKE STORAGE SIGN MINUS	X	X
MSP	-1623,7	MAKE STORAGE SIGN PLUS	X	X
PAC	0737	PLACE COMPLEMENT OF ADDRESS IN INDEX		
PAX	0734	PLACE ADDRESS IN INDEX		
PCS	-1505	PLACE CHARACTER FROM STORAGE	X	X
PDC	-0737	PLACE COMPLEMENT OF DECREMENT IN INDEX		
PDX	-0734	PLACE DECREMENT IN INDEX		
PLT	-1341,7	STORAGE PLUS TEST	X	X
PXA	0754	PLACE INDEX IN ADDRESS		
PXD	-0754	PLACE INDEX IN DECREMENT		
SAC	-1623	STORE ACCUMULATOR CHARACTER	X	X
SXA	0634	STORE INDEX IN ADDRESS		
SXD	-0634	STORE INDEX IN DECREMENT		
TIX	2000	TRANSFER ON INDEX		
TMT	-1704	TRANSMIT		
TNX	-2000	TRANSFER ON NO INDEX		
TSX	0074	TRANSFER ON SET INDEX		
TXH	3000	TRANSFER ON INDEX HIGH		
TXI	1000	TRANSFER WITH INDEX INCREMENTED		
TXL	-3000	TRANSFER ON INDEX LOW OR EQUAL		

DIRECT DATA CONNECTION INSTRUCTIONS

SYM	NUMERIC	NAME	F	T
PSLB	-0664	PRESENT SENSE LINE, CHANNEL B	X	X
PSLC	-0665	PRESENT SENSE LINES, CHANNEL C	X	X
PSLD	-0665	PRESENT SENSE LINES, CHANNEL D	X	X
PSLE	-0666	PRESENT SENSE LINES, CHANNEL E	X	X
SSLB	-0660	STORE SENSE LINES, CHANNEL B	X	X
SSLC	-0661	STORE SENSE LINES, CHANNEL C	X	X
SSLD	-0661	STORE SENSE LINES, CHANNEL D	X	X
SSLE	-0662	STORE SENSE LINES, CHANNEL E	X	X

1401 OPTION INSTRUCTIONS

SYM	NUMERIC	NAME	F	T
SLFA	-1760,01501	STATUS LINE OFF, CHANNEL A		X
SLNA	-1760,01541	STATUS LINE ON, CHANNEL A		X

IBM

**International Business Machines Corporation
Data Processing Division
112 East Post Road, White Plains, New York**