Burroughs

AE 500 Series

EQUIPMENT REFERENCE MANUAL

PRICED ITEM

Printed in U.S. America

February 1976

2007332

LIST OF EFFECTIVE PAGES

NOTE: Insert latest changed page; dispose of superseded pages.

TOTAL NUMBER OF PAGES IN THIS MANUAL IS 166 CONSISTING OF THE FOLLOWING:

Page No.	Issue
Title	Original
A Page	Original
i thru ix	Original
x Blank	Original
1–1 thru 1–4	Original
2–1 thru 2–7	Original
2–8 Blank	Original
3–1 thru 3–13	Original
3–14 Blank	Original
4–1 thru 4–9	Original
4–10 Blank	Original
5–1 thru 5–13	Original
5–14 Blank	Original
6–1 thru 6–10	Original
7–1 thru 7–42	Original
8–1 thru 8–18	Original
9–1 thru 9–3	Original
9–4 Blank	Original
10–1 thru 10–3	Original
10–4 Blank	Original
A1 thru A12	Original
B1 thru B4	Original
C1	Original
C2 Blank	Original
D1	Original
D2 Blank	Original
E1 thru E2	Original
F1 thru F2	Original
G1	Original
G2 Blank	Original

A PAGE

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION 1
GENERAL DESCRIPTION

INTRODUCTION

The AE500 is a new product series (figure 1–1) designed to complement and increase the throughput capability of Burroughs' line of intelligent terminal systems. The series includes four styles to provide the following:

Up to 28K Bytes of high speed memory (1 MHz)
60 CPS matrix printer
15" forms handler with pin feed
1 asynchronous or synchronous data communications line
1 or 2 magnetic tape cassette drive

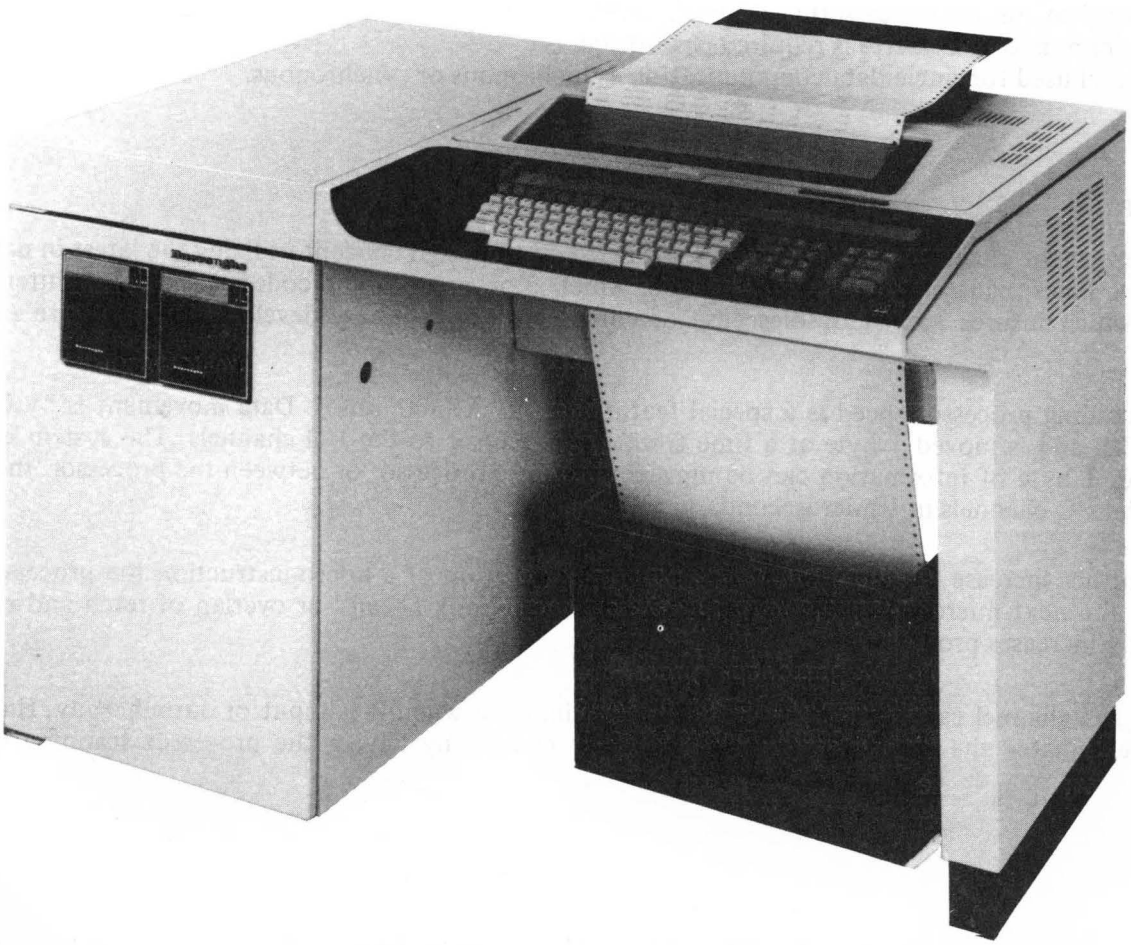


Figure 1–1. AE500 Intelligent Terminal System

MACHINE DESCRIPTION

The AE500 series is a free-standing, operator attended, intelligent terminal system designed to provide on-line as well as off-line data processing. The following features are provided:

A serial matrix impact printer which provides 60 character-per-second printing and a 150 print-position pin-feed forms handler.
A new powerful LSIC (Large Scale Integrated Circuitry) processor system employing the latest in advanced design concepts and data processing techniques.
The incorporation of specialized processors to handle I/O functions.
On-board diagnostics are provided through a Customer Confidence Program which is easily initiated by the operator.

For on-line operation, the AE500 series is equipped with a single data communication channel capable of transmitting and receiving messages in an asynchronous or synchronous mode with variable buffering. Data communication line procedures are provided, and operation may be on voice grade switched or dedicated lines, as well as on a direct connect interface.

I/O CHANNELS

There are 4 dedicated I/O channels which are used as follows:

1 channel to control the console printer and the pinfeed
1 channel used for keyboard input
1 channel to control the cassette subsystem (A9490-21).
A minimum of 1 Cassette is required per AE500.
1 channel used for single data communication asynchronous or synchronous.

PROCESSOR

The processor characteristics include an advanced LSIC design concept utilizing the latest in data processing techniques, nano coded micro-instructions which allow an optimum code set, and a soft interpretive structure which insures against obsolescense of equipment through easy development of future enhancements.

Outstanding processor speed is a special feature of the AE500 series. Data movement is byte serial, 8 bit parallel and is moved 1 byte at a time from the processor to the I/O channels. The system clock is 1 MHz and, 1 byte of information can be moved within the processor or between the processor, the memory, and the I/O channels in 1 microsecond.

To further increase the processing power, during execution of a micro-instruction the processor will "look" at the next micro-instruction to be executed. This "look ahead" or overlap of fetch and execute, significantly increases processor throughput.

Each I/O channel can interrupt the processor to indicate when it is input or output ready. Hardware interrupt eliminates the processor overhead normally created by having the processor scanning the I/O channels.

MEMORY

The AE500 series incorporates MOS LSIC (Metal Oxide Semiconductor Large Scale Integrated Circuits). Memory is modular in 4KB increments and consists of 4KB of ROM (Read only Memory) used for Interpreter bootstrap (cold start) and permanent customer confidence programs, plus up to 28 KB of RAM (Random Access Memory) available for interpreter and user area.

KEYBOARD

A standard electronic keyboard provides the communication link between the operator and the AE500 system. This keyboard has an alphanumeric typewriter keyboard for individual national needs in addition to separate ten-key numeric keyboard and special function keys. The keyboard includes an upper row of 16 Program Select Keys (PSK) to implement various program options.

OPERATOR COMMUNICATION SYSTEM

The AE500 series have an easy to read operator communication system on the front panel above the keyboard (figure 1–2). These indicators display the status of the AE500 its peripherals, and the data communication operations, as well as the availability of the program select keys for the operator.

The PSK indicator lights, which are located underneath the PSK strip, light when a PSK is enabled. The PSK strip consists of a translucent material on which descriptions may be typed to enable flexibility for individual program requirements.

PRINTER

The AE500 has a serial matrix impact printer which prints at 60 characters-per-second. Each character is formatted on a 7 x 7 matrix. The dot matrix print is of outstanding quality and one original and up to five copies can be printed. A single color (black) ribbon is provided.

To provide maximum print throughout, the system has separate keyboard and print buffers. There are 150 print positions per line with printing at 10 characters per inch. Positioning is bi-directional at an average of 160 characters-per-second.

FORMS HANDLER

The unit is equipped with a single pin feed device for handling forms from 3.0" (76mm) to 16.75" (425mm) wide. It is capable of handling fanfold, single, or multiple part forms with folds from 3.5" (89mm) to 12" (305mm). A scale over the printing surface guides the operator when determining print position. Forms can be manually spaced and finally adjusted by the operator. The forms are advanced in 1/6" increments at 6 lines per second or by vertical slew at 30 lines per second.

For complete visibility of the print line, a double line advance is automatically performed if no print has occurred for a few seconds. This elapsed time is programmable through a systems register. Before the next printing occurs, the form is automatically reversed to the original print line.

The continuous form feed mechanism provides a fast and easy method of changing loading forms.

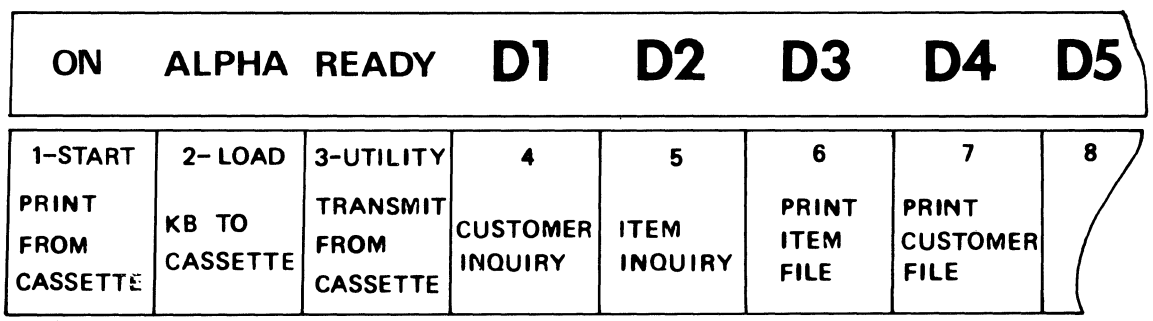


Figure 1–2. Panel Indicators

MAGNETIC TAPE CASSETTE SUBSYSTEM

Up to two cassette stations can be housed in the AE500 system. At least one cassette station is required for the Basic Memory loading. After memory loading it may be used as an additional data storage device to provide the following advantages:

High capacity – Storage capacity per cassette is 204,800 characters per 300 feet of tape cassette. Total system memory can be stored on approximately 40 feet of cassette tape.

High speed – Read/write speed is 10 inches/second, search speed is 30 inches/second, rewind speed is 60 inches/second. Approximate time to load the full memory capacity of the AE500 is 60 seconds. This enables rapid changing of programs, including all system registers, at the terminal site.

Integrity – The cassette system incorporates separate read and write heads to permit read after write validity checking, as well as separate data and clock synchronization tracks to insure portability between systems.

Multi-function capability – can be used for interpreter loading, program loading, and/or the storage of program data.

DATA COMMUNICATIONS

The AE500 system will communicate either in the asynchronous or synchronous mode with a central computer or another terminal over leased or switched lines or by Two-wire Direct Interface (TDI) up to 1,000 feet (300 metres) or by Burroughs Direct Interface (BDI) up to 15,000 feet (4,500 metres).

The Line Protocols available with this system include Burroughs Basic Mode procedure Point to Point Batch, and the new bit-oriented Burroughs Data Link Control procedures.

SYSTEMS SOFTWARE

The systems software of the AE500 series provides for the execution of programs whose object code is in System Language 3 or System Language 5 (SL3 or SL5). The source program is written in COBOL and may be compiled into SL3 or SL5 code using compilers available on Burroughs Medium Systems.

The software also provides utilities and loaders for configuring the software according to the user's needs and general routines for data transfer from one peripheral device to another.

SECTION 2
PROCESSOR AND MEMORY

GENERAL

The purpose of this section is to provide an understanding of the AE500 processor and memory, and to describe the processor to memory and processor to input/output interface.

The processor, memory, peripheral control, and power supplies are located in the processor cabinet, together with one or two cassette drive units. The processor furnishes the computing power for the system software and for the user programs, and provides the communication path between memory and input/output. It operates under the control of micro-instructions stored in memory. Memory provides storage for the user programs, data, and system software.

PROCESSOR

A simplified block diagram of the processor is shown in figure 2–1.

The processor is constructed around a function unit which provides 32 data processing operations. Sixteen of these are algebraic operations and provide the calculating power of the processor. The remaining 16 are boolean operations, and provide the ability to make logical decisions. The function unit has two 8-bit wide input buses, and one 8-bit wide output bus.

Fifteen special purpose registers are provided within the processor. These registers are used by micro-programs to address memory, store intermediate results, and control input output operations.

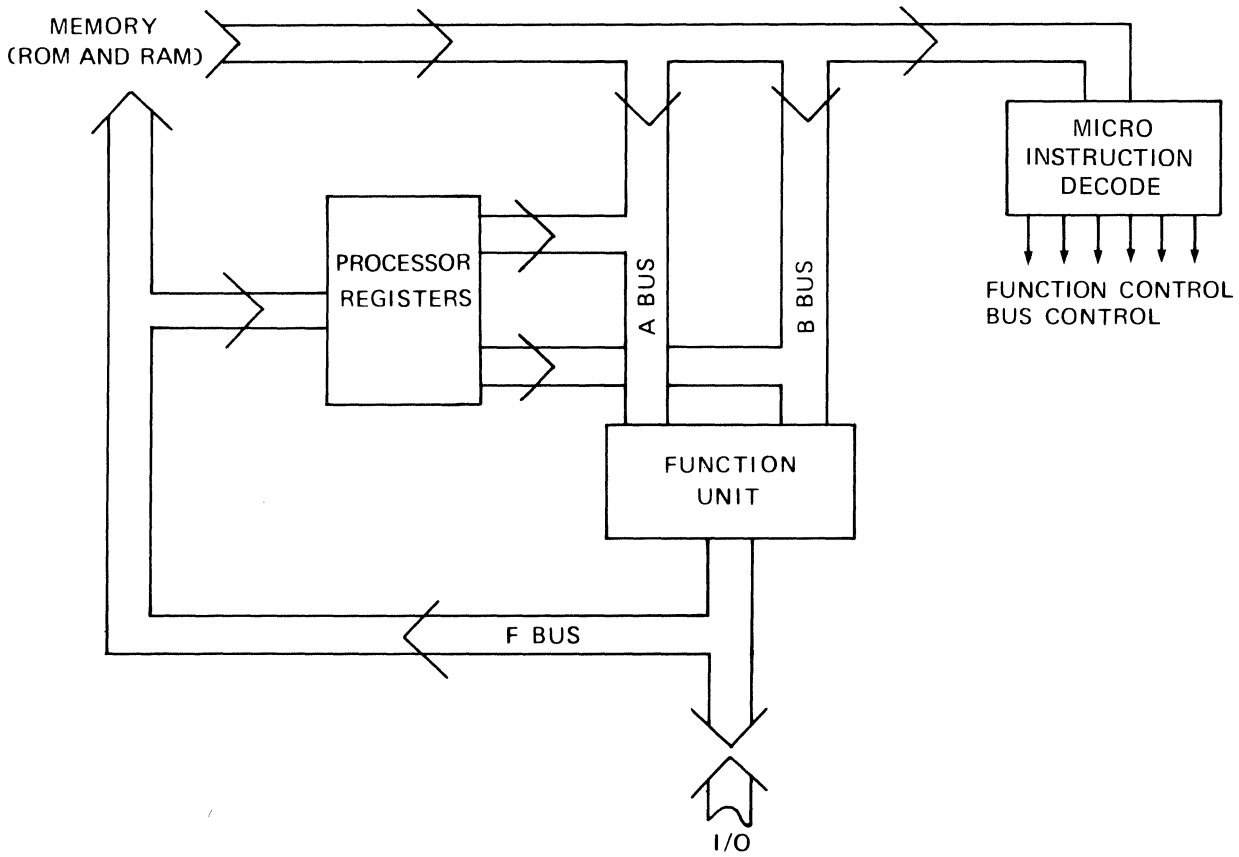


Figure 2–1. The Processor

The processor operates under control of micro-instructions stored in memory. While power is applied to the machine, timing machine state logic within the processor continuously fetches micro-instructions from memory (starting at memory address 0 when power is first applied). The micro-instruction decode logic translates the micro-instruction into control signals for the processor data paths and the function unit. Two buffers are provided in the decode logic to improve processor throughput. One buffer stores the micro-instruction currently in execution. The second buffer permits the next micro-instruction in execution sequence to be fetched from memory, and partially decoded, at the same time as the current instruction is executing. Figures 2–2 and 2–3 show the overlapped fetch and execute, and the look ahead decode, in comparison with conventional timing.

The micro-instruction format used is of variable length, the length being dependent upon the requirements of the micro-instruction. This permits micro-programs to reside in a smaller area of memory than would be possible if the micro-instructions were of fixed format. A large range of operations are provided by the micro-instructions from single bit access to multiple byte data movement.

The processor architecture is configured around three data buses A, B, and F. The A and B buses permit two source fields to be input to the function unit, and the F bus directs the subsequent result to a destination field. Micro-instruction decode signals open paths from processor registers, memory, and Input/Output to the three buses. All data paths and buses within the processor are 1 byte (8 bits) wide, and memory is accessed in 1 byte increments.

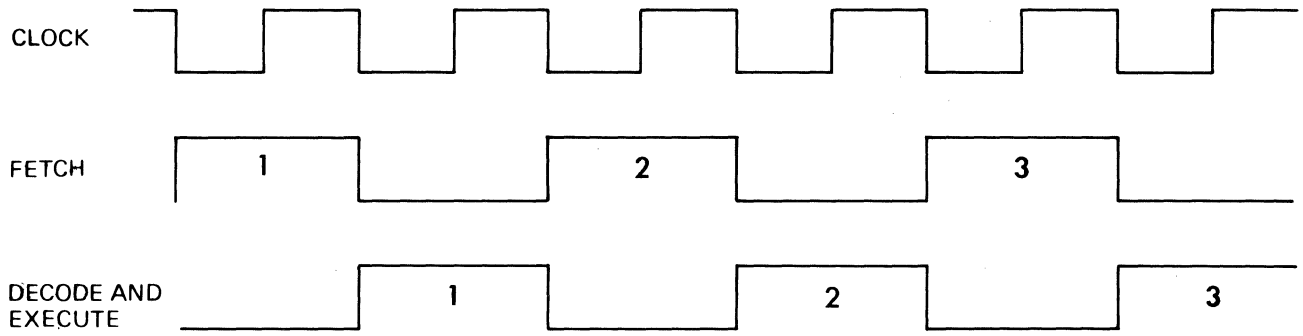


Figure 2–2. Conventional Timing

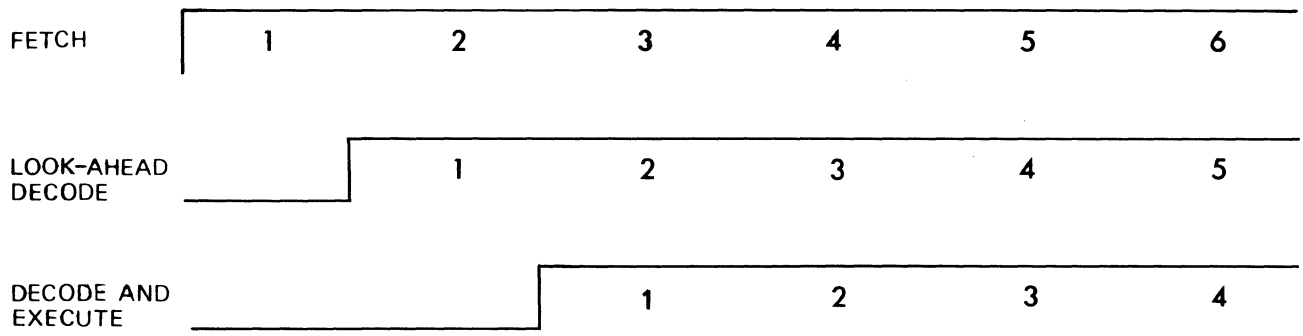


Figure 2–3. Overlapped Timing

All the processor registers can be stored in memory, permitting the resources of the processor to be shared by discrete processes.

An important characteristic of the processor is the provision of hard (or hardware) interrupt. Through this mechanism, peripheral devices can be allowed to interrupt the processor's current task, to ensure immediate responses to Input/Output (I/O) requirements.

The processor is implemented in Large and Medium Scale Integrated Circuits (LSIC and MSIC) for compactness and reliability.

Nine LSIC's provide:

27 bytes of storage organised as 15 special purpose registers
The Timing Machine State Logic (TMS)
Micro-Instruction Decode Logic
Micro-Instruction Address Pointer and a 4 level stack

Three MSIC's provide:

The function unit
The look-ahead logic
Micro-instruction buffers
Memory and Input/Output interfaces

The processor clock operates at 1MHz, and is derived from a 20MHz crystal controlled master oscillator. The processor clock determines the basic speed at which micro-instructions are performed; in 1μsec the processor can perform any of the 32 functions provided by the function unit upon 1 byte of data, taking as source fields memory or processor registers, and directing the result to memory, I/O or processor registers. (Data from I/O is transferred to memory or processor registers directly from the F Bus).

MEMORY

Two types of memory circuits are used in the memory; Read/Write (known as random access memory or RAM), and read only memory (ROM). Both types of memory are provided on printed circuit boards (PCB's) in 4K byte modules. The printed circuit boards have identical interfaces with the processor and are used to provide a contiguous memory of up to 32K bytes, as shown in figure 2–4. Random access memory provides fast access storage (1μsec) for user program, data and system software. Data stored in RAM cannot be retained when machine power is turned off. For this reason, read only memory is used to provide the basic load routine for the system memory. ROM contents are permanently written at the time of manufacture.

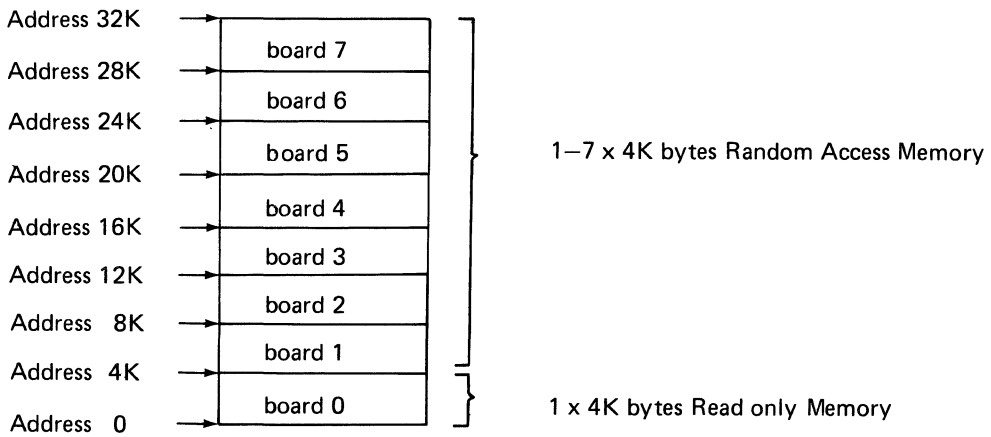


Figure 2–4. Physical Memory Organisation

RANDOM ACCESS MEMORY (RAM)

Random access memory (RAM) is a type of large scale integrated circuit (LSIC) memory which may be written to and read from. It is frequently referred to as Read/Write memory.

RAM provides fast access memory for user programs, data, and system software. The memory cycle time (read or write) is 1μsec, and memory may be accessed every microsecond. Each memory access cycle will transfer 1 byte (8 bits) to or from memory.

Up to 7 RAM boards may be supplied in a system, to provide 28K bytes of read/write memory. Each RAM board contains 4K bytes of random access memory, organised as 4096 x 9 bits (8 data, 1 parity). A system with less than 28K bytes of RAM may have further 4K RAM boards installed at any time.

Two memory interface printed circuit boards (PCB's) are required to permit the processor to access memory. These boards provide the circuitry to convert a memory address from the processor into "board" and "memory cell" select lines, and to cycle the selected memory cells to yield or replace their contents. Data is input to memory from the processor, or input to the processor from memory through a byte wide data highway. The data highway is gated into the processor A, B, or F buses by control signals from the micro-instruction decode logic.

Circuitry is provided in the memory interface PCB's to generate and check the value of the memory parity bit. Every byte of data stored in memory has an additional bit associated with it which ensures the integrity of the stored data. The memory interface circuits accept 1 byte (8 bits) from the processor, and computes the value of a 9th bit (parity) which is stored in conjunction with the data. The same computation is performed when data is taken from memory, and if the current computed value does not agree with the stored value, then a flag is set for micro-program interrogation, and an interrupt is generated. This interrupt will be "trapped" by the ROM micro-program, as described in the ROM subsection of the section. If the computed and stored values agree, then the parity bit is discarded, and the stored byte is passed to the processor. The memory read cycle is non-destructive.

A further function of the memory interface circuits is to refresh the contents of each memory cell at least once every 2 milliseconds (0.002 seconds), since random access memory contents decay if this is not done. Only 32 refresh cycles are required to refresh all of memory, and each cycle takes 1 microsecond (1×10^{-6} seconds). The memory refresh circuits perform refresh cycles whenever the processor does not require memory access. Thirty-two one microsecond periods which do not require memory access are usually available in 2 milliseconds (2000 microseconds) of processing time. If the end of a 2 millisecond timing period is reached and 32 refresh cycles have not been performed, then the processor is prevented from accessing memory until the remaining few cycles have been performed.

Certain parts of RAM are initially loaded with system software, using the bootstrap load micro-program in ROM. This procedure is described in Section 6. System software provides a user oriented interface to the machine, and permits interpretation of user programs in languages which cannot be interpreted directly by the hardware. System software is described in Sections 6 and 9 of this manual.

Once system software has been loaded, it will be used to load and execute user programs. User programs in RAM are addressed relatively, that is, addresses within the user program are maintained by system software relative to the start of the program, and have no connection with the absolute addresses as depicted in Figure 2–4. In fact, the memory address format used in programs will normally be different from the absolute memory address format. User program address formats are described in Section 6 of this manual.

Data for user programs will also be stored in random access memory, and may be brought into memory as required by the user program, using the high speed I/O peripherals available to the system.

READ ONLY MEMORY (ROM)

Read only memory (ROM) is a type of large scale integrated circuit (LSIC) memory whose contents are fixed at the time of manufacture. Its primary function is to store the "bootstrap load" micro-program by which the system software is loaded into RAM. ROM contents will not be changed by a memory write operation, and will be retained with power removed from the machine.

One ROM board is supplied in a system, containing 4K bytes of read only memory organised as 4096 x 10 bits (8 data, 1 parity, 1 parity inhibit). No further ROM boards need to be installed in a machine, since ROM is provided to initially load RAM.

Read only memory is contiguous with random access memory, has the same interface with the processor (the memory interface is described in the Random Access Memory Sub-Section), and is identical in operation to random access memory. The only difference is that ROM contents will not change if a write operation is performed.

Figure 2–4 shows that ROM encompasses absolute memory addresses 0–4095. When power is applied to the machine (or if the cold start button is pushed) micro-program execution will start at absolute address 0. The micro-program in ROM will effect the following:

1. – Determine why the ROM program has been initiated.

   If an Input/Output device has interrupted, control is passed to System Software routines in RAM.

   If a memory parity error has occurred, the ROM program will loop, alternately illuminating and extinguishing the keyboard indicators. The cold start button must be pressed to escape from this loop.

   If neither of the above conditions exist, the program advances to the next step below.

   – Clears random access memory (write zeros to memory starting at absolute location 4096). This is to ensure that no parity errors are present in memory.

   – Determine the physical size of random access memory available to the system. This information is required by system software, to prevent user programs from attempting to access non-existent memory.

   – Scan the peripherals available on the system. System software maintains tables cross referencing I/O devices and I/O channels to enable user programs to address devices rather than channels.

   – Provide a Keyboard halt permitting the operator to select the Cold Start Procedure.

   – Load random access memory with the first part of system software, and transfer control from ROM to RAM.

   The operation of the Cold Start Procedure is described in Section 6.

2. – Micro-programmed subroutines are stored in ROM. These routines are general purpose routines used by the system software. They include functions such as binary to decimal conversion and user program relative address to absolute machine address translation.

3. – The first part of the customer confidence program is stored in ROM. Operation of the System Test switch alters the memory address circuits to select an alternative absolute address 0 within ROM. When power is applied to the machine (or if the cold start button is pushed) micro-program execution will start at the alternative absolute address 0. The micro-program will test the processor, memory, and sufficient I/O to permit the full confidence program to be loaded. A description of the customer confidence program is to be found in Section 9.

INPUT/OUTPUT INTERFACE

The input/output interface provides the communication link between the processor and I/O devices. The interface is identical for each I/O device.

The input/output interface can be considered in two parts:

the I/O control, and
the device dependent controllers.

I/O CONTROL

The I/O control controls the data highway between the processor and device dependent controllers, and provides five control signals to each channel (ADDR, REQ, DIRN, CNTL, and IOX) as detailed later in this section.

The I/O control permits up to five devices to be attached to a system. Figure 2–5 shows the I/O control and the common interface to each channel.

DATA HIGHWAY

An 8-bit bi-directional data highway connects the processor through a data distributor which provides 8 input and 8 output lines to each device controller. The data highway is gated onto the processor F bus by control signals from the micro-instruction decode logic. Data/Control information present on the data highway is available to all device controllers. The device controller which will recognise the information is determined by the Address lines (ADDR), as described later in this section. The type of information (control or data) and the direction of transfer are determined by the signals CNTL and DIRN.

Address Lines (ADDR)

A special purpose register within the processor may be loaded under micro-instruction control to select one of the five I/O device controllers. All subsequent I/O communication will be with this device controller, until the register is reloaded. The register provides a separate address line to each I/O channel.

Request Lines (REQ)

A signal line is provided from each device controller which may be gated onto the processor A bus for examination by system software micro-programs. A device controller will raise its request line when it requires processor attention, either for a data transfer, or because it is in an error condition.

Micro-program control may permit any or all request lines to produce a hardware interrupt, through the use of a special register, MASK. Bit positions 0–4 of the MASK register are gated with corresponding request lines 0–4. If a request is present on a channel, and the corresponding MASK bit is set, the interrupt signal is generated. The interrupt signal will cause a micro-program subroutine to be invoked, which will determine the origin of the interrupt and pass control to the appropriate micro-program handler.

Direction (DIRN)

This signal is derived from the micro-instruction decode, and determines the direction of data or control information transfer during an I/O communication with the processor.

Control (CNTL)

This signal is derived from the micro-instruction decode, and indicates that the data highway will be used to pass control information. Normally when control information is being passed,direction (DIRN) will be from the processor to I/O. A special case is "Status Read", when CNTL and DIRN indicate control information from I/O to processor.

Input/Output Exchange (IOX)

This signal is derived from the micro-instruction decode and permits traffic along the data highway.

DEVICE DEPENDENT CONTROLLERS

The function of device dependent controllers is to translate processor commands into timed, specific control of the device. The complexity of a device controller is therefore governed by the requirements of the device. Device controllers may be required to perform many sequenced steps in the execution of a processor command.

The interface between the processor and each device controller is identical, in order to permit rapid installation of additional I/O devices at any time. The common interface is shown in figure 2–5.

The device controllers for peripherals requiring complex control are unusual in that they incorporate a powerful micro-programmed processor, similar to the main system processor. These device controllers are extremely versatile, and require little main processor intervention. Up to 8K bytes of independent read only memory program are provided for these controllers. This memory is located on the device controller PCB, and is in addition to the system memory. Currently, the cassette drive units and synchronous data communications are provided with these controllers. The power of these controllers is illustrated by the fact that both cassettes are handled by a single controller. The use of these controllers speeds up system software by taking the responsibility for peripheral control away from the main processor.

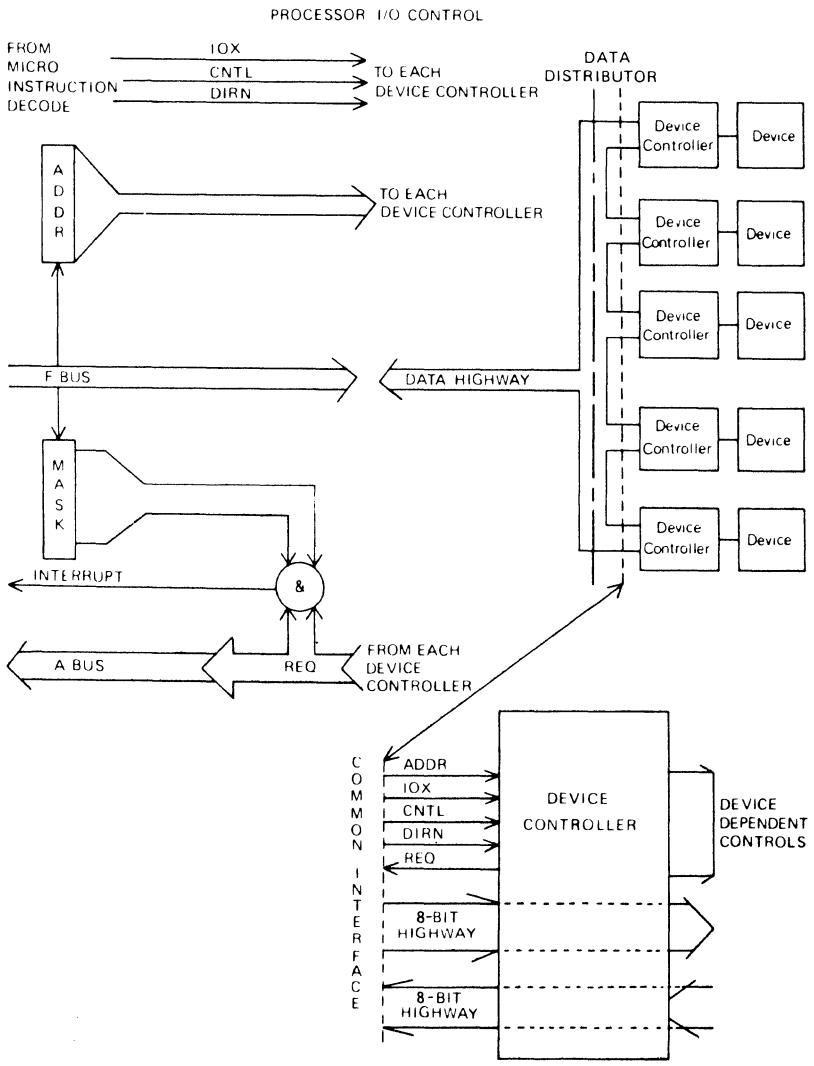


Figure 2–5. Input/Output Interface

SECTION 3
CONSOLE

The AE500 console consists of a 64 character keyboard, a matrix printer, and a forms handler. Each unit is described in this section.

KEYBOARD

Each key on the keyboard is used to transmit its own unique code to the processor. The code is then transferred and stored in the keyboard buffer. The use of that code is determined by the interpreter which may treat it as an Alpha character, a numeric digit or as a functional code.

The numeric keyboard, typewriter keyboard, program select keys and indicators are described below:

THE READY KEY

The ready key is the single rectangular key located above and to the left of the Program Keys. The basic function of the ready key is to effect a change from one system state to another. For details of the use of the ready key in each system state, refer to Section 6.

NUMERIC KEYBOARD

The numeric keyboard shown in figure 3–1 consists of 12 numeric keys, a decimal point (.) key, RESET key, per hundred (C) key, per thousand (M) key, reverse entry (RE) key and four operation control keys (OCK's).

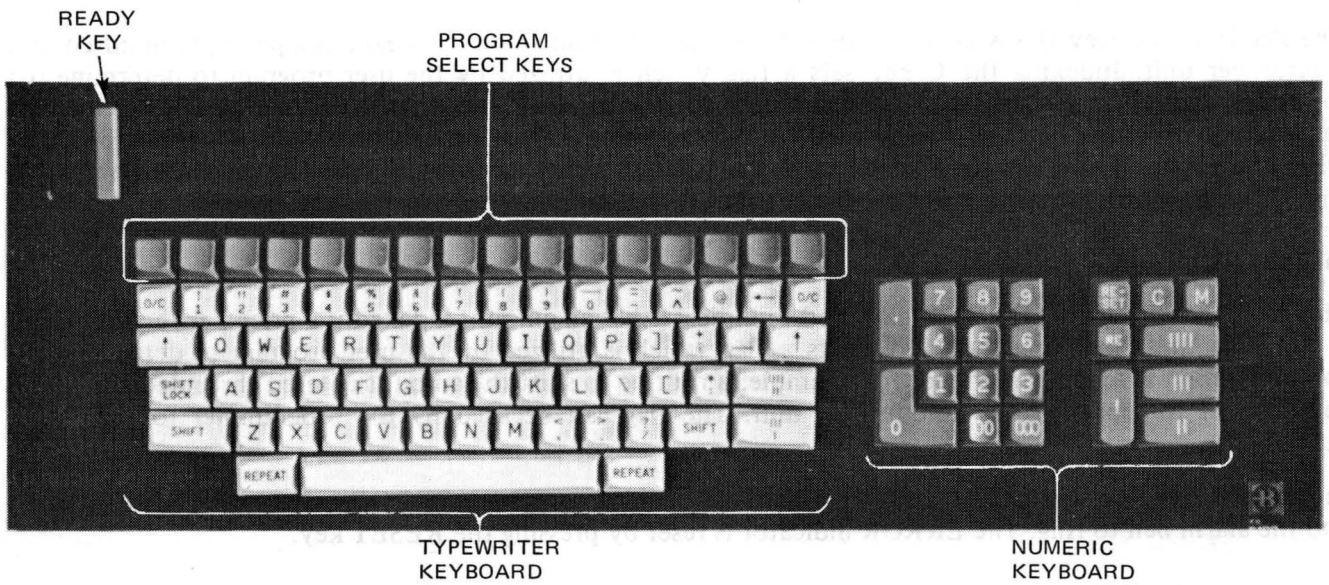


Figure 3–1. AE500 Keyboard

Numeric Keys

The numeric keys consist of keys marked 0 through 9, a double zero key (00), and a triple zero key (000). A raised dot on the 5 key allows the operator to determine the key positions by touch.

Decimal Point (.) Key

The program instruction which accepts the data from the buffer determines where the decimal point occurs in the numeric field. Numeric digits indexed before indexing the decimal key are aligned to the left of the decimal point in the field defined. Numeric keys indexed after indexing the decimal key are aligned to the right of the decimal point in the field defined. If more numeric keys are indexed before or after indexing the decimal key than are specified by the program instruction, the ERROR indicator is illuminated and the alarm bell sounds. The error condition is reset by pressing the RESET key.

Reset Key

The RESET key, located on the numeric keyboard, is used to correct keyboard errors. When used with a numeric keyboard instruction after a keyboard error alert (ERROR indicator on) or prior to the use of an operational control key, it performs the following functions:

1) Clears the accumulator.
2) Extinguishes the ERROR light, if the light is on.
3) Re-initiates the active program instruction requiring the numeric data.

The RESET key can also be used with a typewriter keyboard instruction to correct errors or re-initiate the instruction. If the system is in an input condition (ALPHA indicator on without an ERROR condition), pressing the RESET key re-initiates a typewriter keyboard instruction and moves the print head to the start position; subsequent entry of data destroys and replaces the previously entered data. If an error has occurred (ERROR indicator on), pressing the RESET key extinguishes the ERROR indicator but does not move the print head. To re-initiate the typewriter keyboard instruction, the RESET key must be pressed again; subsequent entry of data destroys and replaces the previously entered data.

Per Hundred Key (C)

The Per Hundred Key (C) is normally used to indicate that an amount indexed is a price per hundred, not a price per unit. Indexing the C key sets a flag which is utilized by the user program to determine if a special routine is to be performed, for example, a routine to divide the amount entered by 100 to calculate the amount per unit. The flag can be used by the programmer for any other purpose. Indexing the C key when the C and M keys are not enabled cause the ERROR indicator to be illuminated and the alarm bell to ring. The ERROR indicator is reset by pressing the RESET key.

Per Thousand Key (M)

The Per Thousand Key (M) is normally used to indicate that an amount indexed is a price per thousand, not a price per unit. Indexing the M key sets a flag which is utilized by the user program to determine if a special routine is to be performed, for example, a routine to shift the amount entered three decimal places to the right to represent the amount per unit. The flag can be used by the programmer for any other purpose.

Indexing the M key when the C and M keys are not enabled causes the ERROR indicator to be illuminated and the alarm bell to ring. The ERROR indicator is reset by pressing the RESET key.

Reverse Entry Key (RE)

Indexing the Reverse Entry Key (RE) changes the sign of the data entered from plus to minus. This facility for using negative data allows proper arithmetic and can be used to condition the printing of special symbols to indicate negative amounts.

Indexing the RE key when it is not enabled causes the ERROR Indicator to be illuminated and the alarm bell to ring. The ERROR indicator is reset by pressing the RESET key.

Operation Control Keys (OCK)

The numeric keyboard contains four Operation Control Keys (OCK I, II, III, IIII) that are used to terminate typewriter (alpha) or numeric keyboard entries. Indexing an OCK sets a corresponding OCK flag, and this flag may be used in the user program to condition certain routines or functions. OCK I is raised above the plane of the other numeric key tops for easier access and can be used as the "normal" terminator (depending on program requirements).

TYPEWRITER KEYBOARD

The typewriter keyboard shown in figure 3–1 consists of 48 typing keys, 2 SHIFT keys, a SHIFT LOCK key, 2 line advance keys (↑), 2 REPEAT keys, a space bar, a backspace key (←) and 2 operation control keys.

The program instruction which accepts the data from the keyboard buffer determines the format and type of data required.

When a program instruction calls for an alphanumeric data entry, the alphanumeric keyboard indicator (ALPHA) is lit. If the numeric keyboard is used when a typewriter keyboard entry is required, entries (except operation control keys and the RESET key) are rejected, the ERROR Indicator light is illuminated and the alarm bell sounds. The ERROR indicator is reset by pressing the RESET key. If the keyboard buffer capacity is exceeded, the ERROR indicator light is illuminated and the alarm bell rings. The ERROR indicator is reset by pressing the RESET key.

Typing Keys (64 Character)

The typing keys consist of 32 keys which transmit the same code whether in the shift or non-shift position and 17 keys which transmit a different code depending on whether they are used in a shift or non-shift position (Refer to Appendix B – KEYBOARD VARIATIONS).

Shift Keys (SHIFT)

Either of the two Shift Keys (SHIFT) when pressed with any shift character key, shifts the character to upper case. A SHIFT key used with any non-shift character key is ignored. Depression of a SHIFT key will disable the Shift Lock feature.

Shift Lock Key (SHIFT LOCK)

When the SHIFT LOCK key is pressed, all subsequent characters are shifted to upper case until the SHIFT key is depressed.

Line Advance Keys (↑)

The left key marked "↑" causes the form in the forms transport to be vertically spaced 1/6". The right key has no effect on form movement.

Repeat Keys (REPEAT)

These keys are pressed at the same time as a repeatable key to cause that key to be repeated. All keys are repeatable except the operation control keys and the numeric keys on the numeric keyboard.

Space Bar (Unmarked)

The Space Bar causes the printer carrier to space 1/10 inch to the right if printing is enabled.

Backspace Key (←)

The lower case backspace key (←) causes the printer carrier to space 1/10 inch to the left if printing is enabled. If the typing instruction is entering the data into memory, the backspace key also decrements memory one character position for each 1/10 inch backspace. The upper case backspace key (←) causes the printer carrier to space 1/10 inch to the left if printing is enabled. If the typing instruction is entering data into memory, the shifted backspace key does not decrement memory.

Operation Control Keys (OCK)

The operation control keys (OCK) on the typewriter keyboard perform the same function as the OCK's on the Numeric Keyboard, that is, to terminate a keyboard entry. Indexing OCK I with the upper shift key becomes OCK III; OCK II with the upper shift key becomes OCK IIII.

PROGRAM SELECT KEYS

Sixteen Program Select Keys (PK's) are provided to permit operator selection of alternate routines and functions. These unmarked blue keys are located immediately above the typewriter keyboard and from left to right are PK1 thru PK16.

An indicator light associated with each PK informs the operator which key(s) has been enabled, and an identification strip shows the PK and indicator functions. The identification strip can easily be changed by the operator when PK functions change for a different application, or are reassigned within the same application. The clear plastic cover is lifted off the legend strip holder, the paper legend strip is changed and the plastic cover is replaced. Pressing a non-enabled PK, while executing a user program, causes the ERROR indicator to light and the alarm bell to ring. Section 6 describes the functions of the PK's within each System State.

INDICATORS

Up to 50 indicator lights are provided to communicate information to the operator as shown in figure 3–2. These lights are arranged in two rows, with the lower row corresponding to program select keys. Each of these indicators is located above its corresponding program select key (PK) and when illuminated, informs the operator that the PK is enabled.

The upper row of indicator lights convey special information as described below:

Group S Indicators

ON indicator (Green)
This indicator located above program select key indicator 1, shows that the power is applied to the unit.

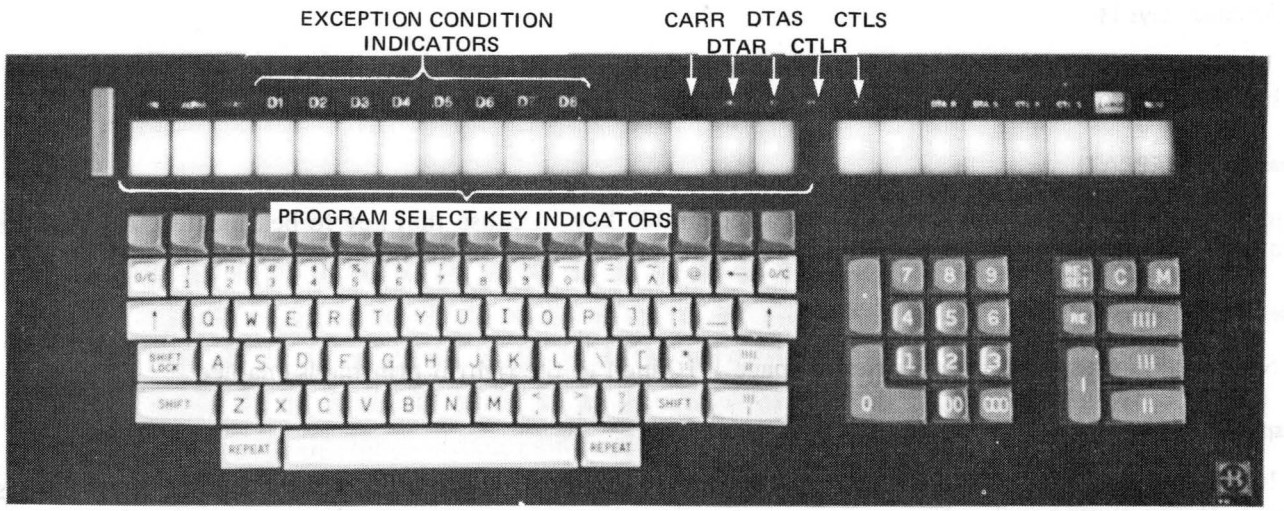


Figure 3–2. Indicator Lights

ALPHA Indicator (Amber)
This indicator, located above program select key indicator 2, shows that an alpha keyboard entry is permitted.

READY Indicator (Green)
This indicator, located above program select key indicator 3, shows that the machine is in the Ready State, that is, the operator can execute the warm start procedure, as described in Section 6.

ERROR Indicator (Red)
This indicator, located on the right side of the keyboard shows that certain errors have occurred.

NUMERIC Indicator (Amber)
This indicator, located on the right side of the keyboard shows that a numeric keyboard entry is permitted.

Exception Condition Indicators

Exception Condition Indicators are located above program select key indicators 4 through 11 and can be used in different ways. Appendix C lists exception condition indicator light patterns and reasons for errors.

Data Communications Indicators

Five indicator lights are provided for Data Communications. These lights are blue and represent the Data Communications status. The Data Communications Indicators are shown below:

CARR - This indicator signifies the presence of the carrier frequency. The function of the remaining lights are dependent on the line procedure in use, but are typically:

DTA R - This indicator is switched on by the line procedure after successful receipt of a message and is extinguished by the program when the incoming message is unloaded from its buffer.

DTA S - This indicator is lit by the user program when the output buffer is loaded, and is extinguished by the line procedure when the data has been successfully transmitted.

CTL R - This indicator is lit and extinguished by the line procedure in response to receipt of certain control codes which are line procedure dependent.

CTL S - This indicator is lit and extinguished by the line procedure in response to transmission of certain control codes which are line procedure dependent.

CONSOLE PRINTER

The console printer is a serial impact matrix printer which graphically constructs characters using a needle print head.

Examples of the printed characters are shown in Figure 3–3.

The print head shown in figure 3–4 is moved laterally to any of the 150 print positions by the carrier drive. Positioning in either direction is performed at a minimum average speed of 16 inches (40.64 cm) per second for a long displacement (9 inches or more). The maximum time for a carriage return is 1 second.

PRINT CHARACTERS

The basic 64 characters set consists of letters A through Z, numerals 0 through 9 and 27 symbols as shown in figure 3–5. The keyboard variations are shown in Appendix B.

The nominal character is 0.102 inches (2.591 mm) high and 0.070 inches (1.778 mm) wide.

Horizontal character spacing is 0.1 inches (2.54 mm) ± 0.007 inches (0.178 mm) measured between vertical center lines of adjacent characters. The maximum accumulated tolerance is 0.018 inches (0.457 mm) over 150 print positions.

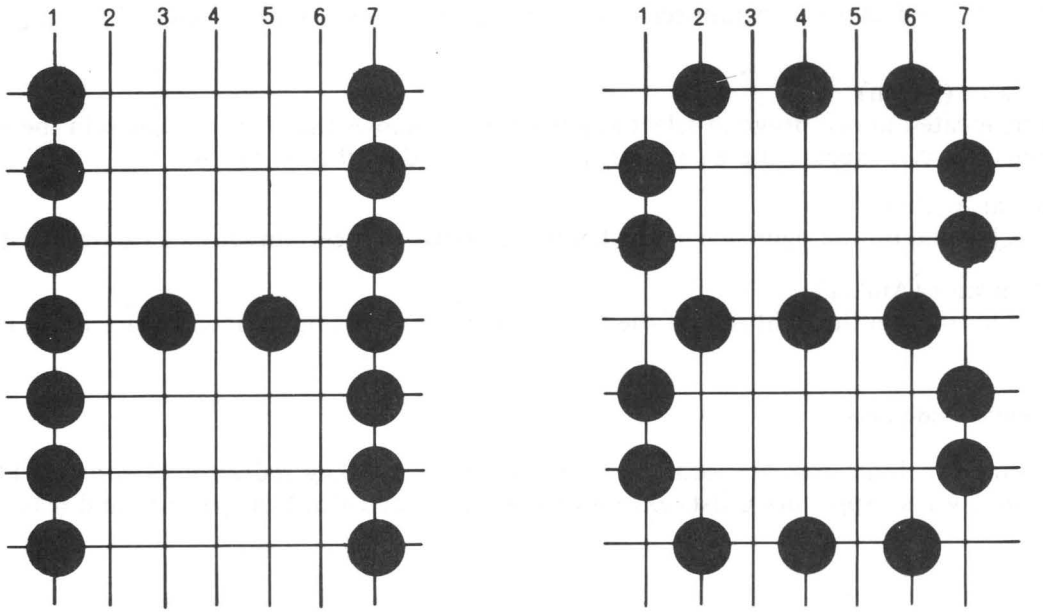


Figure 3–3. AE500 Printed Characters

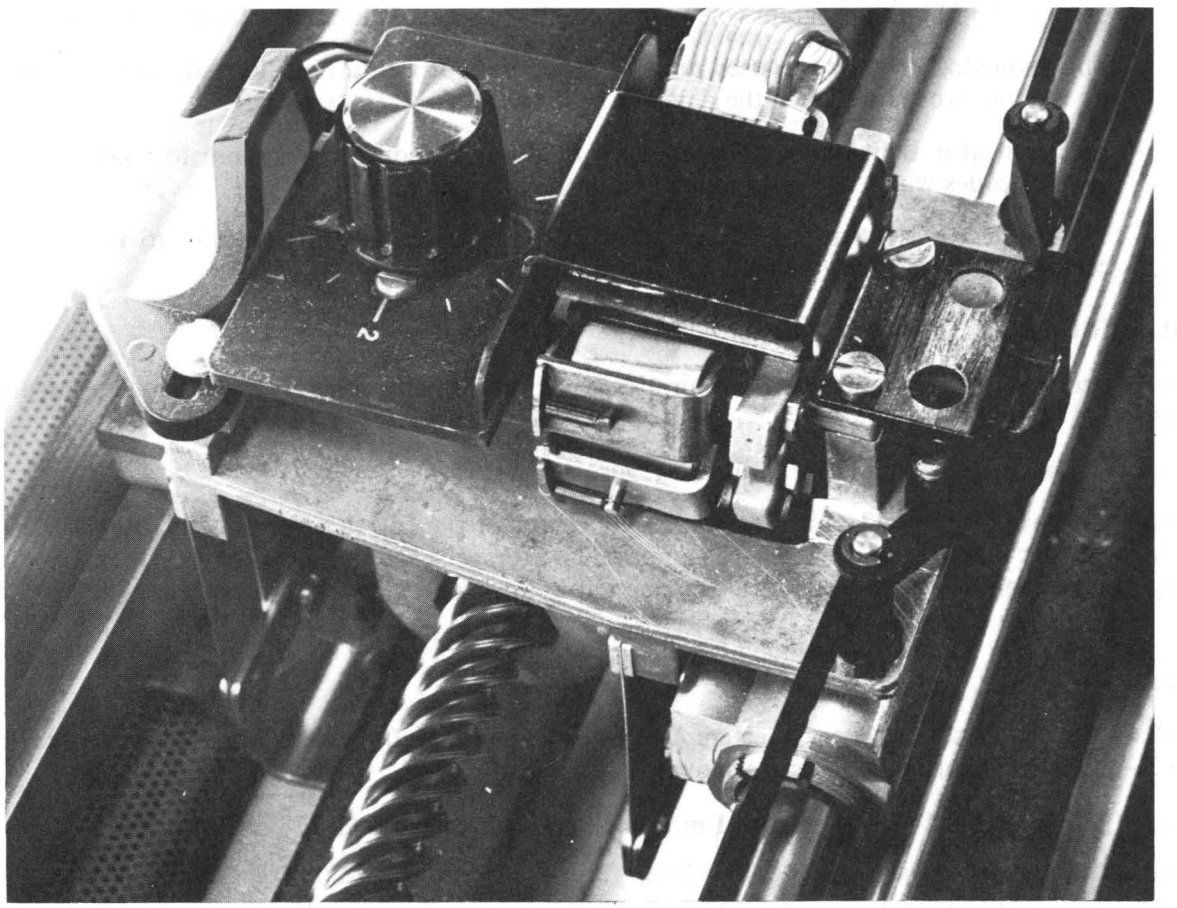


Figure 3–4. Console Printer

Forward and backward spacing is in increments of 0.1 inches (2.54 mm). Punctuation requires 0.1 inches, (2.54 mm) of printing area.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z				1 2 3 4 5 6 7 8 9 0	
.	PERIOD (DECIMAL POINT)	]	CLOSING BRACKET	=	EQUALS
,	COMMA	$	DOLLAR SIGN	\	REVERSE SLANT
;	SEMICOLON	%	PERCENTAGE SIGN	<	LESS THAN
:	COLON	&	AMPERSAND	>	GREATER THAN
"	QUOTATION MARKS	#	NUMBER SIGN		
!	EXCLAMATION MARK	@	COMMERCIAL AT	^	CIRCUMFLEX
?	QUESTION MARK	+	PLUS SIGN		
(	OPENING PARENTHESIS	/	SLANT	-	HYPHEN (MINUS SIGN)
)	CLOSING PARENTHESIS	*	ASTERISK	'	APOSTROPHE (ACUTE ACCENT)
[	OPENING BRACKET				

Figure 3–5. AE500 Basic Character Set

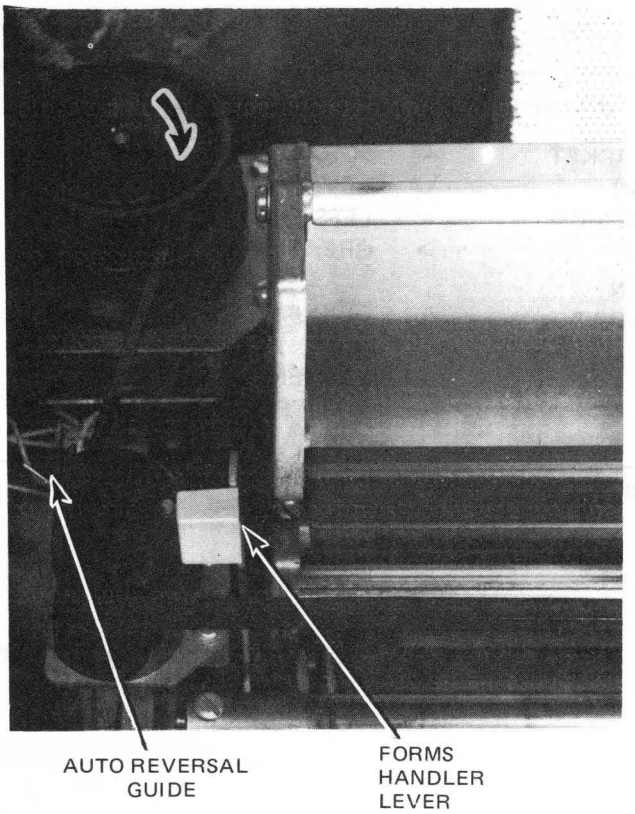
PRINTER RIBBON

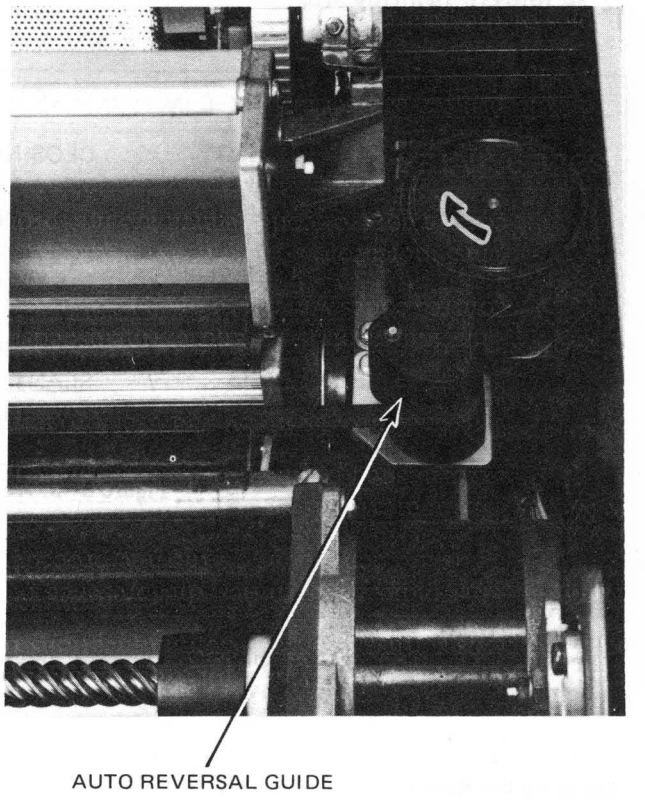
The single color printer ribbon is moved between two spools by two motors with automatic reversal mechanisms. Figure 3–6 shows the ribbon path.

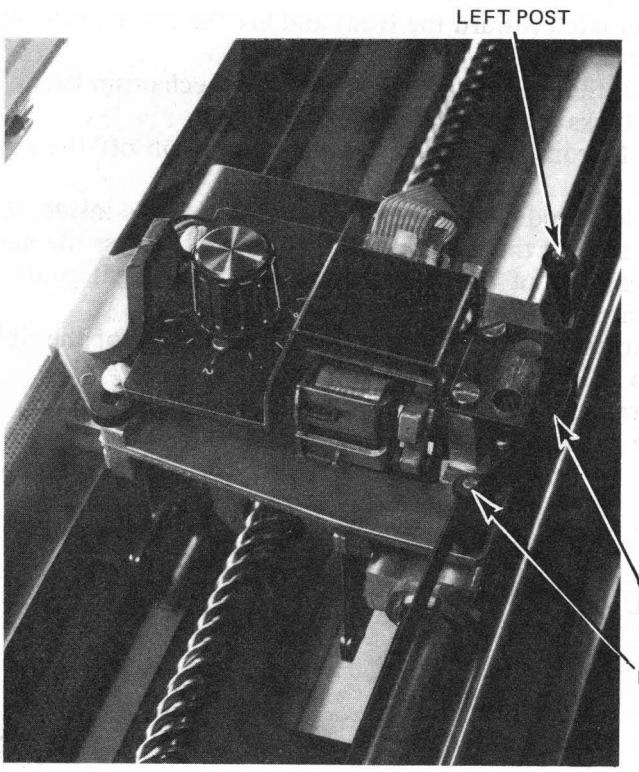
Changing the Ribbon

The following procedure should be followed to change the printer ribbon:

1. Return system to a non-execute state, that is Ready, Standby or Cold (see Section 6).
2. Open printer cover – slide the printer cover latch toward the front and lift the cover to the open position. This places the printer in an idle state.
3. Open the forms handler – push the lever on the front left hand side of the mechanism back. The form handler unit opens and automatically locks in the upper position.
4. Remove the ribbon – lift the ribbon spools from the holder and slip the ribbon off the guides and posts.
5. Place one spool of the new ribbon on the left holder with the ribbon feeding to the inside. Bring the ribbon through the auto reversal guide, around the near side of the left post, across the needle face (front of print head), around the near side of the right post, through the right guide and place spool on right holder. Manually take up slack ribbon.
6. Eyelets are attached to each ribbon 7" from each end to activate reversal at the end of the ribbon. These eyelets must not be between the two automatic reversal guides.
7. Close forms handler – pull forward and forms handler will lock in position.
8. Check that all printer mechanism levers are in the operational position.
9. Close printer cover.







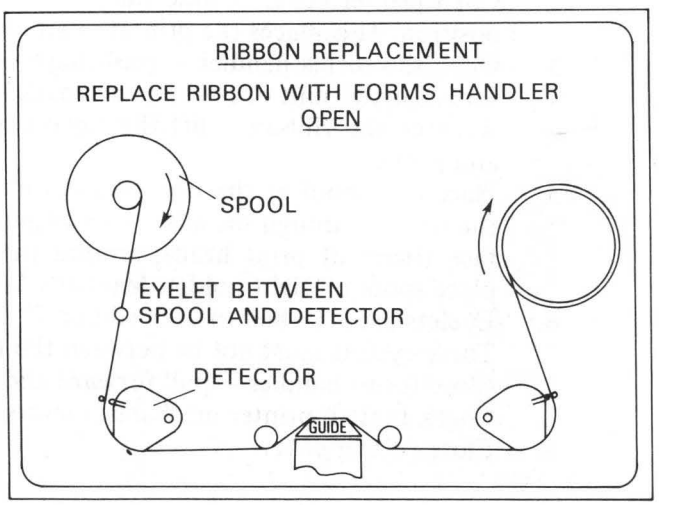


Figure 3–6. Ribbon Path

FORMS HANDLER

The forms handler shown in Figure 3–7 is used to hold and control the advance of the marginally punched pin feed forms. This stationary forms handler can accomodate forms up to 16.75 inches (425 millimeters) wide. The pin feed mechanisms can be moved along the length of the platen by the operator, to allow for different width forms. A scale marked with 1/10 inch (2.54 millimeters) graduations and numeric markings at each inch graduation is provided for use in 'lining up' the form.

VERTICAL SPACING

Forms can be spaced under program control at a single line advance rate of six lines per second and a vertical tabulation rate of 30 lines per second. Forms can also be advanced using the line advance keys. Manual adjustment of forms spacing must be performed with the printer cover open.

Forms may be manually spaced in either direction in 1/12 inch (2.108 mm) increments by turning the twirler at the right end of the Forms Handler. If fine adjustment is required, the lever on the right of the device, behind the twirler, must be pushed back. The twirler then produces spacing in less than 1/12 inch (2.108 mm) increments. The lever must be pulled forward again to return to normal spacing.

VISIBILITY MODE (CLEAR HEAD FEATURE)

If in a keyboard/printer operation the printer receives no print characters in a specified time (0.1 to 9.9 seconds), the paper is automatically moved up 2 lines so that the print head is not obscuring the last print line. The paper is moved down again when a print character is received or any key is depressed. The next position is indicated by an arrow on the print head.

(See AE500 System Software Operation Guide, Basic System Register Utility for setting clear head time).

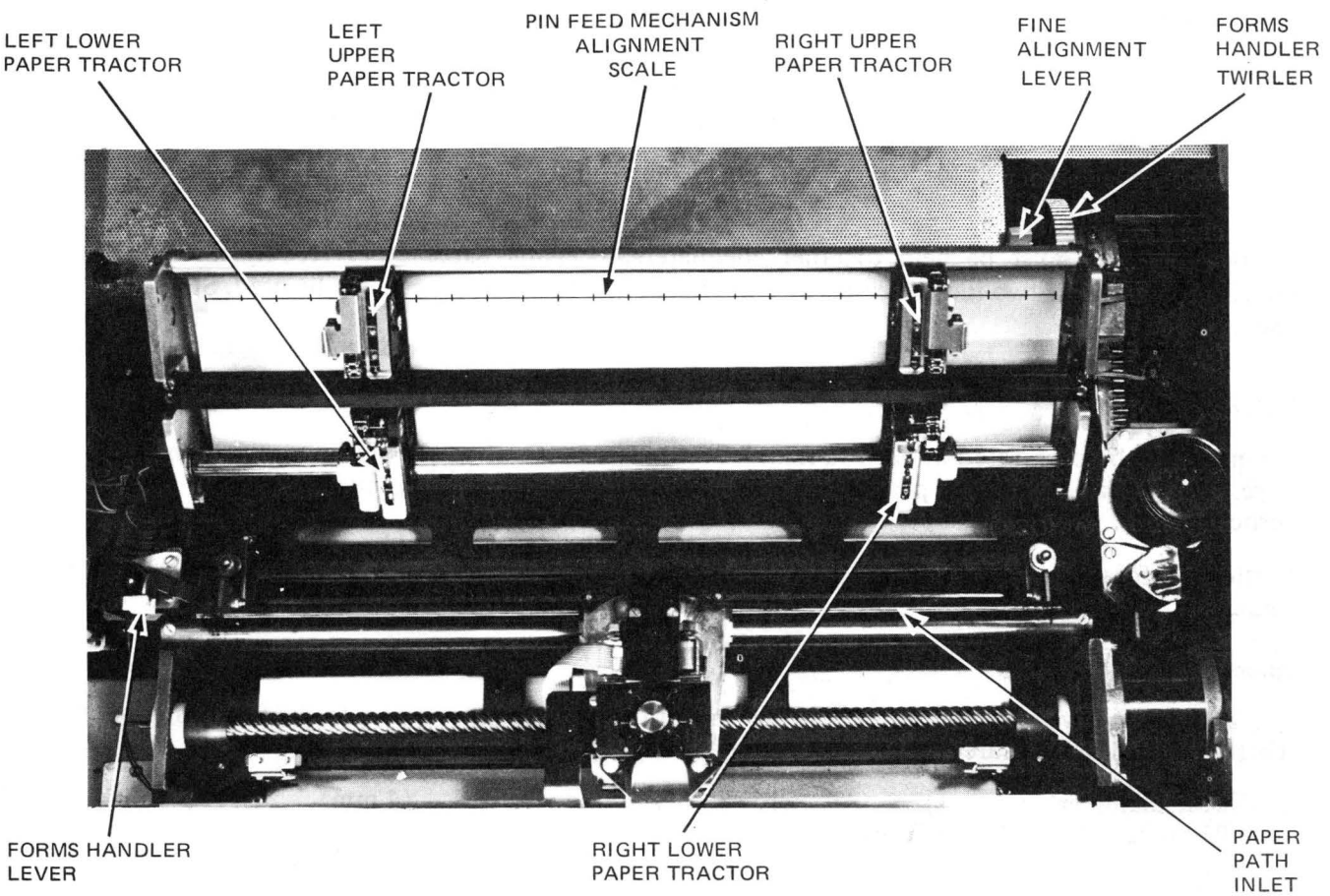


Figure 3–7. Forms Handler

PAPER LOADING

The forms handler cover must first be opened (the machine must be in a non-execute state before opening the printer cover). This disengages a switch which forces the printer into an idle state.

The complete forms handling unit can be opened by moving a lever on the left side of the mechanism. When placed in the upper position, the forms handling unit is automatically locked there. The paper path inlet is accessible externally, and is located in line with the print striking surface, just under the printer.

When the first page of the form is correctly inserted into the paper tractors, the forms handling unit may be closed and latched into the closed position.

FORMS SPECIFICATIONS

The forms specifications for the continuous pin-feed are as follows:

	Minimum	Maximum
Width (including pin feed margins)	3 inches (76 mm)	16.75 inches (425 mm)
Fold	3.5 inches (89 mm)	11 inches (279 mm)
Thickness	.0026 inches (.066 mm)	.020 inches (.508 mm)

Folds less than 3.5 inches or greater than 11 inches may prevent the orderly refolding of the forms after they exit to the rear of the console. If more than one copy of the print is required, multi-part forms can be used. Multi-part forms may be carbonless transfer paper or may be interleaved with carbon paper. Up to 5 copies may be obtained. Carbon paper is recommended to have a thickness of .007 inches (.018 millimetres).

Carbonless transfer paper is recommended to have the following thickness:

1st sheet	.0030 inches with tolerance (.076 mm with tolerance	.0035 inches .091 mm)
2nd to 5th sheet	.0032 inches with tolerance (.081 mm with tolerance	.0038 inches .096 mm)

Multi-part forms must be held together adequately to enable satisfactory feeding through the printer. Glueing of forms at two inch intervals or crimping at half-inch increments can be used. Staples must not be used.

Perforations

Paper should be perforated at each horizontal fold. The perforation should be at a 90° angle to the paper edge. Additional horizontal perforations may be made between folds, but these must not cut into the sprocket holes.

Vertical perforations may be made at a minimum distance of .25 inches (6.35 mm) from the sprocket hole center.

Sprocket Holes

Pin-feed forms have sprocket holes punched in both left and right margins, to the following specifications. These specifications ensure satisfactory feeding of the paper through the printer.

Hole diameter	.157 inches	(4 mm)
Distance between hole centers (nominal)	.5 inches	(12.7 mm)
Edge of paper to center of hole (nominal)	.236 inches	(6 mm)

The cumulative tolerance on the distance between the centers of any two holes within a distance of 10 inches (254 mm) must not exceed .012 inches (0.3 mm). The line through the hole centers in the right hand margin must be parallel to the line through the hole centers in the left hand margin within a tolerance of .005 inches (.125 mm) in a length of 10 inches (254 mm). If carbon paper with sprocket holes is used, the sprocket holes of the carbon paper may have a diameter of .161 inches (4.1 mm). Figure 3–8 shows a pin feed form.

FORMS HANDLER OPTIONS

The following optional features are available:

Media Present Detector. This option detects the end or a breakage of continuous forms. When this occurs, printing is inhibited until the condition is corrected.

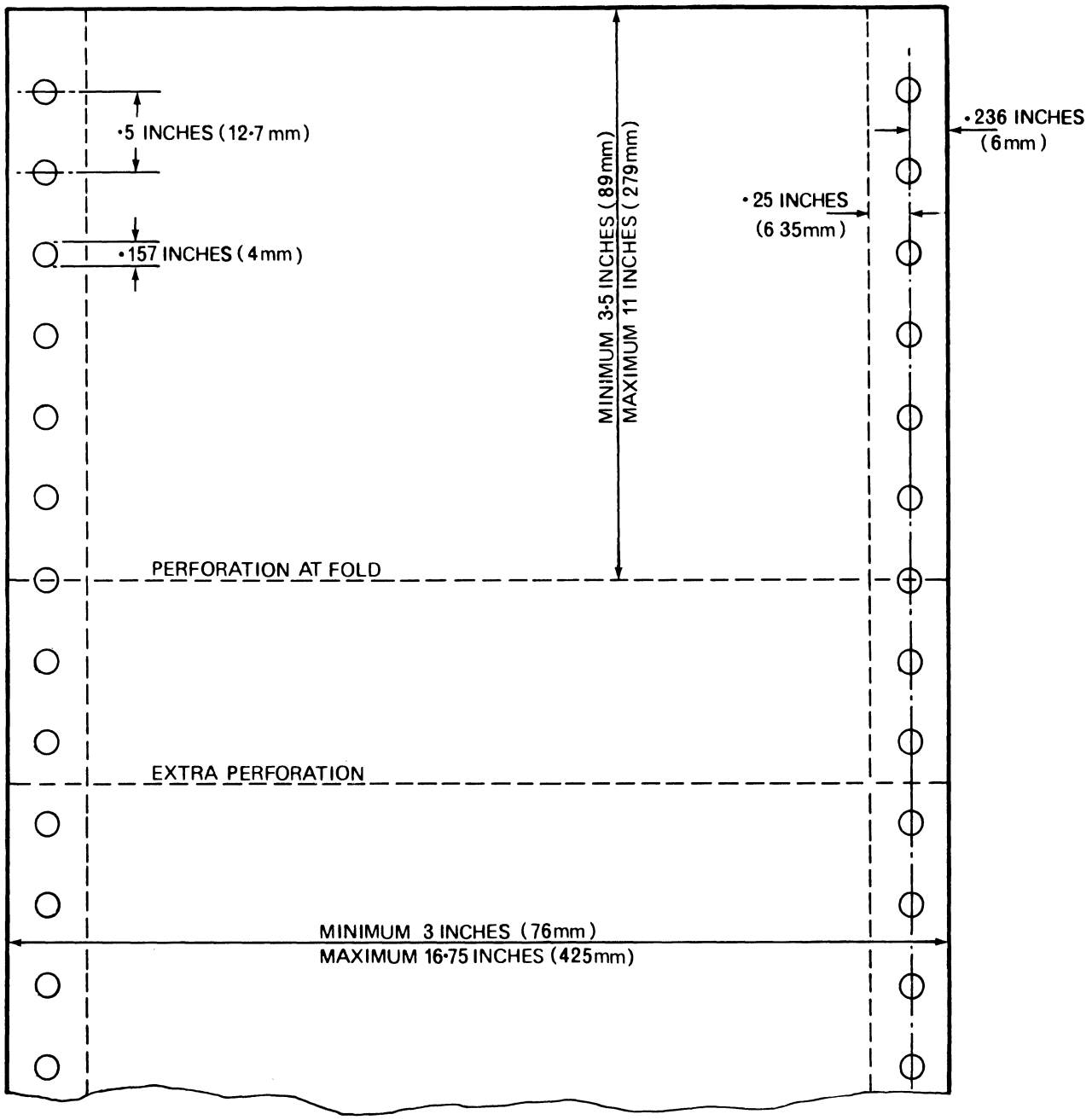


Figure 3–8. Pin Feed Form

OTHER OPERATOR CONTROLS

There are three operator controls located in the processor cabinet, as shown in figure 3–9.

COLD START BUTTON

The cold start button is of the momentary contact type, and performs two functions depending upon the position of the system test switch.

With the system test switch in the normal position (down), pushing the cold start button will initiate a cold start procedure. The cold start procedure is a micro program residing in read only memory, and its interaction with the system operator is described in Section 6 of this manual.

With the system test switch in the transferred position (up), pushing the cold start button will initiate the Customer Confidence Program. The Customer Confidence Program is a micro program residing in an alternative area in read only memory, and is described in Section 9 of this manual.

It must be noted that in either case, the contents of random access memory will be overwritten.

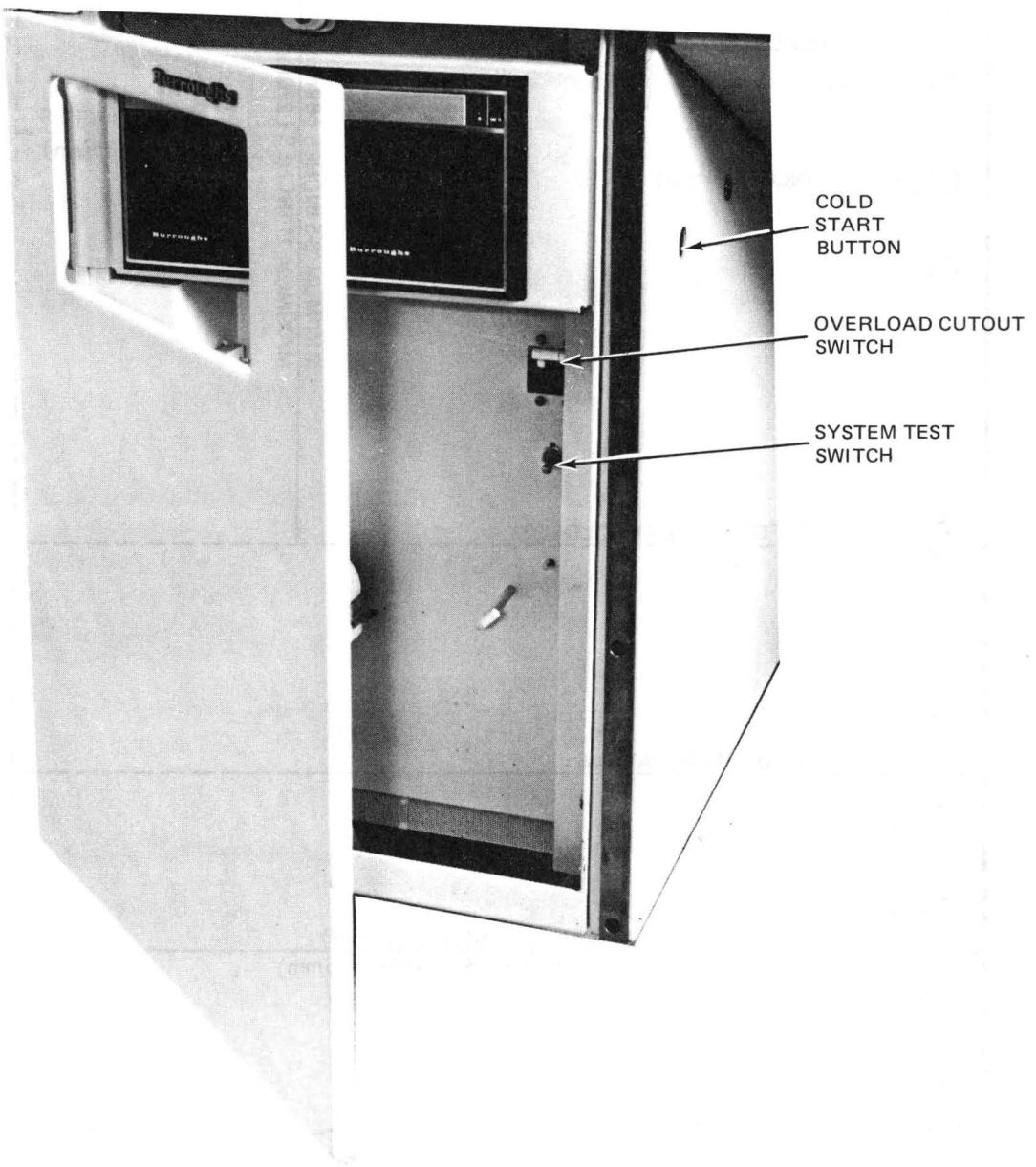


Figure 3–9. Operator Controls

SYSTEM TEST SWITCH

When this two position toggle switch is in the normal position (down), pushing the cold start button will initiate the cold start procedure, which is described in Section 6.

When the system test switch is in the transferred position (up), pushing the cold start button will initiate the customer confidence program, which is described in Section 9.

OVERLOAD CUTOUT

The overload cutout is a safety device intended to protect the operator and the system. If the voltage supplied to the system, or the current drawn by the system becomes too high, the overload cutout will disconnect the supply. The overload cutout reset lever is normally in the uppermost position. If the overload cutout should operate, it may be reset after 10 seconds by moving the reset lever upwards. If it should operate again, either immediately or within a short time, then a field engineer should be called.

SECTION 4
PERIPHERALS

MAGNETIC TAPE CASSETTE

This Input/Output storage subsystem for the AE500 Series is shown in Figure 4–1. The functional and operational descriptions of this device are provided in this section.

CASSETTE

The tape Cassette is a twin hub coplanar type (figure 4–2), and contains a minimum of 282 feet (86M) of computer grade magnetic tape 0.15 inches (3.81mm) wide. The plastic cover of a cassette cartridge protects the tape from dust contamination and provides for simpler operator handling. Data file names and creation dates may be conveniently marked on the cover for information purposes. Data storage capacity of a cassette cartridge, at a recording density of 800 bits per inch, is approximately 800 x 256 byte records.

CASSETTE USE

The cassette should be labelled "A" on one side and "B" on the other side; however, since one of the two tracks is used as a clock track, only one side may be used for recording information.

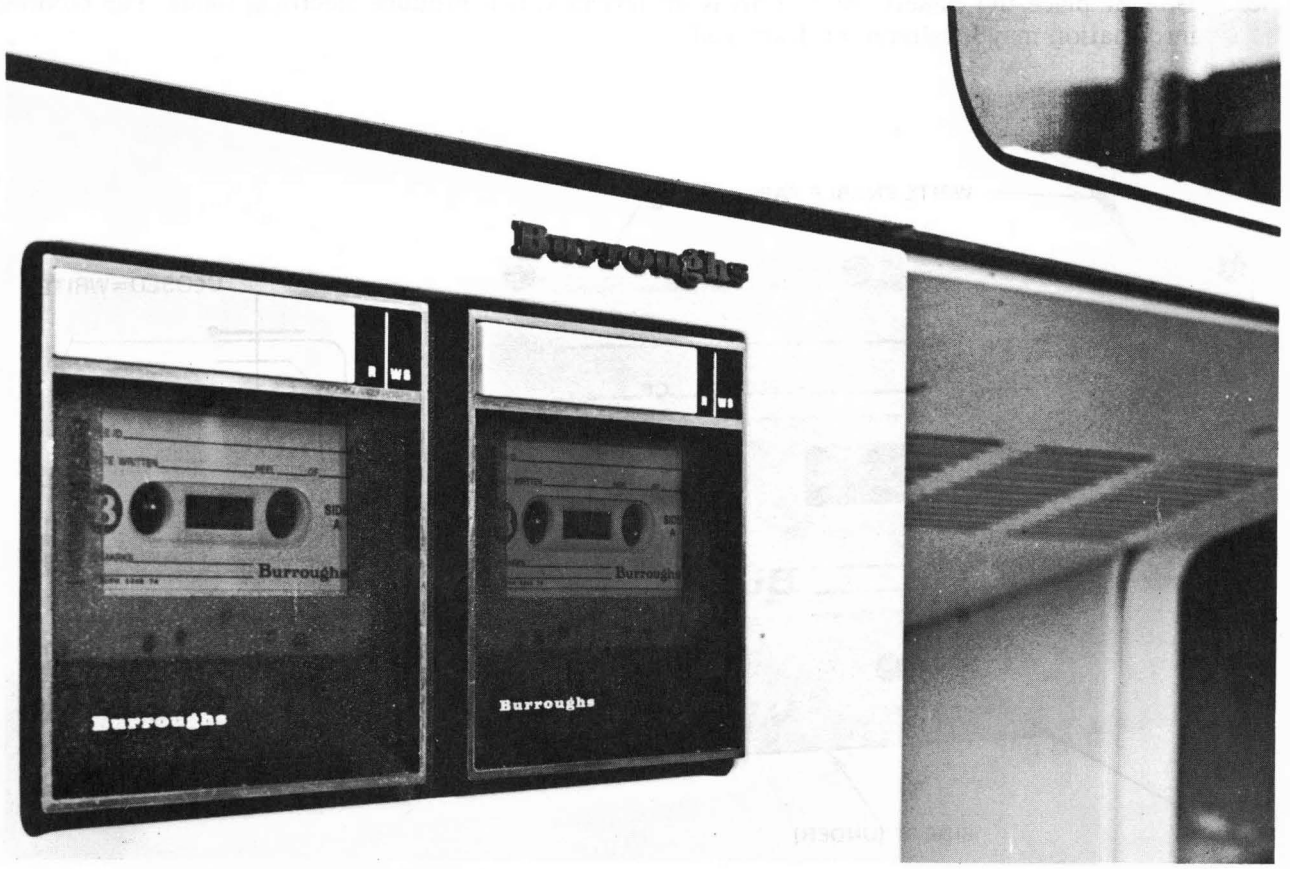


Figure 4–1. Input/Output Storage Subsystem

DATA RECORDING TECHNIQUE

The recording technique features are described below:

a. Dual Track Recording is used.
   One track is used for data and one track is used for clocking to assure valid recording and reading of data.
b. Data is NRZI (Non-Return to Zero Inverted) encoded at 800 bits per inch.
c. Encoding is bit serial (100 characters per inch) in 8 bit code, with the least significant bit written
d. Variable record length with 1 character CRC (cyclic redundancy check) – maximum record length is 256 data characters.
e. Tape mark code to identify file locations.

WRITE ENABLE TABS

Two write enable tabs are fitted to each cassette cartridge as shown in figure 4–2. A WRITE operation can only be performed for a side when the left tab is closed (i.e. covers the write enable hole). Data recorded on a cassette may be protected by opening the write enable tab.

TAPE CASSETTE HANDLING

The following rules should be observed when handling a cassette cartridge, if reliability is to be maintained.

a. Always replace the cassette in its container after use.
b. Rewind the tape before placing in container.
c. Never touch the recording surface or tape with fingers.
d. Never force the cassette into the drive unit or damage may result to both.
e. Do not place the cassette on hot areas or devices which produce electrical fields. The recorded information may be altered or destroyed.

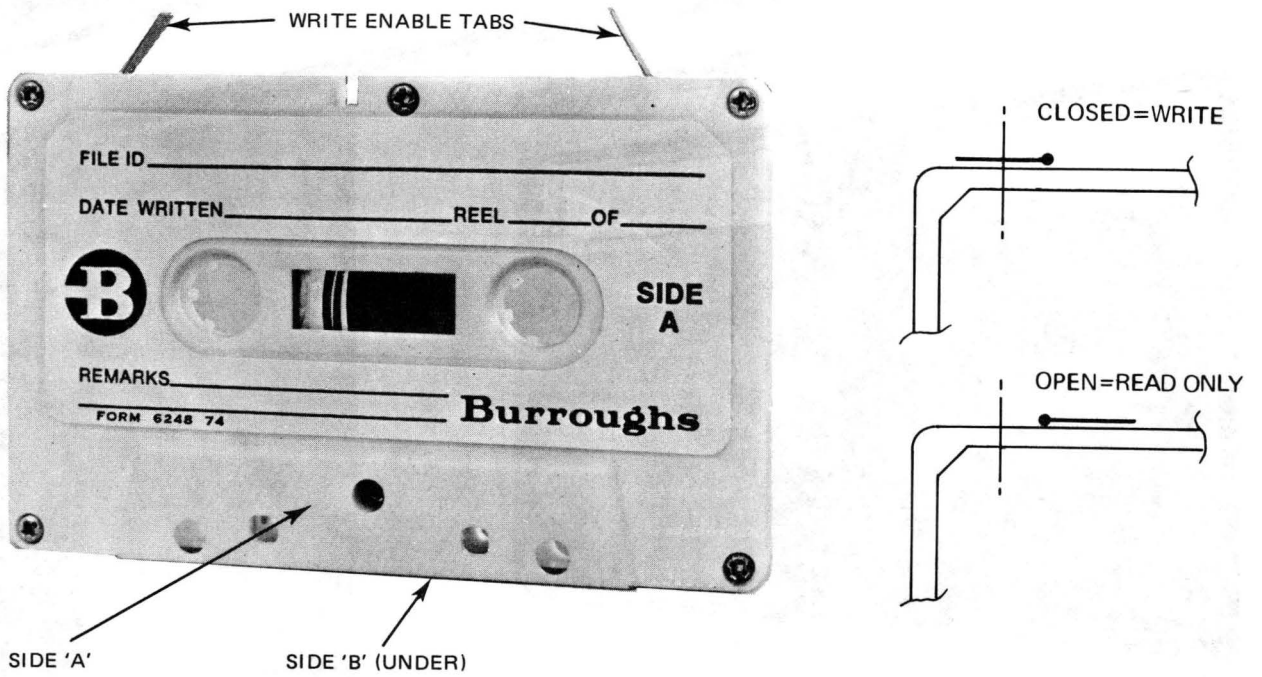


Figure 4–2. Magnetic Tape Cassette

CASSETTE TAPE FORMAT

Figure 4–3 shows a cassette tape with an example of data format. Note that the elements involved in the control and format of a cassette tape are:

CLEAR LEADER
BOT MARK (BEGINNING OF TAPE)
INTER-RECORD GAP
RECORD
TAPE MARK
FILE
EOF (END OF FILE)
EOT MARK (END OF TAPE)

A discussion of each of these follows.

NOTE: The elements controlled by the S-Program may be varied to provide different formats. For example, two tape marks may signal a start of the first file.

CLEAR LEADER

Clear leader precedes and follows the magnetic tape on the cassette. The leader is used by hardware in conjunction with the READY light on the drive unit to inform the operator of tape status. Three lamp conditions are possible: Light OFF indicates the tape is at clear leader and can be removed by the operator. Light ON indicates the tape is off clear leader and capable of receiving commands from the processor. The operator must not remove the cassette when the ready light is ON. Light BLINKING ON AND OFF indicates that operator intervention is required to correct a cassette problem. The conditions that create a blinking light are as follows:

The lamp will blink when the cassette is hung up on clear leader at the end of tape. Under this condition no S-Program commands will move the tape. If a READ or WRITE command is given and if a function complete or beginning of tape (BOT) is not detected after three seconds, this indicates that the cassette is in backwards, a tape jam has occurred, or an attempt is being made to read a blank tape. This blinking light status is flagged to the processor. The blinking light can only be stopped by opening the cassette drive door as in the case of the rewind-write inhibit function.

BOT MARK (BEGINNING OF TAPE)

Writing of the cassette tape commences from the BOT marker which is a punched hole in the magnetic tape. Before writing starts an initial erased gap from the BOT marker is provided by hardware.

INTER-RECORD GAP

The gaps between written areas of the tape are called Inter-Record Gaps. These are a result of the required time for the cassette drive to attain write speed before writing and to stop after writing is completed. Hence, this is a result of hardware control.

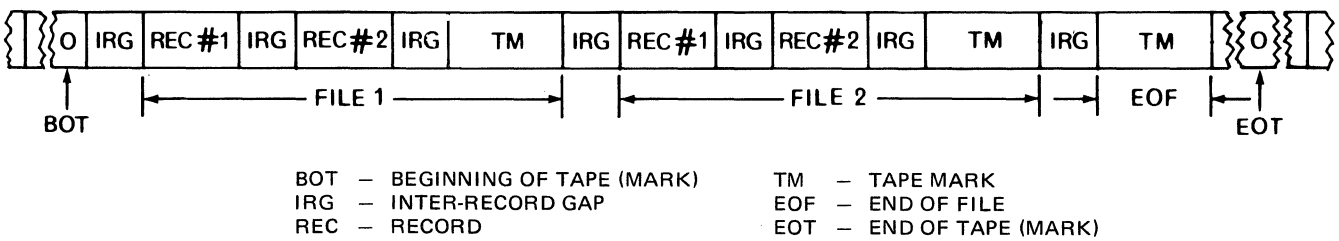


Figure 4–3. Cassette Tape Example Format

RECORDS

All recorded information is written in RECORDS. Two types of records are used, CONTROL and DATA records.

Control records (TAPE MARKS) are created automatically when an S-program issues a WRITE TAPE MARK command. The proper record format is generated automatically.

The DATA RECORD requires S-program control in setting up buffers with data. Each record contains control characters as well as data characters. Figure 4–4 and 4–5 show the elements of data records and control records (Tape Mark) respectively. The elements are:

Preamble/Postamble Character
Data Characters
CRC Character
NUL Character

Each character is located in a byte of eight bit positions along the tape. There is no character parity bit. They are discussed below.

Preamble/Postamble Character

The PREAMBLE/POSTAMBLE CONTROL CHARACTER is an 8 bit "1010 1010" character. It is formatted by the processor and appears at the beginning/end of a series of characters loaded by the processor when writing on the tape. The preamble/postamble is written on the tape.

Data Characters

Encoding of data characters is in 8 bit code. Normally, the data code used for compatible information interchange is the ASCII 7 bit code. This is recorded in the 7 least significant bit positions of the 8 bit character. The 8th bit position always contains a zero when using the ASCII code.

Providing that the character size on the tape is 8 bits, any code set can be used for the encoding of data. This is accomplished by the use of translation tables.

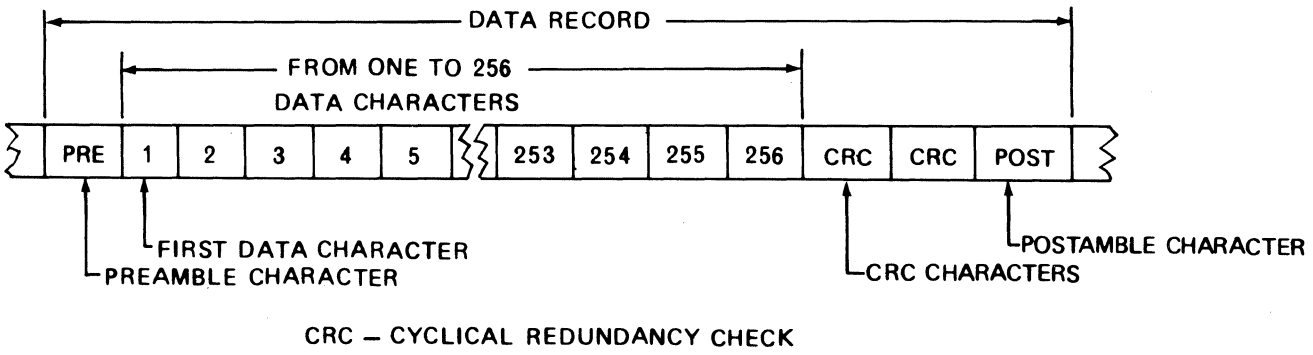


Figure 4–4. Cassette Data Record Format

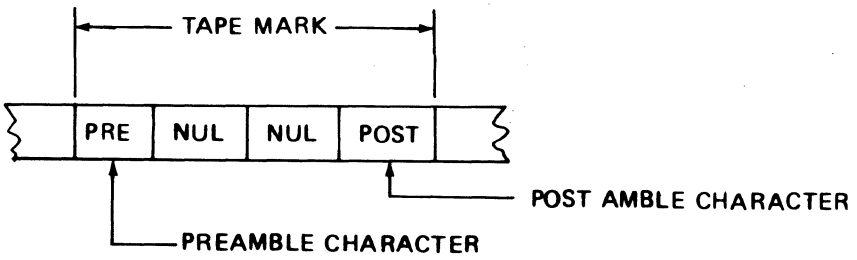


Figure 4–5. Cassette Tape Mark Format (Control Record)

CRC-Cyclical Redundancy Check Characters

The CRC CONTROL CHARACTER is generated in hardware during a write operation and is used as an error check in hardware during a read operation. It contains 16 bits and is a function of polynomial. It is developed from the serial bit data stream of the record being written in a least-significant-bit-first sequence. The CRC generation does not include the preamble and postamble characters.

NUL Character

All eight bits of the NUL character are zero. This data character is generated automatically when a Tape Mark is written on the tape.

TAPE MARK

The Tape Mark is actually a special record which is used in a control capacity. Generally, a tape mark signals to the S-program an end of a data file. Two tape marks may signal to the S-program the end of recorded data, or the end of a complete file. Special S-instructions permit writing, forward to tape mark, and reverse to tape mark operations.

FILE

Creating a file format is a SYSTEM SOFTWARE function. Generally a file consists of a number of records separated by inter-record gaps and ending in a tape mark.

EOT (END OF TAPE)

The EOT marker is a punched hole near the end of the magnetic tape. It provides a signal to the processor hardware that the end of magnetic tape is being approached. A maximum of two (2) additional records may be written after the EOT mark is detected.

CASSETTE DRIVE UNIT

There are one or two cassette drive units on AE500 System. Each unit utilizes a computer grade magnetic tape cassette cartridge. Cassette data is written bit serially and is encoded by the drive unit hardware using the NRZI (Non-Return to Zero Inverted) method.

While writing, two tracks are written on the tape simultaneously, one track for data, the other for clock signals. The clock signals recorded on the tape are used for synchronizing the READING back of the data from the DATA TRACK during some future READ operation.

The cassette drive unit utilizes a DUAL GAP HEAD (one part for Write and one for Read) for reading and writing on the cassette tape. This allows data to be checked immediately after it is written on the tape instead of making two forward passes of the tape for each record written (Write, Backspace and Read).

Cassette power is supplied by the AE500 processor. The unit is powered on or off with the processor. A rewind operation is initiated automatically when a cassette is inserted in the drive or when power is turned on.

OPERATING FUNCTIONS AND SPEEDS

The various functions and speeds are listed below:

a. WRITE – 10 inches per second.
b. WRITE TAPE MARK – 10 inches per second.
c. ERASE – 10 inches per second.
d. REWIND – 60 inches per second.
e. REWIND AND INHIBIT WRITE – 60 inches per second.
f. BACKSPACE – 10 inches per second.
g. READ – 10 inches per second.
h. SEARCH – 10 inches per second.
i. FORWARD TO TAPE MARK – 30 inches per second.
j. REVERSE TO TAPE MARK – 30 inches per second.

CONTROLS

Drive Unit Door

A horizontal bar at the upper front of the drive opens the door when depressed for the insertion or removal of a cassette. Opening the door creates a NOT READY condition. When the cassette is inserted in the drive and the door is closed, an automatic rewind to the clear leader will result.

NOTE: The OPERATOR must NOT open the drive unit door unless the READY LIGHT (see below) is OFF or flashing.

Opening the cassette drive unit door during the processing operation will result in a NOT READY condition and subsequent closing of the door will result in an automatic rewind. Information may be destroyed if the door is open when the READY light is ON.

Drive Unit Power

Power is supplied and controlled by the processor.

INDICATORS

Ready Light

The Ready Light is labelled "R" as shown in figure 4–6. The READY LIGHT will be ON when the tape is not at clear leader. When the ready light is ON, the operator should not open the cassette door since the tape would automatically rewind to the beginning of tape upon reinsertion and the prior position on the tape would be lost. The operator should remove the cassette only when the ready light is OFF.

The READY LIGHT will be OFF when the tape is at the beginning of the tape (CLEAR LEADER). The drive unit door may be opened and a cassette cartridge may be inserted or removed when the READY LIGHT is OFF

The READY LIGHT will blink ON/OFF if the tape forward movement reaches the CLEAR LEADER at the end of the tape. Under this condition OPERATOR intervention is required. To alleviate this condition, the operator must remove the cassette and rewind the cassette about six inches to get off the clear leader. This can be accomplished by turning the cassette hubs with a pencil or finger. Reinserting the cassette into the recorder will then rewind the cassette to the clear leader at the beginning of tape.

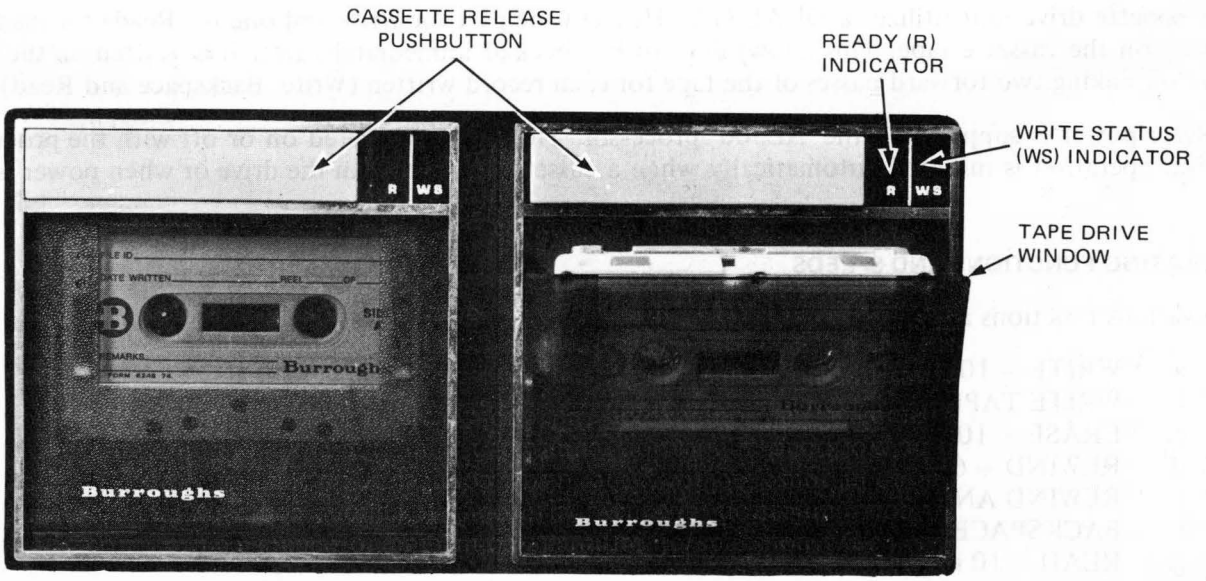


Figure 4–6. Cassette Drive Unit

Write Enable Light

This lamp when ON indicates that the cassette is properly in place and that writing is enabled.

Exception Condition Indicators

These are used to inform the operator of the device status. The NOT READY condition results when the drive unit door is open. READ or WRITE errors may be due to tape in BACKWARDS, TAPE JAM, or an attempt to read a blank tape. For details of these indicators, refer to Appendix C.

HARDWARE FUNCTIONS

The cassette drive unit control hardware, the processor interface hardware, cassette interpreter segment, and S-PROGRAM defines how the cassette will function. Automatic control functions are associated with the hardware and interpreter. S-PROGRAM controls are responsible for the PROGRAMMATIC FUNCTIONS.

Automatic Functions include control of the cassette device once a programmatic control function has been initiated. Also, automatic stop and rewind to clear leader are included. The hardware/interpreter automatic control functions are discussed under PROGRAMMATIC FUNCTIONS where applicable.

Automatic Stop

When the tape is moving in reverse, it is automatically stopped if it reaches the clear leader.

If the tape is moving forward and detects clear leader at the end of tape, the cassette drive will stop and the recorder will be "hung up". The cassette controller logic will ignore all motion commands if this situation occurs. To remove this condition, operator intervention will be required.

To alert the operator that intervention is required the ready light will blink off and on.

Automatic Rewind to Clear Leader

When power is applied to a drive unit containing a cassette or when a cassette is inserted in a drive unit, the cassette will automatically rewind to the clear leader at the beginning of the tape and stop.

Cassette Function Overlap

The cassette subsystem is NOT a buffered device. However, processing of a new instruction will commence as soon as the BASIC cassette operation is completed, during the time that hardware is stopping the drive unit tape movement (FUNCTION COMPLETE). If the new instruction is a cassette instruction, the processing will wait until the current cassette stopping phase is completed (approximately 75 ms) before the new instruction will continue executing. This is true regardless of whether or not the new cassette instruction is for the cassette unit currently going through the stopping phase.

Efficient programs will utilize the stopping time to perform non-cassette functions. For example a fairly long subroutine may be entirely executed during this time frame.

Only the REWIND function is an exception to the above. The processing of the next instruction will commence as soon as the current REWIND begins. Hence, both cassette stations may REWIND at the same time. It is also possible to READ or WRITE (or any other non-rewind function) on one station while rewinding on the other station provided that the REWIND command is given prior to the READ or WRITE (etc.) command.

WRITE CONSIDERATIONS

Records should be written sequentially even if records are of fixed length. Files should be corrected and updated by generating a new tape from the one which requires an update.

PROGRAMMATIC FUNCTIONS

These functions are described below.

Exception conditions for a given function are flagged by the Processor. The S-program is responsible for checking error conditions after the execution of every cassette function. Operator action in the event of an error condition will depend on the particular applicational requirements (see Appendix F for E and L flags).

Write

The WRITE function is used to create data records. Data is written to the cassette from specified areas of memory (buffers), as designated by the SYSTEM SOFTWARE/USER PROGRAMMER.

Hardware and microprogram responsibility includes generating the PREAMBLE/POSTAMBLE CRC character.

During write the written data is read back after being written. The written data will be checked in hardware by recognizing the preamble and then calculating the CRC (cyclical redundancy check). This calculated CRC will effectively be compared with the generated CRC written on tape. The presence of the proper postamble will also be checked by the hardware. An error in the preamble, postamble or CRC check will be flagged to the processor.

The microprogram will generate the CRC during the write data mode of operation. The preamble or postamble is not used in the generation of the CRC.

Write Tape Mark

The WRITE TAPE MARK function is used to create CONTROL RECORDS (Tape Marks).

The hardware and microprogram responsibility is essentially the same as during the WRITE function. The TAPE MARK has no CRC (CYCLICAL REDUNDANCY CHECK). Only the PREAMBLE and POSTAMBLE are generated and checked by the hardware/microprogram.

Read

The READ function retrieves a single RECORD from the tape. The record may be either a DATA record or a CONTROL record (Tape Mark). Reading a CONTROL record will result in a SYSTEM SOFTWARE flag being set. A DATA record will be read into a designated memory location.

The hardware/microprogram during the read mode will determine if the transport is reading a bona fide record. This essentially requires the digital filtering of any inter-record gap noise due to electrical noise or bad tape. The hardware will also check for the presence of a preamble. If no preamble is detected in a record, the message will be altered to the extent that the CRC will be incorrect. This will result in setting the READ ERROR FLAG. In the case of a good record, hardware will strip away a detected preamble. The data will be sent to the processor. Hardware will detect the end of record. The micro-program will send the DATA to the READ buffer.

Search

The cassette is searched for a field that equals a field defined by the SYSTEM SOFTWARE.

The data field on tape, to satisfy the equal condition, must start in the most significant digit of the tape record.

The search ends when one of the following conditions is located:

a. The equal condition is located.
b. A tape mark is located.
c. A clear leader condition occurs.

If the equal condition is located, the tape data record will be read in a normal READ process.

Forward to Tape Mark

A forward tape movement under hardware/microprogram control to a Tape Mark will result when the SYSTEM SOFTWARE instruction is given.

Reverse to Tape Mark

A reverse tape movement under hardware/microprogram control to a Tape mark will result when the SYSTEM SOFTWARE instruction is given.

Backspace

When the SYSTEM SOFTWARE requests the BACKSPACE function the tape on the cassette drive specified will backspace (Reverse) until an inter-record gap is reached.

Rewind

A rewind of cassette to clear leader will result when the SYSTEM SOFTWARE rewind instruction is given. The hardware and microprogram then control the rewind.

Once a rewind has been initiated there is no way for the SYSTEM SOFTWARE to stop the tape. The internal tape cassette logic holds the drive in this state until clear leader is reached.

It is possible to have one cassette drive in the rewind state and the other cassette drive in the read or write mode.

Rewind and Inhibit Write

This function is the same as REWIND with the added feature that the WRITE functions are inhibited when the cassette reaches clear leader.

Upon reaching the clear leader, the drive will go ready but the hardware will inhibit any write commands. Any write commands that may be received will not cause motion but will indicate a function complete.

The only way the write inhibit can be removed is by opening the cassette drive door that holds the cassette in place.

Erase

The ERASE function provides a forward erase of approximately 4.5 inches of tape.

MAINTENANCE

The tape path and Read-Write Head should be cleaned, once a day, with a cotton swab and isopropyl alcohol. Care should be taken not to leave any cotton in the tape path after cleaning.

SECTION 5
DATA COMMUNICATIONS SUBSYSTEM

INTRODUCTION

The purpose of this section is to describe the hardware characteristics of the data communications subsystem. The functions and capabilities of individual components, and the hardware requirements for particular communication links and network configurations are described. System software is provided to control the hardware and provide a simple user interface with data communication lines. The available system software is described in Sections 6 and 8 of this manual.

DATA COMMUNICATION NETWORKS

A wide variety of network configurations is possible, ranging from two systems connected by a single communication link, to many thousands of terminals connected to a data centre through a variety of communication links. A communication link may be through telephone company facilities using switched or leased lines, or may be an in-house facility, either direct connect or through data sets. The line procedure used will depend upon network and application requirements, and the mode of communication (asynchronous or synchronous, simplex, half duplex, or full duplex) upon the volume of traffic and quality of the communication link.

COMMUNICATION MODES

A communication link may be used to:

a) Pass information in one direction only (Simplex),
b) Pass information in both directions, but not simultaneously (Half Duplex),
c) Pass information in both directions simultaneously (Full Duplex).

Information may be passed asynchronously or synchronously. In asynchronous transmissions, small "parcels" of bits (usually characters) are enclosed within start and stop bits. The time interval between the start and stop bits is precise, but the interval between start bits in different "parcels" is arbitrary. In synchronous transmissions, no start and stop bits are transmitted. The transmitter and receiver must be in perfect synchronisation throughout the entire transmission of a message. Data sets designed for synchronous transmission usually provide accurate timing signals to maintain bit synchronisation. Synchronous line procedures transmit and recognise synchronisation characters at appropriate points in the procedures, to establish and verify character synchronisation.

PHYSICAL LINKING

The communication link between data communication systems may be voice grade lines (requiring data sets), or may be a direct connection. Systems may also be concatenated, permitting one communication link to service groups of terminals.

Voice Grade Lines

Voice grade lines were not designed for the transmission of digital information. They are only used because the public telephone installations provide a vast network of such lines. In order to use voice grade lines for digital information transfers, the information must be translated into signals which are similar to vocal transmissions. This function is provided by the data set. Data sets provide an audio frequency tone which is "modulated" (altered in amplitude, frequency, and/or phase) to "carry" the digital signal. This audio tone is known as the carrier wave, and data sets are frequently known as MODEMS since they modulate and demodulate the carrier wave.

The interface between data sets and data communication equipment is defined by E.I.A. Standard RS-232-C. This interface provides a comprehensive group of interchange circuits, not all of which are used by a particular data set/terminal combination. The subset of the standard interface which is used by AE500 systems is described later in this section. The AE500 data communication interface also includes interchange circuits which provide compatibility with CCITT V.24 and V.26 data sets.

Direct Connect

Data communication systems may be connected without data sets when the distance between them is limited (for instance, an "in-house" facility). Two methods of direct connection are provided for use with AE500 systems, one permitting connection at up to 1000 feet (304 metres), and the other permitting connection at up to 15000 feet (4572 metres). These facilities are described later in this section.

AE500 DATA COMMUNICATIONS HARDWARE

Two data communication hardware controllers are available for use with AE500 systems. The Asynchronous Data Communication Controller (ADC) provides the ability to communicate in the asynchronous mode at bit rates of up to 1800 bits per second through data sets, or up to 9600 bits per second using direct connect. The Synchronous Data Communication Controller (SDC) provides the ability to communicate in the synchronous mode at bit rates up to 4800 bits per second half duplex, or 2400 bits per second full duplex. Both controllers provide full duplex capability.

Each controller is used in conjunction with an external interface printed circuit board (PCB). Three types of interface PCB's are available to provide the following capabilities:

a) CWM – Concatenate/Wraparound/Modem interface board. This PCB permits the AE500 to communicate with other systems using RS-232-C or CCITT V.24 and V.26 compatible data sets. The CWM may be used both with the Asynchronous and with the Synchronous Data Communication Controller.

b) TDI – Two-wired Direct-connect Interface. This PCB permits the AE500 to be directly connected to other systems at distances up to 1000 feet. TDI only applies to the Asynchronous Data Communication Controller.

c) BDI – Burroughs Direct-connect Interface. This PCB permits the AE500 to be directly connected to other systems at distances up to 15000 feet. BDI only applies to the Asynchronous Data Communication Controller.

All of the above interface PCB's provide "local wraparound", enabling the Customer Confidence Program to test all the functions of the data communication controller in a "live" environment.
NOTE: TDI and BDI terminals cannot be mixed.

ASYNCHRONOUS DATA COMMUNICATIONS CONTROLLER

The Asynchronous Data Communications Controller (ADC) permits the AE500 to be installed in a wide variety of asynchronous data communications environments. Character format and operating speed may be adjusted to suit many applications, as described later in this section. The ADC can be used with the printed circuit boards mentioned above to permit operation in direct connect, common carrier, and concatenated environments.

The Asynchronous Data Communication Controller (ADC) is effectively a complex buffer between the processor and the external interface board. Character serial data from the processor is transferred in bit serial form to the external interface board, and is transmitted via the communication link. Similarly, bit serial data received from a remote location is transferred in character serial form to the processor. Parity generation/checking and character format are automatically handled by a combination of hardware and System Software, as described later in this section.

EXTERNAL INTERFACE

Data set command signals are produced by the ADC under line procedure control, and data set status signals can be passed to the line procedure through the ADC status byte as shown in figure 5–1. The Data Set Interface is compatible with RS-232-C and CCITT V.24 data sets, and consists of all the circuits shown in Table 5–4 except those marked "S" (Synchronous only) in the left-most column. The Data Set Interface signals from the ADC are converted to the required external voltage levels by the appropriate interface board. The interface boards also permit concatenated operation, and are described later in this section. The ADC provides signals for control of the concatenation and wraparound functions of the interface board, as shown in figure 5–1.

INTERNAL INTERFACE

The interface between the ADC and the processor is the I/O Common Interface described in Section 2 of this manual. Figure 5–1 shows the internal interface.

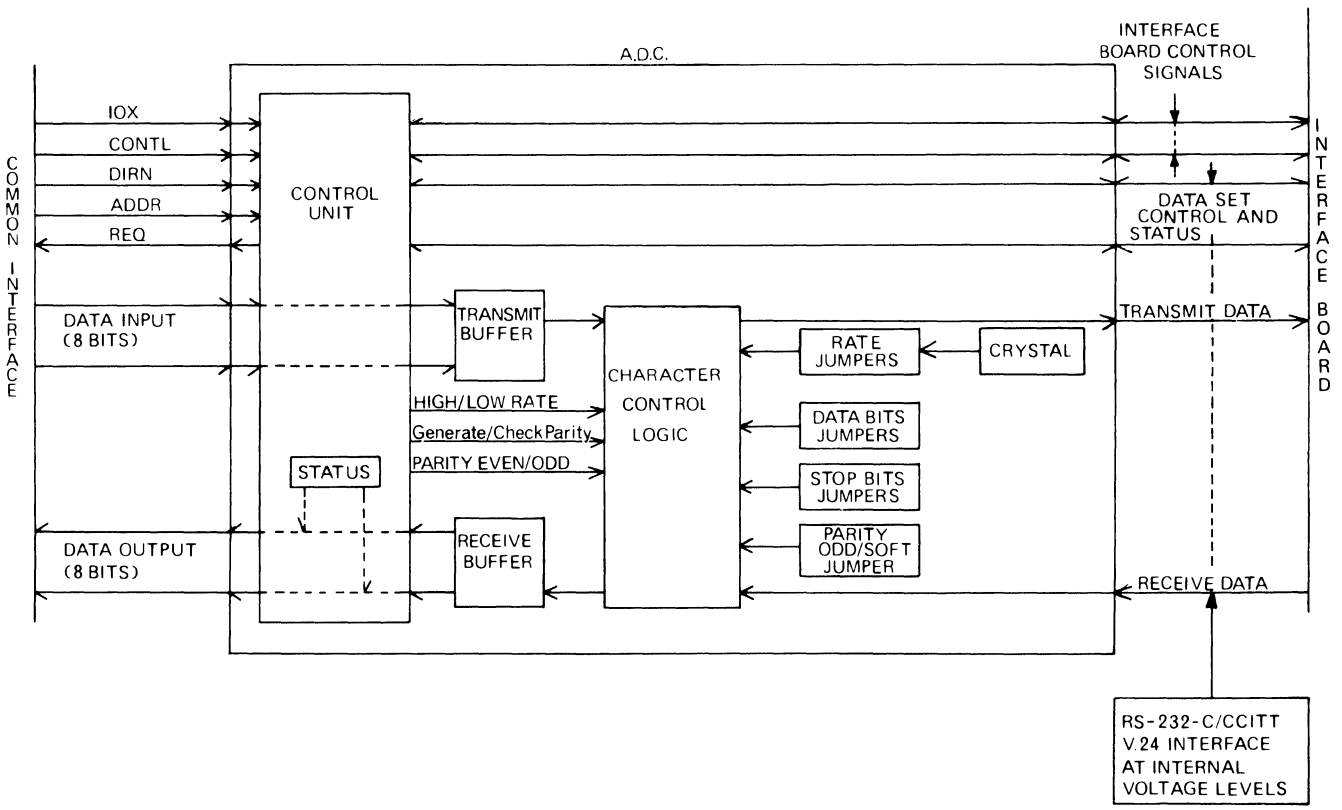


Figure 5–1. Asynchronous Data Communications Controller Block Diagram

ADC VARIABLE CHARACTERISTICS

The Asynchronous Data Communications Controller (ADC) may be operated in a variety of network configurations. Operating characteristics may be altered by a combination of hardware adjustment and system software control, as described below. The operating characteristics which may be altered are shown in Table 5–1.

Table 5–1. Asynchronous Controller Variable Characteristics

VARIABLE CHARACTERISTIC	HARDWARE SELECTION*	SYSTEM SOFTWARE SELECTION**
Number of Data Bits	5, 6, 7, or 8	See Table 5–2 "TRANSPARENCY MODE"
Number of Stop Bits	1 or 2	No Control
Generate/Check Parity	No Control	Yes/No
Parity Even/Odd	Odd Parity or Software Selection. Refer to Table 5–2	Odd or Even if the Hardware is set for Software Selection. Refer to Table 5–2.
Transmit/Receive Rate	A high and a low rate fixed by Oscillator Crystal replacement and Jumper installation as shown in Table 5–3	The Higher or Lower Rate of the two rates fixed by Hardware Setting

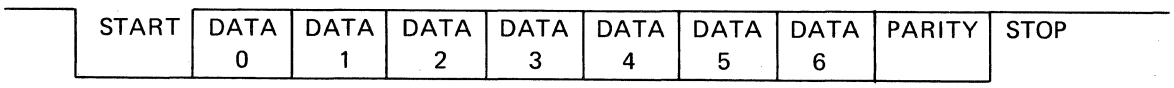
* These selections may be altered by Field Engineering

** Refer to Sections 6 and 9 of this manual for details of software control.

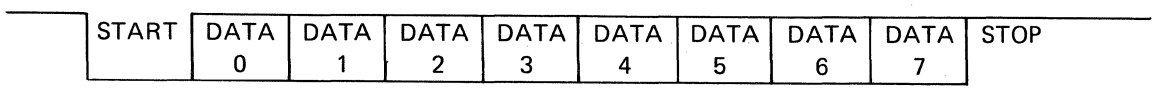
Number of Data Bits

The ADC is adjustable by wire jumper selection to enable transmission and reception of 5, 6, 7, or 8 data bits. The number of data bits does not include parity. The ADC transfers, least significant bit first, the specified number of bits to/from the least significant end of the 8 bit I/O data highway. System Software parity control permits the replacement of the parity bit by the next more significant data bit from the 8 bit I/O data highway. This capability cannot be used if the ADC is adjusted for 8 data bits, since no 9th bit is available.

Example 1–7 data bits, 1 stop bit, parity required.



Example 2–7 data bits, 1 stop bit, no parity.



Section 8 of this manual gives details of the "data bit" requirements of each available line procedure.

Number of Stop Bits

Wire jumper selection is provided to adjust the number of stop bits which are used to terminate a character. One or two stop bits can be selected. Section 8 of this manual gives details of the "stop bit" requirements of each available line procedure.

Generate/Check Parity

The ADC may be used either in parity mode, or in transparent mode. System software will condition the ADC for the appropriate mode in response to user requirements. When the ADC is conditioned for parity mode, the polarity of the parity is determined by a further selection, as detailed later. When the ADC is conditioned for transparent mode, the parity bit is replaced by the next more significant bit available on the data highway. Table 5–2 shows the parity mode resulting from the various selections. Refer to Section 8 of this manual for the parity requirements of available line procedures.

Parity Even/Odd

A wire jumper selection on the ADC board is provided for parity control. The ADC may be adjusted for fixed odd parity, or for system software selection. System Software may select odd or even parity in response to user requirements. Table 5–2 shows the combined effect of the parity control selections. Section 8 of this manual details the parity requirements of available line procedures.

Transmit/Receive Rate

The ADC may be conditioned to operate at one of two transmit/receive rates. The pair of rates is selected by hardware jumper and oscillator crystal selection as detailed in Table 5–3. System software determines which of the rates, "HIGH" or "LOW", is to be used in response to user requirements. This capability permits communication to be continued at the lower rate when an unacceptable error rate prevents communication at the higher rate.

Table 5–2. Asynchronous Data Communication Parity Control

ADC JUMPER SETTING	SYSTEM SOFTWARE SPECIFIES PARITY REQUIRED? **	SYSTEM SOFTWARE PARITY MODE **	RESULTANT PARITY
ODD	YES	EVEN	ODD
ODD	NO	EVEN	TRANSPARENT*
ODD	YES	ODD	ODD
ODD	NO	ODD	TRANSPARENT*
SOFT SELECT	YES	EVEN	EVEN
SOFT SELECT	NO	EVEN	TRANSPARENT*
SOFT SELECT	YES	ODD	ODD
SOFT SELECT	NO	ODD	TRANSPARENT*

* TRANSPARENT MODE –

If the ADC is set for 5, 6, or 7 data bits, the parity bit will be replaced by the 6th, 7th or 8th data bit.

This selection should not be made if the ADC is set for 8 data bits, since no 9th data bit is available.

** Refer to Sections 6 and 9 of this manual for details of software control.

Table 5–3. Transmit/Receive Rates available using the Asynchronous Data Communications Controller

Crystal Frequency	153.6 KHz		57.6 KHz	
Software Rate Select*	HIGH	LOW	HIGH	LOW
Available Rates Selected by Wire Jumpers	9600	4800	3600	1800
	4800	3200	1800	1200
	3200	2400	1200	900
	2400	–	900	–
	4800	2400	1800	900
	2400	1600	900	600
	1600	1200	600	450
	1200	–	450	–
	2400	1200	900	450
	1200	800	450	300
	800	600	300	225
	600	–	225	–
	1200	600	450	225
	600	400	225	150
	400	300	150	112
	300	–	112	–

* Refer to Sections 6 and 9 of this manual for details of software control.

SYNCHRONOUS DATA COMMUNICATIONS CONTROLLER

The Synchronous Data Communications Controller (SDC) permits the AE500 to be installed in a wide variety of synchronous data communications environments. Parameters supplied by System Software provide for automatic handling by the SDC of character format and line procedure functions. The SDC is used in conjunction with the Concatenate/Wraparound/Modem Interface (CWM) to permit operation in a common carrier environment.

The SDC consists of a micro programmed supervisory processor, controlling a Device Control printed circuit board (PCB) which provides the basic synchronous functions. The micro program for the supervisory processor resides in read only memory (ROM) provided in the SDC. Random access memory (RAM) is provided in the SDC and is used for data buffering, command and parameter transfers from the AE500 processor, and as working storage. The SDC micro program drives the Device Control PCB through interpretation of command and parameter information inserted in SDC RAM by System Software. Figure 5–2 shows the Synchronous Data Communications Controller.

The SDC is effectively a complex buffer between the processor and the external interface board. Character serial data from the processor is transferred in bit serial form to the external interface board, and is transmitted via the communication link. Similarly, bit serial data received from a remote location is transferred in character serial form to the processor. The SDC permits half duplex operation at rates up to 4800 bits per second, or full duplex operation at rates up to 2400 bits per second. Vertical parity generation and checking, and block check (either LRC or CRC) generation and checking are handled completely by the SDC, from parameters supplied by the user through System Software.

EXTERNAL INTERFACE

Data set command signals are produced by the SDC under line procedure control, and data set status signals can be passed to the line procedure through the SDC status. The Data Set Interface is compatible with RS-232-C and CCITT V.24/V.26 data sets, and consists of all the circuits shown in Table 5–4. The Data Set Interface signals are converted to the required external voltage levels by the CWM board. Figure 5–2 shows the external interface between the SDC and the CWM board.

INTERNAL INTERFACE

The internal interface between the SDC and the processor is the I/O Common Interface described in Section 2 of this manual. Figure 5–2 shows the internal interface between the SDC and the processor.

SDC VARIABLE CHARACTERISTICS

Operating characteristics of the SDC can be altered to suit user requirements by means of parameters passed to the SDC by System Software. The variable characteristics are detailed below.

CHARACTER PARITY OPTIONS

The following parity options may be selected:

a) No character parity generation or checking (8 data bits)
b) Odd character parity (7 data bits + Parity)
c) Even character parity (7 data bits + Parity)

Refer to Sections 6 and 9 of this manual for details of System Software control of parity.

Block Check Options

Block Checking is a method of ensuring the integrity of text data. This is achieved by adding,to the end of the message, a bit pattern whose value is computed from the text data.

The following block checking options may be selected:

a) No block checking is performed
b) Longitudinal Redundancy Check (LRC) is generated and checked
c) Cyclic Redundancy Check (CRC) is generated and checked.

Refer to Sections 6 and 9 of this manual for details of System Software control of Block Checking.

The LRC is an 8 bit character (bits 0 through 7), formed by taking an exclusive – OR of all the characters to be included in the check. On transmission, if character parity is selected, the value of the LRC parity bit will be that which makes the LRC parity odd or even as appropriate. If parity is not

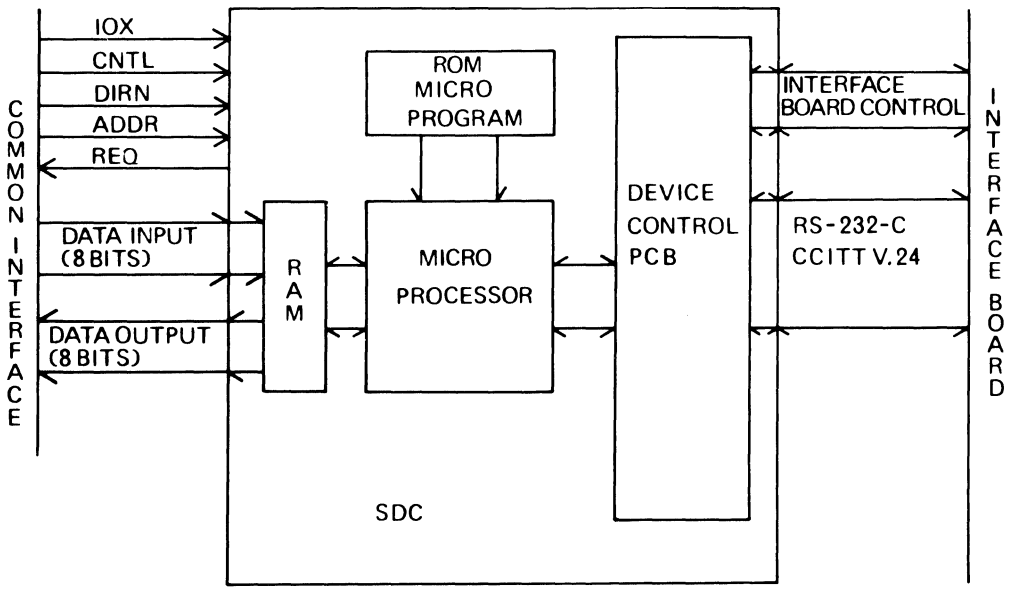


Figure 5–2. Synchronous Data Communications Controller (SDC)

selected, bit 7 (the parity bit) will be the exclusive – OR of all the other bit 7's of the message. Similarly, on receive, with parity selected, the LRC character will be checked for correct parity, and then the remaining 7 bits will be compared with the LRC accumulation. With parity not selected, an 8 bit comparison will be performed.

The CRC is a 16 bit quantity added to the end of a message block. Its value is calculated by taking the remainder obtained after dividing the binary value of the message by a polynomial. The polynomial used is $X^{16} + X^{15} + X^2 + X^0$.

Transmit/Receive Rate

The SDC transmit/receive rate is determined by the Transmitter and Receiver Signal Element Timing pulses from the data set. Some data sets permit selection of the transmit/receive rate by the SDC, through the data set signals "Select Standby" and "Data Signal Rate Selector", as shown below.

E.I.A. RS-232-C and CCITT V.24 Data Sets:

"Data Signal Rate Selector"	ON selects the higher rate of two available rates.

CCITT V.26 (facility 3) Data Sets:

"Select Standby"	OFF Selects the primary rate (2400 bits/second)
"Select Standby"	ON Selects the alternate rate. The alternate rate is determined by "Data Signal Rate Selector". ON = 1200 bits/second OFF= 600 bits/second

BPO DATEL MODEL 7C Data Sets:

"Select Standby"	OFF selects Phase Modulation at the following rates: "Data Signal Rate Selector" ON = 2400 bits/second OFF = 1200 bits/second
"Select Standby"	ON selects Frequency Modulation at the following rates: "Data Signal Rate Selector" ON = 1200 bits/second OFF = 600 bits/second

The SDC will condition the above signals as defined by the user through System Software. The user can specify the actual conditions required, or can specify that the downstream conditions will be sensed and copied.

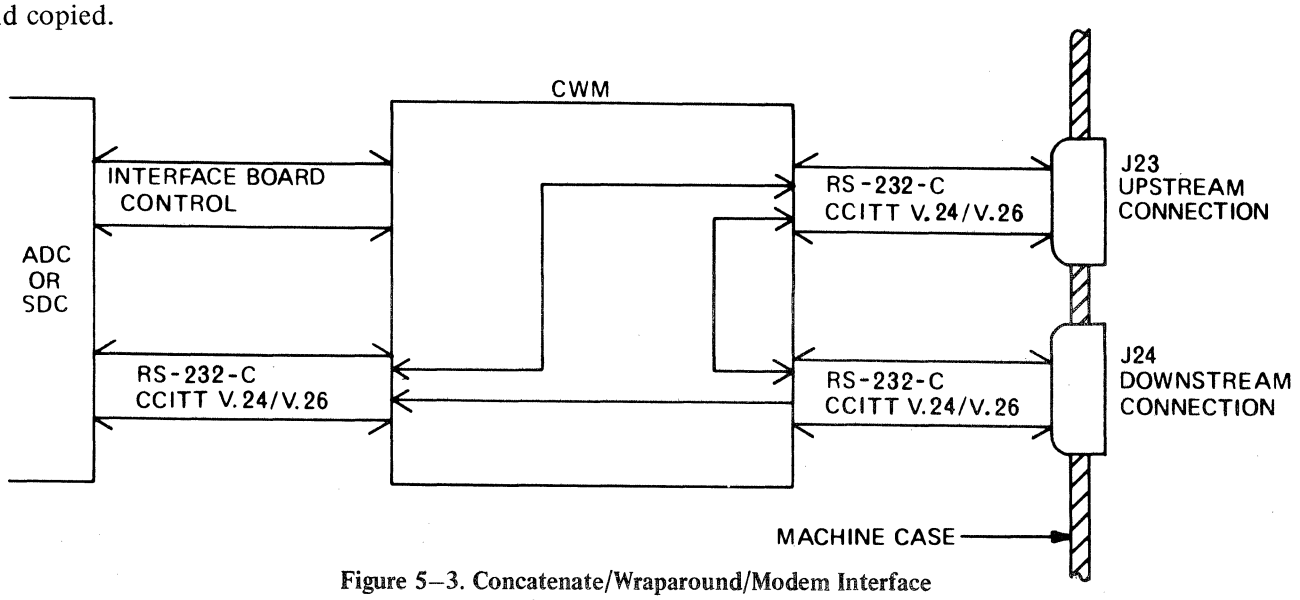


Figure 5–3. Concatenate/Wraparound/Modem Interface Board, CWM

CONCATENATE/WRAPAROUND/ MODEM INTERFACE (CWM)

The Concatenate/Wraparound/Modem interface (CWM) is a single printed circuit board which allows the Asynchronous Data Communications Controller (ADC) or the Synchronous Data Communications Controller (SDC) to be connected to a data set. System Software controlled "local wraparound" is provided by the CWM, permitting the Customer Confidence Program to test all functions of the data communications subsystem. Figure 5–3 shows the external connections from the CWM, and Figure 5–5 shows the position of these connections on the machine exterior.

CONNECTION TO DATA SET

The CWM provides on RS-232-C/CCITT V.24, V.26 compatible upstream connection, J.23, as shown in Figure 5–4(a). This connection may be coupled to a data set using an appropriate external cable. Table 5–5 shows the cables required for particular data set types.

Wire jumper selection is provided on the CWM board to permit the following:

a) "Clear to Send" may be strapped to "Request to Send", making the AE500, and all downstream systems, independent of the data set "Clear to Send" signal. This jumper must be installed if any Burroughs TC500 is positioned downstream.

b) "Data Terminal Ready" may be strapped to "Data Set Ready" if "Data Set Ready" is not available from the data set.

c) "Data Signal Rate Selector" may be replaced by "Select Transmit Frequency". Some low speed CCITT data sets provide two carrier frequencies to permit full duplex operation on two wires. Since these are low speed data sets, no "fall back" capability is required.

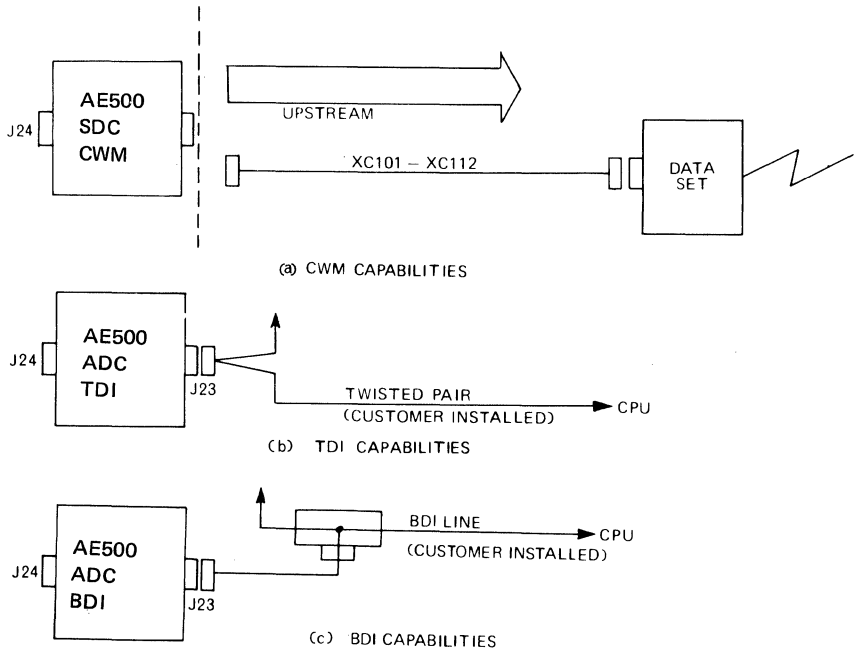


Figure 5–4. AE500 Connections

Table 5–5 shows the data communications cables available, and contains a brief description of each type of cable.

WRAPAROUND

The wraparound facility of the CWM permits full off-line testing of the data communications subsystem by providing System Software controlled connection of output signals to input signals. This facility is used by the Customer Confidence Program to allow the system operator to exercise the hardware functions of the data communications sybsystem. The Customer Confidence Program is described in Section 9 of this manual, and in greater detail in the AE500 System Software Operation Guide.

TWO WIRE DIRECT INTERFACE (TDI)

The Two Wire Direct Interface (TDI) is a single printed circuit board which allows the Asynchronous Data Communications Controller (ADC) to be installed in direct connect (TDI) networks. TDI permits connection of up to 9 terminals with TDI capability, using a maximum cable length of 1000 feet. TDI will support communication at the maximum rate provided by the ADC, that is 9600 bits per second. The wraparound functions required for the Customer Confidence Program are provided, and are identical to those furnished by the CWM. Figure 5–6 shows the external connections from the TDI, and figure 5–5 shows the position of these connections on the machine exterior.

TWO WIRE DIRECT INTERFACE

The TDI provides a two wire interface (signal and ground) for connection to other two wire direct connect interface (TDI) terminals. The two wire twisted pair connecting line is provided and installed by the user, and is connected to the system via J23. Two wire direct connection limits the AE500 to half duplex operation since only one signal line is available for message transmission. Figure 5–7 shows example configurations using the TDI. Note: TDI may not be used in BDI environments.

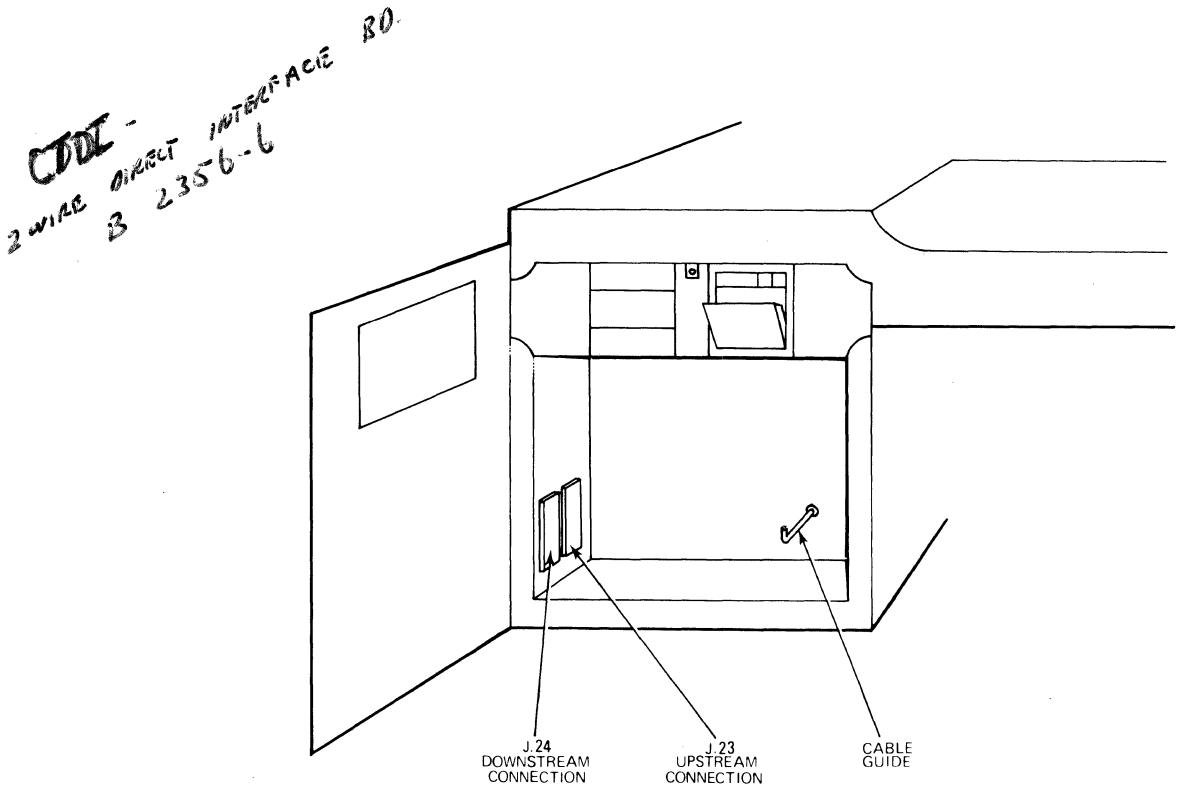


Figure 5–5. Data Communications Connectors

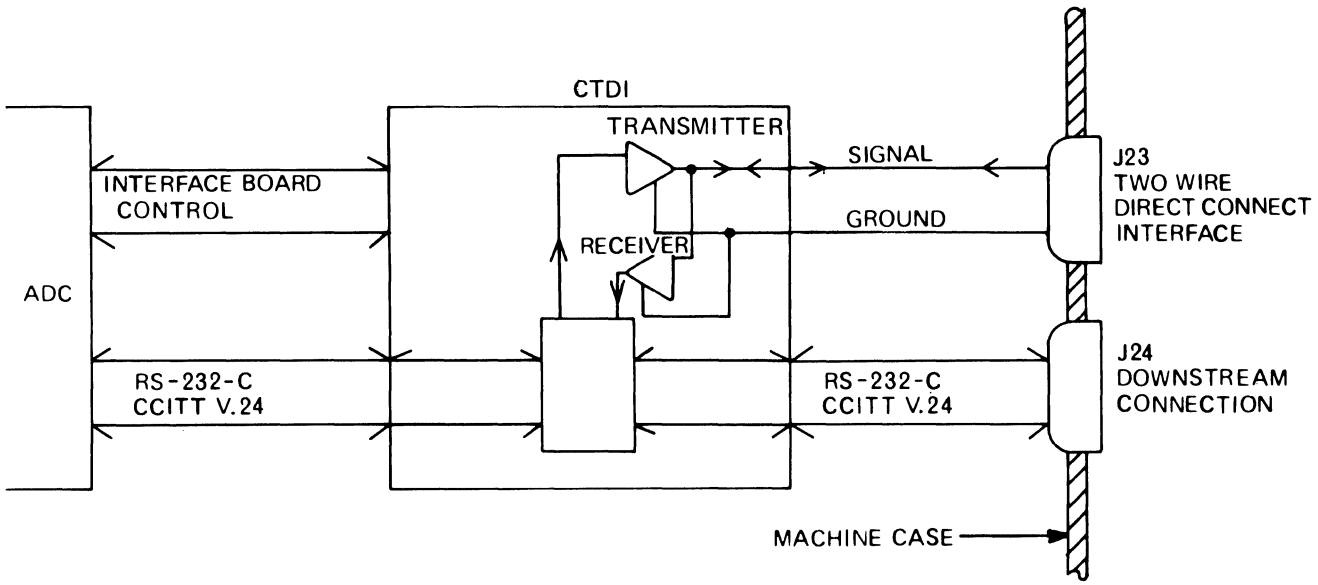
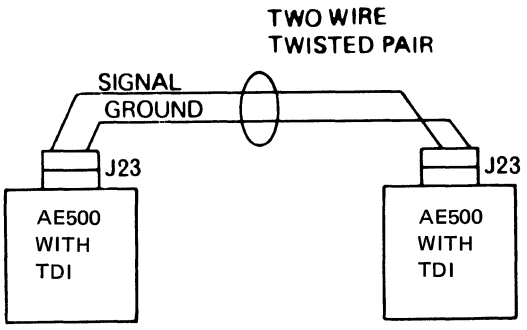


Figure 5–6. Two Wire Direct Connect Interface Board, TDI



TWO WIRE DIRECT CONNECT POINT-TO-POINT CONFIGURATION

Figure 5–7. Example TDI Configuration

Table 5–4. AE500 Data Set Interface

SYNC ONLY = S	Circuit Name EIA (CCITT) [Additional Circuit]	EIA RS-232-C Circuit	CCITT V24/V26 Circuit	Function of Circuit
		→ To data set ← To terminal		
	Signal Ground	AB	102	Provides common ground reference potential.
	Transmitted Data	BA →	103	Transfers bit serial data from Terminal to Data Set.
	Received Data	BB ←	104	Transfers bit serial data from Data Set to Terminal.
	Request to send	CA →	105	Instructs the Data Set to enter the transmit mode.
	Clear to Send (Ready for Sending)	CB ←	106	Informs the Terminal that the Data Set is conditioned to transmit data. Response to "Request to Send".
	Data Set Ready	CC ←	107	Informs the Terminal that the Data Set is "prepared" to transmit or receive data. "Prepared" requires "Data Terminal Ready" and any data set dependent "Data" buttons to be in the "Data" position.
	Data Terminal Ready	CD →	108/2	Informs the Data Set that the Terminal is ready for connection to the communication channel (line)
	(Connect Data Set to Line)	– →	108/1	Functions as "Data Terminal Ready".
	Ring Indicator (Calling Indicator)	CE ←	125	Informs the Terminal that the ringing signal is being received. The call will be answered by raising "Data Terminal Ready".
	Received line Signal Detector	DF ←	109	Informs the Terminal that a carrier frequency is being received.
	Data Signal Rate Selector	CH →	111	Conditions the Data Set for the high or low signalling rate, where applicable.
S	Transmitter Signal Element Timing	DB ←	114	Bit synchronising signal (clock) for synchronous transmission.
S	Receiver Signal Element Timing	DD ←	115	Bit synchronising signal (clock) for synchronous reception.
	(Select Transmit Frequency)	– →	126	Selects one of two carrier wave frequencies for certain CCITT low speed Data Sets.
S	(Select Standby)	– →	116	Causes the Data Set to switch to its standby facilities, where applicable.
	[Data Mode]	–	–	Is used in Auto dial and Auto answer modes with the Burroughs TA714 data set and the Bell CBS coupler
	[Originate]	–	–	Is used in Autodial and Auto answer modes with the Burroughs TA714 data set and the Bell CBS coupler.

Table 5–5. Data Communications Cables

AE500 TO DATA SET			
U.S., CANADA JAPAN, ITALY	U.K. AND BPO TYPES	S.A. AND SEL TYPES	LENGTH (feet)
BELL type	BPO type	SEL type	
XC101	XC105	XC109	15
XC102	XC106	XC110	25
XC103	XC107	XC111	50
XC104	XC108	XC112	100

Data communication cables should not be connected or disconnected unless the Data Communications Virtual Machine is inactive.

SECTION 6
VIRTUAL SYSTEM

INTRODUCTION

The virtual machine concept is that of a system in which the hardware functions are not directly controlled by the user program but by software which interprets the user program into hardware actions.

The virtual system software of the Burroughs AE500 series implements the Burroughs System Languages 3 and 5 (SL3 and SL5). The AE500 systems provide both user program (S-program) and Data Communication capabilities using a single processor by implementing two virtual machines. This implementation consists of the EXECUTIVE which controls the operation of the system, the BASIC VIRTUAL MACHINE which provides the means of interpreting the S-program instructions, and the DATA COMMUNICATIONS VIRTUAL MACHINE which is responsible for controlling the data communications hardware and maintaining the line disciplines.

Communication between the two virtual machines is controlled by the EXECUTIVE and is by means of SYSTEM REGISTERS. These registers are locations in memory which have specific meanings and can be accessed and modified by both virtual machines. Figure 6–1 shows the concept of communication between the various components of the virtual system.

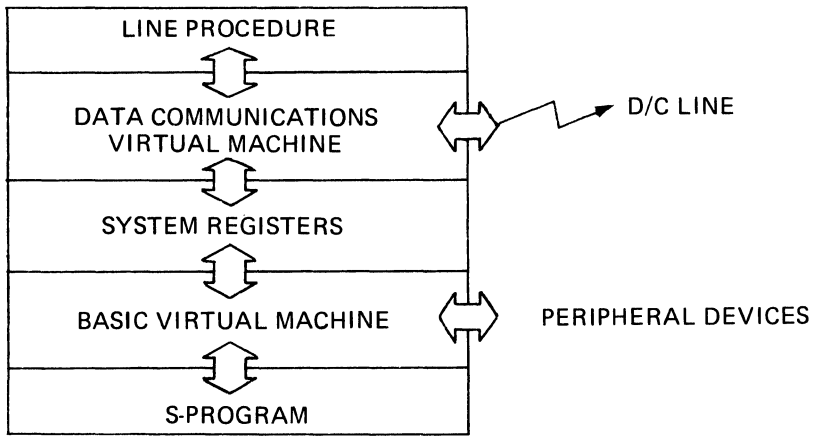


Figure 6–1. Virtual System Communication

The operation of the virtual system is now discussed in detail. The methods of loading the virtual system software (COLD START) are described at the end of this section.

EXECUTIVE

The executive provides the basic control states of the system, controls all interrupts from peripheral devices and decides which of the two Virtual Machines should be in use at any moment.

The system will normally be running in the BASIC mode except when the executive has sensed a Data Comm interrupt in which case execution is passed to the DATA COMM mode, for as long as is required to process that interrupt, before returning to the BASIC mode.

The system is placed under control of the executive at the end of the COLD START operation.

CONTROL STATES

The system control states are the various states in which the system can be at any time. There are three states within the EXECUTIVE and a fourth, the COLD STATE, which is described later. The three states are described below and the manner of changing from one state to another is outlined in figure 6–2.

READY STATE

This may be viewed as being the central or default state of the system as all program execution starts from here and all program termination causes control to return to here. From the READY STATE, there are five options open to the operator through the use of five keys (all other keys are ignored).

The READY KEY will cause the system to enter the STANDBY STATE.

The RESET KEY enables the system to re-start execution of an S-program that was terminated using the READY KEY. (If the resident S-program did not terminate in this way, depressing this key will have no actual effect).

PROGRAM SELECT KEY (PSK) 1 starts the execution of the S-program which is resident. (If there is none resident, immediately after performing COLD START, this key will be ignored).

PROGRAM SELECT KEY 2 enables the loading of object files into memory. A further selection to determine which peripheral device is to be used for loading must then be made. This option enables any object file, which has been dumped onto an appropriate medium from any SL3 or SL5 system, to be loaded.

PROGRAM SELECT KEY 3 enables the execution of intrinsics (utilities that were selected as resident, at COLD START). A further selection to determine which utility is to be executed has then to be made.

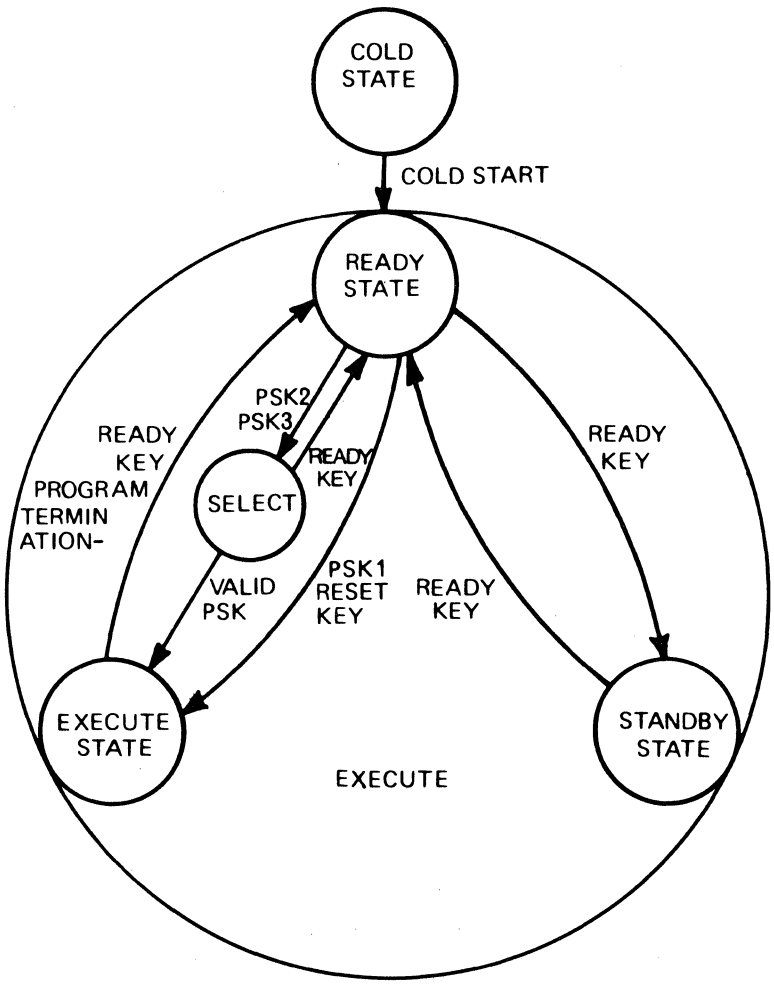


Figure 6–2. Virtual System States

STANDBY STATE

In this state, the system is inactive and is only sensitive to the READY KEY in order to place the system in the READY STATE. All data in memory is retained. The only light illuminated is the ON light.

EXECUTE STATE

The system is in this state when it is executing an S-program, loader routine or intrinsic utility. On termination of this state, the system will always enter the READY STATE. The methods of termination are:

1. Execution of a STOP instruction.
2. Execution of an invalid instruction (either an instruction for which there is no resident segment or an instruction in which the parameters are invalid).
3. Depression of the READY KEY.

BASIC VIRTUAL MACHINE

The Basic Virtual Machine is responsible for the implementation of S-program instructions (S-instructions) and can be considered as the defined interaction between S-instructions, the Interpreter, the System Registers and Hardware functions.

The precise implementation of a given S-instruction is dependent on the settings of the System Registers. An S-instruction is interpreted by micro-instruction strings (see INTERPRETER in this section). These micro-instructions use and modify certain System Registers to determine the specific hardware function to perform.

MEMORY ORGANISATION

The main Memory of the processor is functionally divided into two major areas: Interpreter Memory (including intrinsics and system registers) and User Memory.

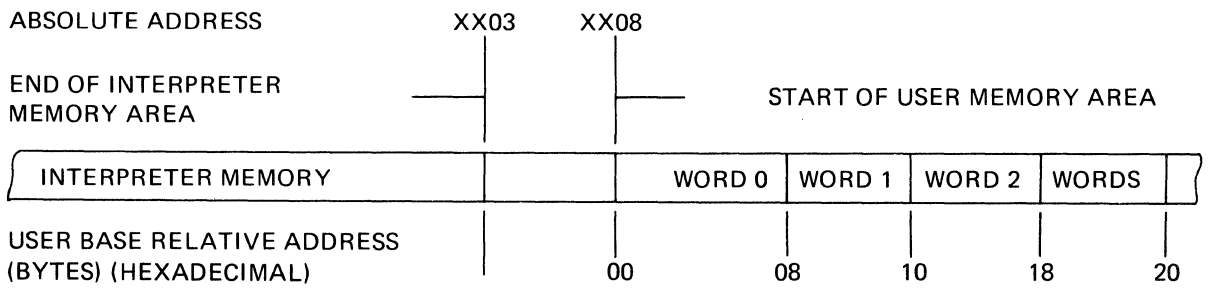
Interpreter Memory

The Interpreter memory area is of variable size depending on the optional interpreter segments and intrinsics included at Cold Start. Resident within the Interpreter Memory Area are the System Registers.

User Memory

User Memory is the Random Access Memory that is available to the user for storage and manipulation virtual machine instructions (S-instructions), data, and I/O buffers. The amount of available user memory is determined by the total RAM minus the size of the Interpreter Memory Area.

User Memory is word orientated and starts at the end of Interpreter Memory Area on the next (modulus 8) word boundary.



A word is 8 bytes (64 bits) in length and can be subdivided in 8 bytes (SL5) or 4 syllables (SL3). One byte can be further subdivided into two 4-bit digits. The numbering convention for these elements of a word is shown below.

WORD	ONE WORD															
BYTE	0		1		2		3		4		5		6		7	
DIGIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

	1 BYTE							
	1 DIGIT				1 DIGIT			
BITS	7	6	5	4	3	2	1	0

SL3 Instruction Format

System Language 3 (SL3) is a virtual machine instruction set organised in memory with four instructions per word. Each instruction consists of 4 hexadecimal digits (2 bytes). These digits are identified, from left to right, as OP code upper, OP code lower, Parameter upper, and Parameter lower, within a syllable.

The relationship of SL3 instructions to memory is as follows:

One SL3 instruction = 4 Digits
= 2 Bytes
= 1 Syllable

Four SL3 instructions = 16 Digits
= 8 Bytes
= 4 Syllables
= 1 Word

Each word of memory is ordered as follows:

WORD	ONE WORD															
SYLLABLES	3				2				1				0			
CHARACTERS (bytes)	0		1		2		3		4		5		6		7	
DIGITS	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Each instruction resides in one syllable of memory and begins and ends on a syllable boundary. That is, the instruction does not cross the syllable boundary.

Processing begins at Word 0, syllable 0

WORD	WORD 0																WORD 1													
SYLLABLE	3				2				1				0				3				2				1				0			
CHARACTER	0		1		2		3		4		5		6		7		0		1		2		3		4		5		6		7	
DIGIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																
CODE	D	F	0	1	C	0	A	A	E	B	0	9	E	D	0	1	E	B	0	9	E	D	0	1	C	8	–	–	E	B	0	F
ORDER OF EXECUTION	4				3				2				1				8				7				6				5			

In general the order of execution (assuming no programmatic jumps) is:

Word	Syllable
n	0
n	1
n	2
n	3
n+1	0
n+1	1

etc.

S L5 Instruction Format

System Language 5 (SL5) is a variable length virtual machine instruction set. It implements the SL3 functions, expanding the memory addressing ability and literal representation in the Parameter field, where required, to give a three byte instruction. The two byte configuration is retained for those instructions that do not involve memory addressing or require extension of the Parameter field.

Since SL5 instructions are two and three bytes, the memory organization differs from that of SL3.

The relationship between SL5 instructions and memory is as follows:

One Type 1 Instruction	=	4 Digits
	=	2 Characters
	=	2 Bytes
One Type 2 Instruction	=	6 Digits
	=	3 Characters
	=	3 Bytes

User memory is segmented into words as follows:

WORD	ONE WORD															
CHARACTERS	0		1		2		3		4		5		6		7	
DIGITS	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Processing starts at Word 0, character 0 of user Memory. Instructions may cross word boundaries:

	WORD 0																WORD 1															
CHARACTER	0		1		2		3		4		5		6		7		0		1		2		3		4		5		6		7		0	
DIGIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	15	14
CODE	E	D	0	1	E	B	0	A	C	0	4	A	D	F	0	1	0	0	E	B	0	F	C	8	0	1	0	8	E	D	0	1	E	B
INSTRUCTION TYPE	1				1				1				2				1				2				1									

EXECUTION STARTS HERE

ORDER OF EXECUTION

DATA COMMUNICATIONS VIRTUAL MACHINE

The Data Communications Virtual Machine is responsible for controlling the data communication hardware, and maintaining the line discipline. It communicates with the Basic Virtual Machine through system registers as depicted in figure 6–1. The exact use of each system register is line procedure dependent, and is described in Section 8 of this manual, but the basic operation is: the Basic Virtual Machine will not access the transmit buffer while the Transmit Ready Flag (D3) is set, and the Data Communications Virtual Machine will not access the receive buffer while the Message Received Flag (D2) is set. The message flow is shown in figure 6–3.

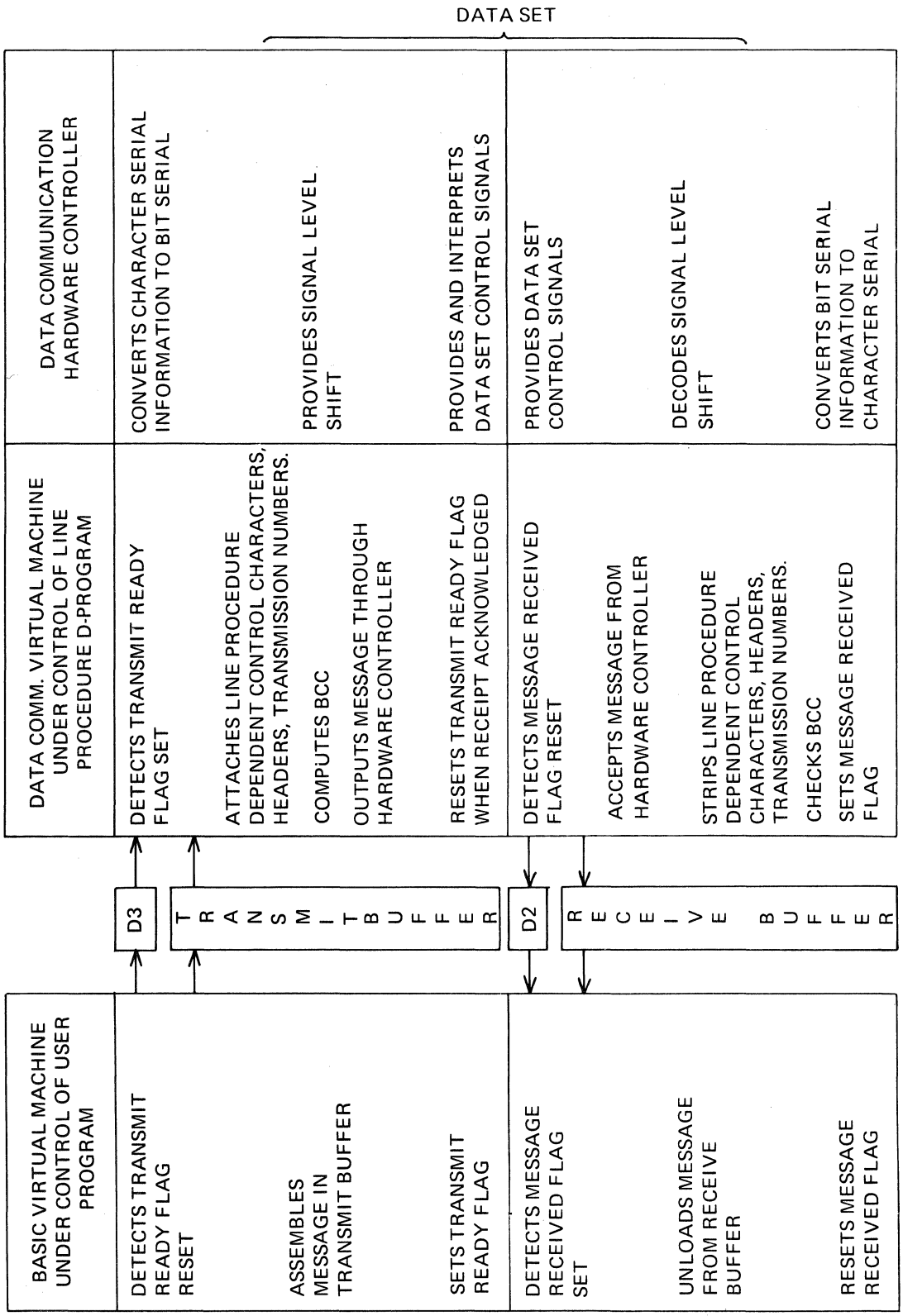


Figure 6–3. Transmit and Receive Data Paths

The Data Communication Virtual Machine operates independently, under control of the active line procedure program. Since the discrete functions of most line procedures are very similar, an "interpretive" structure is used to implement them. Functions which are common to all line procedures have micro-coded routines written to handle them, and an "operation code" is assigned to each.

Line procedure programs consist of sequences of these operation codes. A "program counter" in the Data Communication Virtual Machine points to the next operation code to be interpreted, and a "fetch and decode" micro program routine causes the appropriate handling routine to be executed. The function represented by an operation code is known as an instruction. Instructions are provided to control the data communication hardware, to act upon the Data Communication Virtual Machine registers, and to act upon the system registers which provide the communication between the Data Communication and Basic Virtual Machines. These instructions are referred to as "D-codes", and the line procedure programs are known as "D-programs".

IMPLEMENTATION USING A SINGLE PROCESSOR

As stated earlier in this section, the Data Communications Virtual Machine shares the same physical processor as used by the Basic Virtual Machine. The Data Comm Virtual Machine is given control of the processor when an interrupt is received from a data communications hardware controller. The interrupt is processed by the Data Communications Virtual Machine through interpretation of the Line Procedure D-instructions. When the interrupt has been serviced, the processor is restored to the Basic Virtual Machine. This mechanism is illustrated in figure 6–4.

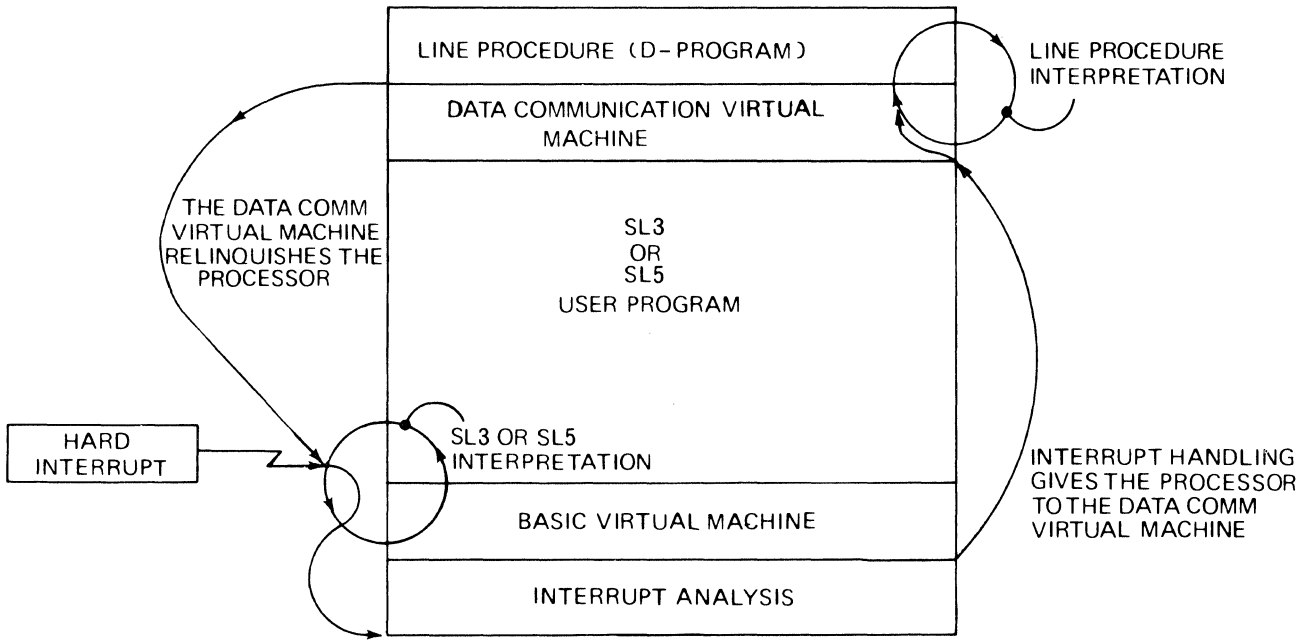


Figure 6–4. Sharing of Processor by the Two Virtual Machines

VIRTUAL MACHINE COMMUNICATION

The communication between the two virtual machines is provided by means of system registers, as shown in figure 6–1. Some of these registers define the hardware operating characteristics required and must normally be loaded before line procedure initialisation. Other registers are associated with the parameters of a line discipline, and may normally be loaded with the line procedure active. Finally, there are registers which provide the direct communication between the two virtual machines, and these are always used when the line procedure is active. Section 8 of this manual describes the use of these registers by each line procedure.

SPECIAL CONSIDERATION

In order to fully utilise the Data Communication facilities of AE500 systems, the DATA HANDLING 1 optional segment must be loaded at COLD START. This segment provides the buffer handling instructions which are needed to load and unload the data comm buffers.

INTERPRETER

The interpreter consists of the micro-code required to implement the Virtual System. It also contains the System Registers, whose functions are described in Sections 7 and 8. The interpreter is wholly resident in memory and is divided as shown in figure 6–5.

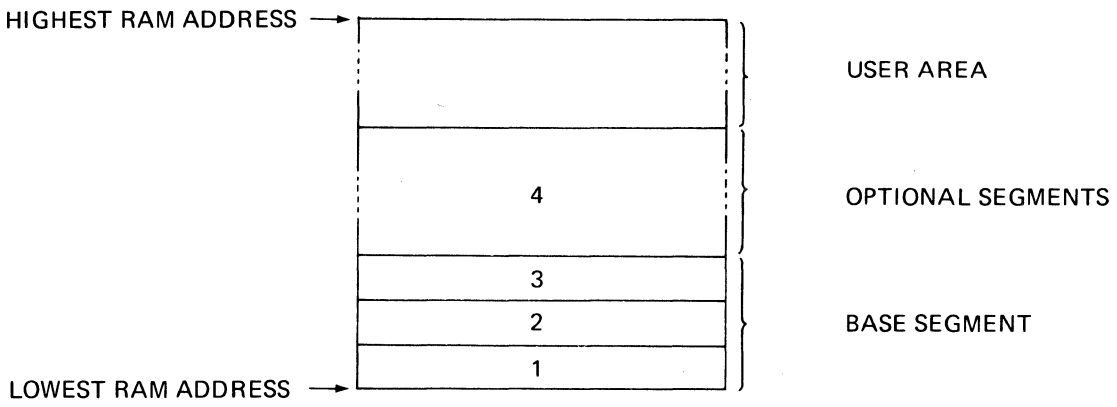


Figure 6–5. SL3/SL5 Memory Division

The interpreter is divided into a Base Segment and various Optional Segments. The optional segments contain the software to handle special software features, system peripherals and data communications line procedures. A list of the S-instructions implemented by each segment is located in Appendix D of this manual.

BASE SEGMENT

The Base Segment consists of 3 parts: Interpreter Registers and Work Areas, Micro-coded routines and Basic S-language instruction routines. These correspond, respectively, with items 1, 2 and 3 on figure 6–5. Each part is discussed below in detail.

Interpreter Registers

Basic System Registers – These are the registers which define the system and are described fully in Section 7.

User Registers – These are described in Section 7.

Interpreter Work Areas

Table of entry points to Global Subroutines – The starting locations of general purpose interpreter routines are stored in this table. This permits optional segments to make use of these subroutines.

Peripheral Tables – These two tables are created at Cold Start by the Bootstrap Loader located in Read Only Memory. The tables show which peripheral device is present on each Input/Output channel in (a) channel order and, (b) device order.

Micro-coded Routines

Global Subroutines – These are general micro-programmed subroutines (for example: get keyboard character routine) which may be needed by any part of the interpreter. A table, stored in a special area, contains the addresses of these routines so that they can be accessed from any part of the interpreter.

Initialisation Routine – This routine places the system in the Ready State. It is entered from: STOP user program or utility, system errors, the STANDBY state on depression of the Ready key.

Interrupt Handling Facilities

Fetch Routine and Decode Tables. – The fetch routine fetches the next S-language instruction to be interpreted and the decode tables contain addresses of the micro-instruction strings used to interpret S-instructions.

Basic S-Language Instruction Routines

These are the micro-instruction strings used to interpret the basic S-instruction set.

OPTIONAL SEGMENTS

There are three types of optional segment: utilities, which are described in Section 9; data communications line procedures, which are described in Section 8; and code segments, which are described below. Each code segment provides the micro-coded interpreter to implement the handling of a peripheral device, translation table or special software routine.

Keyboard and Printer Translation Segment

This segment provides a mapping of codes from those produced by the keyboard hardware to the correct internal representation (ASCII), and from the internal representation to the codes required by the serial printer. This enables different alphanumeric keyboard variants to be implemented just by changing this portion of software. Only one such translation segment must be resident at any one time.

The options currently available for this segment are:

1. U.S.A
2. U.K.
3. French
4. Italian
5. German
6. Portuguese
7. Spanish
8. Scandinavian
9. Croatian
10. Finnish/Swedish

A specification of each keyboard variant and of the internal representation of the codes is located in Appendix B.

Data Communications Base Segment

This segment provides the micro-coded routines to implement the data communications S-instructions and the special D-instruction set in which each line procedure is written. It also provides the data comm system registers which define the data communications features of the system.

The D-instruction set is a special set of instructions which have been created in order to provide a more efficient implementation of line procedures. Each procedure is written using these instructions as the operation of them is different from that of normal optional segments.

Data Communication Transmit and Receive Buffers (SL3)

These buffers, together with the code required to set up various related system registers, must be resident in the interpreter if SL3 data comm activity is to be present. The buffers for SL5 S-programs are located in the S-program itself.

Data Handling 1 Segment

This segment provides for the implementation of data handling S-instructions. "Data handling" is a term used to describe the capability to move and access data byte by byte, as opposed to the basic method which allows movement and access word by word only.

Magnetic Tape Cassette Input/Output (I/O) Segment

This segment provides the routines for the handling of magnetic tape cassette I/O S-instructions. These permit the reading and writing of data from and to the tape and also the positioning of the tape to specified positions.

Magnetic Tape Cassette Translation Tables

These tables provide a mapping of eight-bit codes read from a tape to the desired internal representation, and from the internal representation to the desired eight-bit codes to be written to the tape. These tables are produced by the user using the Translation Table Generator program. The system will use or ignore the translation tables, depending upon the setting of the TAPE CASSETTE INPUT and OUTPUT TRANSLATION System Registers. If either table is not resident, and the associated System Register specifies "translate", then a system condition will result. System Conditions are described in Appendix C.

Software Special Segment

This segment contains some special routines which are not available to the user but are required for the operation of certain utilities.

Check Digit Compute and Verify (CDC/V) Segment

This segment provides the code to implement the CDC and CDV S-instructions. These permit the automatic computation and verification of suffixed check digits by referencing user-defined tables.

COLD START

The COLD START is the method used to load all the Virtual System Software into the system. It is necessary to perform this procedure whenever the system is in the COLD STATE.

The system is in the COLD STATE when the Read/Write Memory contains no valid information. This will occur when power is applied to the system or when the Cold Start Button is pushed.

The Virtual System software is stored on Magnetic Tape Cassette in one of two formats. First, there is the Master Firmware Cassette (MFC) which contains all the software that is currently released for this system. It includes Interpreter Segments, Data Comm Line Procedures, Intrinsics and Utilities, all of which are described more fully in this or succeeding sections. This MFC has inbuilt routines that enable the operator to "tailor" the software to the needs of the user at Cold Start by selecting which interpreter segments/intrinsics are required. Second, there is the Tailored Firmware Cassette (TFC) which contains only the "tailored" interpreter as selected by the operator. This TFC is created at the end of the Cold Start process which uses an MFC.

The COLD START operation is described in detail in the Burroughs AE500 Systems Software Operation Guide. A brief description only is given here. After placing the system in the COLD STATE, the operator will depress one of the two enabled PSK's to start the cold start routine. The system will load, from the MFC, an M-LOADER together with its associated M-INTERPRETER. This loader routine is then executed. Each segment/intrinsic on the MFC consists of a header followed by data. The routine will read the header and display a question to the operator. If the answer is affirmative, the data portion will be loaded, otherwise the data portion is skipped until the next segment/intrinsic is reached. Each segment loaded will be placed adjoining the previously loaded segment so that memory is fully utilised. This loading operation continues until either the operator has decided, as all necessary segments have been loaded, to terminate or when the final segment on the MFC has been loaded. At this point, the operator must decide whether or not to create a TFC incorporating the loaded software. If a TFC is to be created the operator must also decide whether or not to invoke the AUTOLOAD option. This option permits the automatic loading of the file following the interpreter on a TFC when that TFC is used to load the system in the COLD START. At the end of processing the option selected, the system will be placed under the control of the EXECUTIVE and enter the READY STATE.

If performing the COLD START from a TFC, the system will load the TFC automatically and enter the READY STATE as before.

SECTION 7
SYSTEM REGISTERS

The System Registers provide the Virtual System with a soft definition of the configuration. This includes equipment features, micro-program features and operating environment.

All system registers are resident in read/write memory and must be loaded appropriately prior to the use of the feature to which they relate. This will often be prior to program execution, but some may also be changed as a result of program execution.

There are three groups of system registers and they are described fully in the following sections. The groups are: Basic Registers, Data Communication Registers and User Registers.

Each register is described in terms of: its function, format, possible values, how and when loaded, when required, default values and other related system registers. The default values are the values to which the register is set either as a result of System Generation or Loading (Cold Start) or as a result of starting the execution of a resident S-program using PSK1. In most cases, the use of PSK1 from the Ready State will not affect the value in a particular register and is indicated by no default value being quoted.
None of the registers mentioned in this chapter are transportable to any other system. However, certain of the Basic System Registers may be loaded from files created on the TC3500/L8000 systems.

BASIC REGISTERS

This section defines each of the Basic System Registers. Most registers of this group are set up either via the Basic System Register Utility which allows the system register values to be loaded into the registers and/or written to cassette as an object file, or, by loading such an object file using the cassette S-Loader. These systems registers must be loaded appropriately before any use of the system.

USER MEMORY BASE ADDRESS REGISTER

FUNCTION

This register defines the absolute base location of memory that is available for S-programs. It is transparent to the user.

FORMAT

The register size is 16 bits.

LOADING

This register value is automatically generated by the interpreter at System Generation or Load.

USER MEMORY LIMIT ADDRESS REGISTER

FUNCTION

This register defines the absolute upper limit of memory that is available for S-programs. It is transparent to the user.

FORMAT

The register size is 16 bits.

LOADING

This register value is automatically generated by the interpreter at System Generation or Load.

TYPE OF VIRTUAL MACHINE REGISTER

FUNCTION

This register specifies the S-program language, including all subsets, that the system is to interpret.

FORMAT

The register size is 2 bits.

POSSIBLE VALUES

00 = SL3 BASIC, 32 Track, Non Data Comm
01 = SL3 40 Track.
10 = SL3 Data Comm, Tape Cassette.
11 = SL5

LOADING

This register must be set to the required value before program start and may be loaded by one of the following methods:

a) Basic System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Basic System Register Generator or, as part of a program or data object file. This object file may have been created on a TC3500/L8000 system).

REQUIRED

This register must always be specified.

DEFAULT VALUES

System Generate/Load – 11 (SL5)

RELATED REGISTERS

Only if SL3 Data Comm is specified:

SL3 DATA COMM LINE PROCEDURE REGISTER
SL3 CTCC/MESSAGE UNPACK REGISTER

CLEAR PRINT HEAD TIME-OUT REGISTER

FUNCTION

This register specifies the time which the system waits before entering the printer Visibility Mode. It can be altered to suit the application program to be executed.

FORMAT

The register size is 8 bits and it contains a count expressed in hexadecimal. Each count is equivalent to 41 ms.

POSSIBLE VALUES

0.0 to 9.9 seconds. A value of zero will disable the Visibility Mode feature.

LOADING

This register must be set to the required value before program start and may be loaded by one of the following methods:

a) Basic System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Basic System Register Generator or, as part of a program or data object file. If this object file had been created on a TC3500/L8000 system, a value of 0 will be loaded into this register).

REQUIRED

This register must always be specified.

DEFAULT VALUES

System Generate/Load – 1.2 seconds.

MONETARY SAFEGUARD SYMBOL REGISTER

FUNCTION

This register defines which ASCII character is to be used as the monetary safeguard symbol for printing numeric values.

FORMAT

The register size is 8 bits and its format is the column and row numbers of the ASCII character desired.

POSSIBLE VALUES

Any printable ASCII character.

LOADING

This register must be set to the required value before program start and may be loaded by one of the following methods:

a) Basic System Register Generator
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Basic System Register Generator or, as part of a program or data object file. This object file may have been created on a TC3500/L8000 system).

REQUIRED

This register must always be defined.

DEFAULT VALUES

System Generate/Load – 24 ($)

EUROPEAN PUNCTUATION REGISTER

FUNCTION

This register specifies whether or not European punctuation is to be used in printing numeric values.

FORMAT

The register size is 1 bit.

POSSIBLE VALUES

0 = Non-European punctuation (99,999.99)
1 = European punctuation (99.999,99)

LOADING

This register must be set to the required value before program start and may be loaded by one of the

following methods:

a) Basic System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Basic System Register Generator or, as part of a program or data object file. This object file may have been created on a TC3500/L8000 system).

REQUIRED

This register must always be specified.

DEFAULT VALUES

System Generate/Load – 0 (Non-European punctuation).

TAPE CASSETTE READ RETRY REGISTER

FUNCTION

This register specifies the number of automatic read retries that are to be performed by the interpreter before a Read Error is declared.

FORMAT

The register size is 4 bits.

POSSIBLE VALUES

0 to 15 retries (hexadecimal 0 to F).

LOADING

This register must be set to the required value before program start and may be loaded by one of the following methods:

a) Basic System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Basic System Register Generator or, as part of a program or data object file. This object file may have been created on a TC3500/L8000 system).

REQUIRED

This register must always be specified whenever tape cassette input is to be used by an S-program.

DEFAULT VALUES

System Generate/Load – 6 retries.

CONSIDERATIONS

This register should be set to 0 if the S-program contains its own read retry routine.

TAPE CASSETTE WRITE RETRY REGISTER

FUNCTION

This register specifies the number of automatic write retries that are to be performed by the interpreter before a Write Error is declared. This register does not apply to retries of a tape mark write instruction.

FORMAT

The register size is 3 bits.

POSSIBLE VALUES

0 to 4 retries.

LOADING

This register must be set to the required value before program start and may be loaded by one of the following methods:

a) Basic System Register Generator
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Basic System Register Generator or, as part of a program or data object file. This object file may have been created on a TC3500/L8000 system).

REQUIRED

This register must always be specified whenever tape cassette output is to be used by an S-program.

DEFAULT VALUES

System Generate/Load – 4 retries.

CONSIDERATIONS

This register MUST be set to 0 if the S-program contains its own write retry routine.

TAPE CASSETTE TAPE MARK WRITE RETRY REGISTER

FUNCTION

This register specifies the number of automatic write retries of tape marks that are to be performed by the interpreter before a Write Error is declared.

FORMAT

The register size is 2 bits.

POSSIBLE VALUES

0 to 3 retries.

LOADING

This register must be set to the required value before program start and may be loaded by one of the following methods:

a) Basic System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Basic System Register Generator or, as part of a program or data object file. This object file may have been created on a TC3500/L8000 system).

REQUIRED

This register must always be specified whenever tape cassette output is to be used by an S-program.

DEFAULT VALUES

System Generate/Load – 2 retries.

CONSIDERATIONS

This register MUST be set to 0 if the S-program contains its own write retry routine for tape marks.

TAPE CASSETTE INPUT TRANSLATION REGISTER

FUNCTION

This register specifies whether or not a translation table is to be used for converting data, read from a cassette tape, into ASCII.

FORMAT

The register size is 1 bit.

POSSIBLE VALUES

0 = YES
1 = NO

LOADING

This register must be set to the required value before program start and may be loaded by one of the following methods:

a) Basic System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Basic System Register Generator or, as part of a program or data object file. This object file may have been created on a TC3500/L8000 system).

REQUIRED

This register must always be specified whenever tape cassette input is to be used by an S-program.
NOTE: If a translation table is to be used, this table MUST have been declared and loaded at System Generation. Otherwise, a System Error will occur when a cassette read instruction is executed.

DEFAULT VALUES

System Generate/Load – 1 (NO)

TAPE CASSETTE OUTPUT TRANSLATION REGISTER

FUNCTION

This register specifies whether or not a translation table is to be used for converting ASCII data that is to be written to a cassette tape.

FORMAT

The register size is 1 bit.

POSSIBLE VALUES

0 = YES
1 = NO

LOADING

This register must be set to the required value before program start and may be loaded by one of the following methods:

a) Basic System Register Generator
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Basic System Register Generator or, as part of a program or data object file. This object file may have been created on a TC3500/L8000 system).

REQUIRED

This register must always be specified whenever tape cassette output is to be used by an S-program.
NOTE: If a translation table is to be used, this table MUST have been declared and loaded at System Generation Otherwise, a System Error will occur when a cassette write instruction is executed.

DEFAULT VALUES

System Generate/Load – 1 (NO)

SL3 DATA COMM LINE PROCEDURE REGISTER

FUNCTION

This register specifies the identification of the line procedure that is to be used by an SL3 S-program. It is only applicable to SL3 Data Comm/Data Handling type S-programs.

FORMAT

The register size is 8 bits. It contains the 2 digit hexadecimal identification of the line procedure.

POSSIBLE VALUES

Any valid line procedure identification or, 00 if the SL3 S-program is to operate in a non data comm environment.

LOADING

This register must be set to the required value before program start and may be loaded by one of the following methods:

a) Basic System Register Generator
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Basic System Register Generator or, as part of a program or data object file. This object file may have been created on a TC3500/L8000 system).

REQUIRED

This register must always be specified whenever SL3 Data Comm/Data Handling functions are to be used by an S-program.
NOTE: If a line procedure identification is specified, this procedure MUST have been declared and loaded at System Generation. Otherwise, a System Error will occur when the system attempts to initiate the line procedure at Program Start.

DEFAULT VALUES

System Generate/Load – 00

RELATED REGISTERS

TYPE OF VIRTUAL MACHINE REGISTER

SL3 CTCC/MESSAGE UNPACK REGISTER

FUNCTION

This register specifies whether or not Central Terminal Computer Controller (CTCC) and/or Message Unpack functions are to be used by the S-program. This register only applies to SL3 S-programs.

FORMAT

The register size is 1 bit.

POSSIBLE VALUES

0 = NO
1 = YES

LOADING

This register must be set to the required value before program start and may be loaded by one of the following methods:

a) Basic System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Basic System Register Generator or, as part of a program or data object file. This object file may have been created on a TC3500/L8000 system).

REQUIRED

This register must always be specified whenever an SL3 S-program is to be executed.

DEFAULT VALUES

System Generate/Load – 0 (NO)

RELATED REGISTERS

Type of virtual machine register.

DATA COMMUNICATIONS REGISTERS

This section defines each of the Data Communications System Registers. The registers of this group may be set up using one or more of the following methods: by the Data Comm System Register Utility which allows the system register values to be loaded into the registers and/or written to cassette as an object file; by loading such an object file using the cassette S-Loader; or by the use of S-program instructions. These registers must be loaded appropriately before any use of the features which they represent. The use of many of these registers is dependant on the line procedure in use.

To determine which registers are required for each line procedure in use, reference should be made to the Data Communications Line Procedure section.

Certain registers in this group are arranged together in a contiguous area of memory, 16 bytes long, and can be referenced individually by specifying the byte within this area using special SL5 instructions. The registers and their locations are:

BYTE	REGISTER
0	Idle Line Timer Register
1	Data Set Register Group
2 } 3 }	D/C Buffer Length Register
4	Procedure Area 1
5	Transmission Number Register Group
6	Synchronous Compare Character Register
7	Transmit/Receive Flags Register Group
8	D/C Operation Mode Register Group
9	Transmission Rate Register Group
10	Four Wire Receive Delay Register
11	Four Wire Transmit Delay Register
12	Procedure Area 2
13	Ring Limit Register
14	Two Wire Receive Delay Register
15	Two Wire Transmit Delay Register

IDLE LINE TIMER REGISTER

FUNCTION

This register specifies the count value which establishes the Idle Line Timer count. The timer allows for a wait period during a state of no line activity before a disconnect is initiated.

FORMAT

The register size is 8 bits and it contains the value represented in hexadecimal. Each count is equivalent to 1 second.

POSSIBLE VALUES

Any value from 0 to 255.

LOADING

This register must be loaded with the desired value prior to the line procedure initiating any activity which uses the Idle Line Timer.
It may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions (SL5 only).

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUES

System Generate/Load – 1

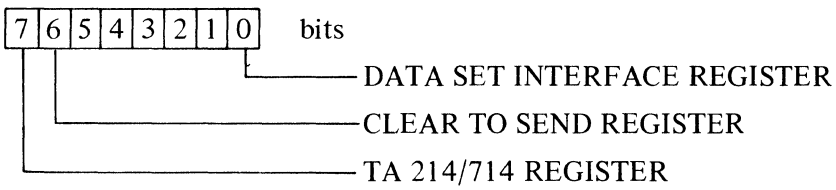
DATA SET REGISTER GROUP

FUNCTION

This register group defines the characteristics of the data sets (modems) and/or interfaces that are used to connect the system to the network.

FORMAT

The register group size is 8 bits and it contains the following registers:



LOADING

This register group must be loaded with the desired value before the line procedure is initiated.
It may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions (SL5 only).

REQUIRED

This register group must always be defined for Data Comm activity.

DATA SET INTERFACE REGISTER

FUNCTION

This register specifies whether or not the line procedure is to set up the Data Set Interface (DSI) at initialisation. The procedure must always set up the DSI except when another procedure has been used to physically connect the two portions of the network together, thus having set up the DSI already.

FORMAT

The register size is 1 bit and it is part of the DATA SET REGISTER GROUP.

POSSIBLE VALUES

0 = NO
1 = YES (Bypass DSI)

LOADING

See the DATA SET REGISTER GROUP.

REQUIRED

This register must always be specified for Data Comm activity.

DEFAULT VALUES

System Generate/Load – 0 (DO NOT Bypass DSI).

RELATED REGISTERS

CLEAR TO SEND REGISTER.

CLEAR TO SEND REGISTER

FUNCTION

This register specifies whether or not the line procedure will wait for a period, defined by the TWO or FOUR WIRE TRANSMIT DELAYS, before looking at the "CLEAR TO SEND" line from the data set (modem) for initiating transmission.

FORMAT

The register size is 1 bit and it is part of the DATA SET REGISTER GROUP.

POSSIBLE VALUES

0 = NO (Line procedure looks at "CLEAR TO SEND" immediately)
1 = YES (Line procedure waits)

LOADING

See the DATA SET REGISTER GROUP.

REQUIRED

This register must always be specified for Data Comm activity.

DEFAULT VALUES

System Generate/Load – 0 (look at CTS immediately).

RELATED REGISTERS

DATA SET INTERFACE REGISTER
TWO WIRE TRANSMIT DELAY REGISTER
FOUR WIRE TRANSMIT DELAY REGISTER
TWO/FOUR WIRE REGISTER

TA 214/714 REGISTER

FUNCTION

This register defines the characteristics of the TA214 or TA714 data set, if it is incorporated in the network. The precise function is dependant on the line procedure.

FORMAT

The register size is 1 bit and it is part of the DATA SET REGISTER GROUP.

POSSIBLE VALUES

The values are dependant on the line procedure.

LOADING

See DATA SET REGISTER GROUP.

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUES

System Generate/Load – 0
Data Comm System Register Generator – 0

D/C BUFFER LENGTH REGISTER

FUNCTION

This register defines the value which represents the maximum length of text data that is to be received or transmitted. It includes the ETX or NUL code.

FORMAT

The register size is 16 bits and it contains the size of the buffer as a hexadecimal number of bytes.

POSSIBLE VALUES

SL3 – 256 bytes only
SL5 – Any size within the bounds of user memory.

LOADING

This register must be loaded with the desired value prior to the line procedure receiving or transmitting text data. The register may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions (SL5 only).

REQUIRED

This register must always be specified for Data Comm Activity.

DEFAULT VALUES

System Generate/Load – 0
Program Start (SL3 only) – 256 bytes (Hex 0100)

PROCEDURE AREA 1

FUNCTION

This register specifies, to the line procedure, any extra variable information that may be necessary to properly execute that line procedure.

FORMAT

The register size is 8 bits.

POSSIBLE VALUES

The values are dependant on the line procedure.

LOADING

The time at which the register is loaded is dependant on the line procedure.
The register may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions (SL5 only).

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUES

System Generate/Load – 0 (Zero)

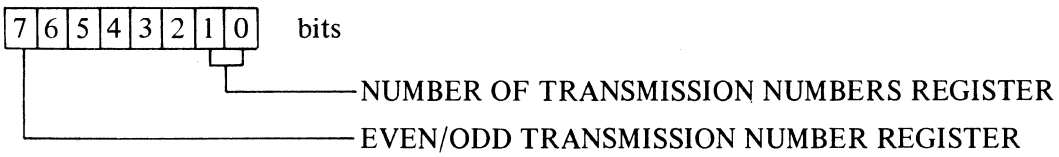
TRANSMISSION NUMBER REGISTER GROUP

FUNCTION

This register group defines the format of the transmission numbers that are to be present in both transmitted and received messages.

FORMAT

The register group size is 8 bits and it contains the following registers:



LOADING

This register group must be loaded with the desired value prior to the line procedure referencing a transmission number.
The registers may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions (SL5 only).

REQUIRED

This register group must always be defined for Data Comm activity.

NUMBER OF TRANSMISSION NUMBERS REGISTER

FUNCTION

This register defines the length of the decimal transmission number that is to be incorporated in the headers of transmitted messages and that is to be expected in the headers of received messages.

FORMAT

The register size is 2 bits and it is part of the TRANSMISSION NUMBER REGISTER GROUP.

POSSIBLE VALUES

00 = No transmission numbers
01 = 1 transmission number
10 = 2 transmission numbers.
11 = 3 transmission numbers.

LOADING

See the TRANSMISSION NUMBER REGISTER GROUP

REQUIRED

This register must always be defined for Data Comm activity.

DEFAULT VALUES

System Generate/Load – 00 (No transmission numbers)

RELATED REGISTERS

EVEN/ODD TRANSMISSION NUMBER REGISTER

EVEN/ODD TRANSMISSION NUMBER REGISTER

FUNCTION

This register specifies whether or not the transmission number is of Even/Odd (alternating 0 and 1) format, or of Decimal (0 to 9) format. If Decimal, the number of transmission numbers is determined by the NUMBER OF TRANSMISSION NUMBERS REGISTER. If Even/Odd (also known as Binary) there is only one transmission number.

FORMAT

The register size is 1 bit and it is part of the TRANSMISSION NUMBER REGISTER GROUP.

POSSIBLE VALUES

0 = NO (Decimal transmission numbers).
1 = YES (Even/odd transmission number).

LOADING

See the TRANSMISSION NUMBER REGISTER GROUP.

REQUIRED

This register must always be specified for Data Comm activity.

DEFAULT VALUES

System Generate/Load – 0 (NO)
Data Comm System Register Generator – 0 (NO).

RELATED REGISTERS

NUMBER OF TRANSMISSION NUMBERS REGISTER

SYNCHRONOUS COMPARE CHARACTER REGISTER

FUNCTION

This character defines the actual SYNC character which is compared in synchronous Data Comm line procedures.

FORMAT

The register size is 8 bits and its format is the column and row numbers of the desired character together with any parity setting that may be required.

POSSIBLE VALUES

Any valid character as agreed by all the terminals on the line.

LOADING

This register must be loaded with the desired value before a synchronous line procedure is initiated.
The register may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions (SL5 only).

REQUIRED

This register must always be defined for synchronous Data Comm activity.

DEFAULT VALUES

System Generate/Load – 00 (NUL).

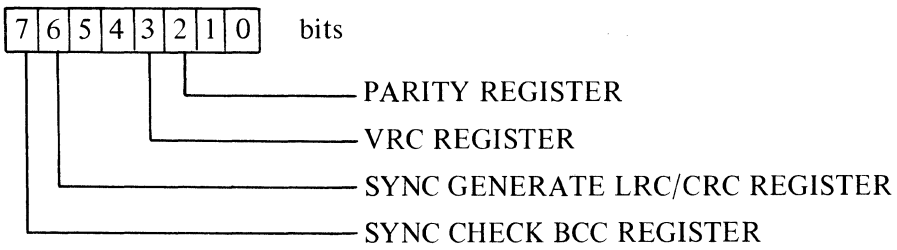
TRANSMIT/RECEIVE FLAGS REGISTER GROUP

FUNCTION

The register group size is 8 bits and it contains the following registers:

FORMAT

The register group size is 8 bits and it contains the following registers:



LOADING

The appropriate set of this register group must be loaded before the line procedure is initiated.
The registers may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions (SL5 only).

REQUIRED

This register group must always be defined for Data Comm activity.

PARITY REGISTER

FUNCTION

This register specifies whether the parity of the character is to be even or odd. This register is only active if the VRC REGISTER is set to 0. In an asynchronous environment, this register is only active if the SOFT SELECT parity option is selected on the ADC board.

FORMAT

The register size is 1 bit and it is part of the TRANSMIT/RECEIVE FLAGS REGISTER GROUP.

POSSIBLE VALUES

0 = Odd Parity
1 = Even Parity

LOADING

See the TRANSMIT/RECEIVE FLAGS REGISTER GROUP.

REQUIRED

This register must always be specified for Data Comm activity.

DEFAULT VALUES

System Generate/Load – 0 (Odd Parity)

RELATED REGISTERS

VRC REGISTER

VRC REGISTER

FUNCTION

This register specifies whether or not the character parity (Vertical Redundancy Check) bit is to be checked or generated during receipt or transmission of characters.

FORMAT

The register size is 1 bit and it is part of the TRANSMIT/RECEIVE FLAGS REGISTER GROUP.

POSSIBLE VALUES

0 = YES (Generate or Check Parity)
1 = NO

LOADING

See the TRANSMIT/RECEIVE FLAGS REGISTER GROUP

REQUIRED

This register must always be specified for Data Comm activity.

DEFAULT VALUES

System Generate/Load – 0 (YES)

RELATED REGISTERS

PARITY REGISTER

SYNC GENERATE LRC/CRC REGISTER

FUNCTION

This register specifies whether Longitudinal (LRC) or Cyclical (CRC) Redundancy Check characters are to be generated during transmission of data.

FORMAT

The register size is 1 bit and it is part of the TRANSMIT/RECEIVE FLAGS REGISTER GROUP.

POSSIBLE VALUES

0 = Generate LRC
1 = Generate CRC
} The actual values are dependant on the line procedure.

LOADING

See the TRANSMIT/RECEIVE FLAGS REGISTER GROUP

REQUIRED

This register must always be specified for synchronous Data Comm activity.

DEFAULT VALUES

System Generate/Load – 0 (Generate LRC)
Data Comm System Register Generator – 0 (Generate LRC).

SYNC CHECK BCC REGISTER

FUNCTION

This register specifies whether or not Block Check Character (LRC or CRC) checking is to be performed on received data.

FORMAT

The register size is 1 bit and it is part of the TRANSMIT/RECEIVE FLAGS REGISTER GROUP.

POSSIBLE VALUES

0 = YES (Check BCC)
1 = NO.

LOADING

See the TRANSMIT/RECEIVE FLAGS REGISTER GROUP

REQUIRED

This register must always be specified for synchronous Data Comm activity.

DEFAULT VALUES

System Generate/Load – 0 (YES).

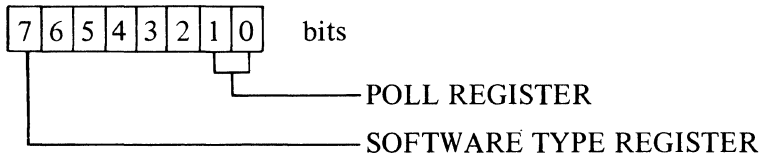
D/C OPERATION MODE REGISTER GROUP

FUNCTION

This register group specifies certain characteristics of the Data Comm operating environment.

FORMAT

The register group size is 8 bits and it contains the following registers:



LOADING

This register group must be loaded before the line procedure is initiated.
The registers may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions (SL5 only).

REQUIRED

This register group must always be specified for Data Comm activity.

POLL REGISTER

FUNCTION

This register specifies whether the terminal will be polled in a standard mode or will be polled in a group poll mode.

FORMAT

The register size is 2 bits and it is part of the D/C OPERATION MODE REGISTER GROUP.

POSSIBLE VALUES

The actual value is dependant on the line procedure.

LOADING

See the D/C OPERATION MODE REGISTER GROUP.

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUE

System Generate/Load – 0 (Standard Poll).

SOFTWARE TYPE REGISTER

FUNCTION

This register specifies the Virtual Machine language of the S-program being executed.

FORMAT

The register size is 1 bit and it is part of the D/C OPERATION MODE REGISTER GROUP

POSSIBLE VALUES

0 = SL5
1 = SL3

LOADING

See the D/C OPERATION MODE REGISTER GROUP.

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUES

System Generate/Load – 0 (SL5).

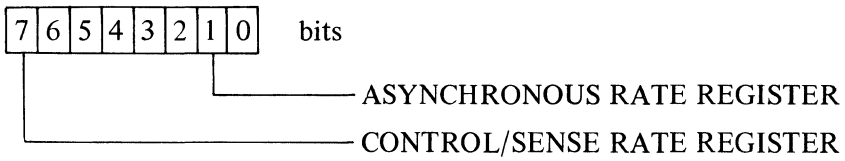
TRANSMISSION RATE REGISTER GROUP

FUNCTION

This register group defines the speed of transmission of data along the data comm network.

FORMAT

The register group size is 8 bits and it contains the following registers:



LOADING

This register group must be loaded before the line procedure is initiated.
The registers may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file)
c) S-program instructions (SL5 only).

REQUIRED

This register must always be defined for Data Comm activity and its use is dependant on the line procedure.

ASYNCHRONOUS RATE REGISTER

FUNCTION

This register specifies whether the transmission rate is to be High (normal) or Low (fall-back). This register is only active if the CONTROL/SENSE RATE REGISTER is set to 1 (Control).

FORMAT

The register size is 1 bit and it is part of the TRANSMISSION RATE REGISTER GROUP.

POSSIBLE VALUES

0 = High (normal) rate
1 = Low (fall-back) rate

LOADING

See the TRANSMISSION RATE REGISTER GROUP.

REQUIRED

The use of this register is line procedure dependant.

DEFAULT VALUES

System Generate/Load – 0 (High Rate).

RELATED REGISTERS

CONTROL/SENSE RATE REGISTER.

CONTROL/SENSE RATE REGISTER

FUNCTION

This register specifies whether the system is to transmit and receive data at the rate specified in the ASYNCHRONOUS RATE REGISTER or at the rate specified by the signal "RATE" coming from the downstream terminal, i.e. further away from the data set (modem).

FORMAT

The register size is 1 bit and it is part of the TRANSMISSION RATE REGISTER GROUP.

POSSIBLE VALUES

0 = SENSE Rate from Downstream.
1 = CONTROL Rate.

LOADING

See the TRANSMISSION RATE REGISTER GROUP.

REQUIRED

The use of this register is line procedure dependant. Generally, the terminal furthest away from the data set (downstream) in a concatenated string should be set to control transmission/reception rate. Terminals closer to the data set (upstream) should be set to sense the downstream rate. If there is only one terminal, it should be set to control rate.

DEFAULT VALUES

System Generate/Load – 0 (SENSE Rate).

RELATED REGISTERS

ASYNCHRONOUS RATE REGISTER.

FOUR WIRE RECEIVE DELAY REGISTER

FUNCTION

This register defines a programmatic delay from the time that "REQUEST TO SEND" goes FALSE (OFF), until the line procedure will look at the "RECEIVED DATA" line. This delay is also known as the "Four Wire Transmit to Receive Delay". This register is only active when the TWO/FOUR WIRE REGISTER is set to 0 (4 Wire).

FORMAT

The register size is 8 bits and it contains the number of 1.75 millisecond increments required for the delay, expressed in a hexadecimal format.

POSSIBLE VALUES

Hex 00 to 90 (0 to 144 increments of 1.75 ms = 0 to 252 ms)

LOADING

This register must be loaded before the line procedure is initiated.
It may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions (SL5 only).

REQUIRED

This register must be specified whenever Data Comm activity is to take place in a four wire mode.

DEFAULT VALUES

System Generate/Load – 1 (1.75 ms).

RELATED REGISTERS

TWO/FOUR WIRE REGISTER.

FOUR WIRE TRANSMIT DELAY REGISTER

FUNCTION

This register defines a programmatic delay from the time that "REQUEST TO SEND" becomes TRUE (ON) until the time that the line procedure will look at "CLEAR TO SEND" (CTS) from the data set. When CTS becomes TRUE (ON), the line procedure will commence transmission. This register is only active when the TWO/FOUR WIRE REGISTER is set to 0 (4 Wire) AND the CLEAR TO SEND REGISTER is set to 1.

FORMAT

The register size is 8 bits and it contains the number of 1.75 millisecond increments required for the delay, expressed in a hexadecimal format.

POSSIBLE VALUES

Hex 00 to 90 (0 to 144 increments of 1.75 ms = 0 to 252 ms)

LOADING

This register must be loaded before the line procedure is initiated.
It may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions (SL5 only).

REQUIRED

This register must be specified whenever Data Comm activity is to take place in a four wire mode.

DEFAULT VALUES

System Generate/Load – 1 (1.75 ms).

RELATED REGISTERS

TWO/FOUR WIRE REGISTER.
CLEAR TO SEND REGISTER.

PROCEDURE AREA 2

FUNCTION

This register specifies, to the line procedure, any extra variable information that may be necessary to properly execute that line procedure.

FORMAT

The register size is 8 bits.

POSSIBLE VALUES

The values are dependant on the line procedure.

LOADING

The time at which the register is loaded is dependant on the line procedure.
The register may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions (SL5 only).

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUES

System Generate/Load – 0.

RING LIMIT REGISTER

FUNCTION

This register specifies the length of time that an Auto Dial line procedure will wait for an answer before disconnecting a line on which it is attempting to call another terminal.

FORMAT

The register size is 8 bits and it contains the number of 6 second increments required for the ring time, expressed in a hexadecimal format.

POSSIBLE VALUES

Hex 00 to FF (0 to 1536 seconds)

LOADING

This register must be loaded before the line procedure is initiated.
It may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions (SL5 only).

REQUIRED

This register must be specified whenever an Auto Dial line procedure is to be used.

DEFAULT VALUES

System Generate/Load – 07 (42 seconds).

TWO WIRE RECEIVE DELAY REGISTER

FUNCTION

This register defines a programmatic delay from the time that "REQUEST TO SEND" goes FALSE (OFF), until the line procedure will look at the "RECEIVED DATA" line. This delay is also known as the "Two Wire Transmit to Receive Delay". This register is only active when the TWO/FOUR WIRE REGISTER is set to 1 (2 wire).

FORMAT

The register size is 8 bits and it contains the number of 1.75 millisecond increments required for the delay, expressed in a hexadecimal format.

POSSIBLE VALUES

Hex 00 to 90 (0 to 144 increments of 1.75 ms = 0 to 252 ms)

LOADING

This register must be loaded before the line procedure is initiated.
It may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions (SL5) only).

REQUIRED

This register must be specified whenever Data Comm activity is to take place in a two wire mode.

DEFAULT VALUES

System Generate/Load – 1 (1.75 ms).

RELATED REGISTERS

TWO/FOUR WIRE REGISTER.

TWO WIRE TRANSMIT DELAY REGISTER

FUNCTION

This register defines a programmatic delay from the time that "REQUEST TO SEND" becomes TRUE (ON) until the time that the line procedure will look at "CLEAR TO SEND" (CTS) from the data set. When CTS becomes TRUE (ON), the line procedure will commence transmission. This register is only active when the TWO/FOUR WIRE REGISTER is set to 1 (2 Wire) AND the CLEAR TO SEND REGISTER is set to 1.

FORMAT

The register size is 8 bits and it contains the number of 1.75 millisecond increments required for the delay, expressed in a hexadecimal format.

POSSIBLE VALUES

Hex 00 to 90 (0 to 144 increments of 1.75 ms = 0 to 252 ms)

LOADING

This register must be loaded before the line procedure is initiated.
It may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions (SL5 only).

REQUIRED

This register must be specified whenever Data Comm activity is to take place in a two wire mode.

DEFAULT VALUES

System Generate/Load – 1 (1.75 ms).

RELATED REGISTERS

TWO/FOUR WIRE REGISTER.
CLEAR TO SEND REGISTER.

D/C SEND BUFFER ADDRESS REGISTER

FUNCTION

This register defines the absolute base memory address of the Data Comm Send Buffer. This is the buffer from which the line procedure takes the text data to be transmitted.

FORMAT

The register size is 16 bits and it contains the address as a hexadecimal byte location.

POSSIBLE VALUES

Any valid memory address.

LOADING

This register must be loaded before transmission of data. For SL3 S-programs the value is automatically generated at System Generation.

For SL5 S-programs the value may be loaded using S-program instructions (This value is the address relative to the USER MEMORY BASE ADDRESS, and is converted to absolute by the interpreter).

REQUIRED

This register must be defined for all Data Comm transmission.

DEFAULT VALUES

System Generate/Load – if SL3 buffers specified – absolute location of buffers. otherwise – zero.
Program Start (SL3 only) – absolute location of buffers.

D/C RECEIVE BUFFER ADDRESS REGISTER

FUNCTION

This register defines the absolute base memory address of the Data Comm Receive Buffer. This is the buffer into which the line procedure inserts text data that has been received.

FORMAT

The register size is 16 bits and it contains the address as a hexadecimal byte location.

POSSIBLE VALUES

Any valid memory address.

LOADING

This register must be loaded before reception of data.
For SL3 S-programs the value is automatically generated at System Generation.
For SL5 S-programs the value may be loaded using S-program instructions (This value is the address relative to the USER MEMORY BASE ADDRESS and is converted to absolute by the interpreter).

REQUIRED

This register must be defined for all Data Comm reception.

DEFAULT VALUES

System Generate/Load – if SL3 buffers specified – absolute location of buffers. otherwise – zero.
Program Start (SL3 only) – absolute location of buffers.

TRANSMIT MACHINE ADDRESS REGISTER

FUNCTION

This register specifies the terminal address that is transmitted in the header portion of messages.

FORMAT

The register size is 64 bits and it is split into eight address characters. Each 8 bit character contains the column and row numbers that represent the ASCII character to be transmitted.

POSSIBLE VALUES

The permissible values are dependant on the line procedure.

LOADING

This register must be loaded with the desired values before the line procedure transmits the terminal address. The precise time of loading is dependant on the line procedure.
The register may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions.

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUES

System Generate/Load – NUL (00) codes.

RECEIVE MACHINE ADDRESS REGISTER

FUNCTION

This register specifies the terminal address that is used to compare with the address in the header portion of the received messages.

FORMAT

The register size is 64 bits and it is split into eight address characters. Each 8 bit character contains the column and row numbers that represent the ASCII character to be compared to the received address character.

POSSIBLE VALUES

The permissible values are dependant on the line procedure.

LOADING

This register must be loaded with the desired values before the line procedure references it for comparison. The precise time of loading is dependant on the line procedure.
The register may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions.

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUES

System Generate/Load – NUL (00) codes.

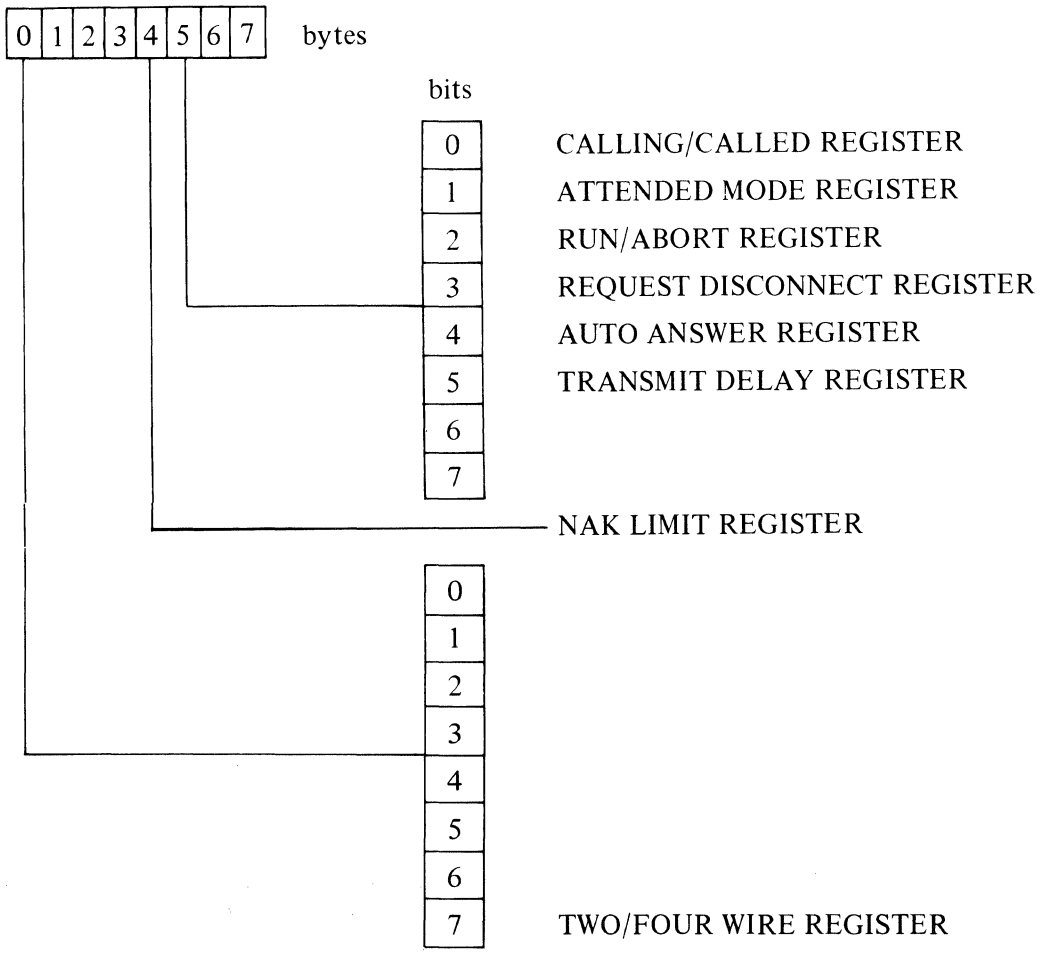
TWO/FOUR WIRE REGISTER GROUP

FUNCTION

This register group defines various characteristics of the Data Comm line and the terminal's operation on that line.

FORMAT

The register group size is 64 bits and it contains the following registers:



LOADING

The precise time at which this register group is loaded is dependant on the line procedure.
The registers may be loaded using S-program instructions.

REQUIRED

The use of this register is dependant on the line procedure.

CALLING/CALLED REGISTER

FUNCTION

This register specifies whether the terminal has the responsibility of initiating the activity on the line (CALLING station) or of responding to the CALLING station (CALLED station).

FORMAT

The register size is 1 bit and it is part of the TWO/FOUR WIRE REGISTER GROUP.

POSSIBLE VALUES

0 = CALLING station
1 = CALLED station.

LOADING

See the TWO/FOUR WIRE REGISTER GROUP

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUES

System Generate/Load – 0 (CALLING)

RELATED REGISTERS

ATTENDED MODE REGISTER
AUTO ANSWER REGISTER

ATTENDED MODE REGISTER

FUNCTION

This register specifies whether the CALLING station is attended (operator present) or unattended (no operator).

FORMAT

The register size is 1 bit and it is part of the TWO/FOUR WIRE REGISTER GROUP.

POSSIBLE VALUES

0 = UNATTENDED
1 = ATTENDED

LOADING

See the TWO/FOUR WIRE REGISTER GROUP.

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUES

System Generate/Load – 0 (UNATTENDED).

RELATED REGISTERS

CALLING/CALLED REGISTER.

RUN/ABORT REGISTER

FUNCTION

This register specifies whether the line procedure should run normally (RUN mode) or if the line should be disconnected (ABORT mode).

FORMAT

The register size is 1 bit and it is part of the TWO/FOUR WIRE REGISTER GROUP.

POSSIBLE VALUES

0 = ABORT mode
1 = RUN mode.

LOADING

See the TWO/FOUR WIRE REGISTER GROUP

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUES

System Generate/Load – 0 (ABORT)

REQUEST DISCONNECT REGISTER

FUNCTION

This register specifies whether or not the line procedure should disconnect the communication line at a point in the procedure. This point is line procedure dependant, but is generally when there are no messages to be transmitted and none are being received.

FORMAT

The register size is 1 bit and it is part of the TWO/FOUR WIRE REGISTER GROUP.

POSSIBLE VALUES

0 = DO NOT DISCONNECT
1 = REQUEST VALID DISCONNECT

LOADING

See the TWO/FOUR WIRE REGISTER GROUP.

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUES

System Generate/Load – 0 (NO DISCONNECT)

AUTO ANSWER REGISTER

FUNCTION

This register specifies whether the CALLED station is to automatically or manually answer an incoming call

FORMAT

The register size is 1 bit and it is part of the TWO/FOUR WIRE REGISTER GROUP.

POSSIBLE VALUES

0 = MANUAL ANSWER
1 = AUTO ANSWER

LOADING

See the TWO/FOUR WIRE REGISTER GROUP.

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUES

System Generate/Load – 0 (MANUAL ANSWER)

RELATED REGISTERS

CALLING/CALLED REGISTER

TRANSMIT DELAY REGISTER

FUNCTION

This register specifies whether or not the line procedure will wait for 10.5 seconds before indicating that the terminal has no data to transmit.

FORMAT

The register size is 1 bit and it is part of the TWO/FOUR WIRE REGISTER GROUP.

POSSIBLE VALUES

0 = DISABLE TRANSMIT DELAY
1 = ENABLE TRANSMIT DELAY

LOADING

See the TWO/FOUR WIRE REGISTER GROUP.

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUES

System Generate/Load – 0 (DISABLE TRANSMIT DELAY).

NAK LIMIT REGISTER

FUNCTION

This register specifies the number of times that a station is willing to receive a NAK reply before attempting to rectify the situation. The precise use of the register is line procedure dependant.

FORMAT

The register size is 8 bits and it is part of the TWO/FOUR WIRE REGISTER GROUP. It contains the value, expressed in hexadecimal format.

POSSIBLE VALUES

Hex 00 to FF (0 to 255 times).

LOADING

See the TWO/FOUR WIRE REGISTER GROUP.

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUES

System Generate/Load – 00

TWO/FOUR WIRE REGISTER

FUNCTION

This register specifies whether the communication line consists of "two" or "four" wires, thus enabling different delays which are specified in the DELAY REGISTERS.

FORMAT

The register size is 1 bit and it is part of the TWO/FOUR WIRE REGISTER GROUP.

POSSIBLE VALUES

0 = 4 WIRE
1 = 2 WIRE

LOADING

This register must be loaded before the line procedure is initiated.
See the TWO/FOUR WIRE REGISTER GROUP.

REQUIRED

This register is always required for Data Comm activity.

DEFAULT VALUES

System Generate/Load – 0 (4 Wire).

RELATED REGISTERS

CLEAR TO SEND REGISTER
FOUR WIRE RECEIVE DELAY REGISTER
FOUR WIRE TRANSMIT DELAY REGISTER
TWO WIRE RECEIVE DELAY REGISTER
TWO WIRE TRANSMIT DELAY REGISTER

SEND TRANSMISSION NUMBER REGISTER

FUNCTION

This register specifies the value of the transmission number which will be transmitted in the header portion of the next message sent. It will be incremented automatically by the line procedure when a positive acknowledgement of the message has been received. The number of transmission numbers is determined by the setting of the NUMBER OF TRANSMISSION NUMBERS REGISTER.

FORMAT

The register size is 64 bits and its format is dependant on the number of transmission numbers. Each number occupies 8 bits and MUST be represented as an ASCII numeric character. The numbers are located in the most significant byte positions of the register.

POSSIBLE VALUES

ASCII 30 to 39 (0 to 9) for each byte of the number. The rest of the register is zero.

LOADING

This register may be loaded at any time prior to transmission of a message.
It may be loaded by using S-program instructions.

REQUIRED

This register must always be specified if transmission numbers are used.

DEFAULT VALUES

System Generate/Load – NUL (00) codes.

RELATED REGISTERS

TRANSMISSION NUMBER REGISTER GROUP.

EXPECTED TRANSMISSION NUMBER REGISTER

FUNCTION

This register specifies the expected value of the transmission number in the header portion of the next message to be received. When the message is received, the line procedure will compare the two numbers. If equal, the register will be incremented. If unequal, the INVALID TRANSMISSION NUMBER flag will be set. The number of transmission numbers is determined by the setting of the NUMBER OF TRANSMISS-ION NUMBERS REGISTER.

FORMAT

The register size is 64 bits and its format is dependant on the number of transmission numbers. Each number occupies 8 bits and MUST be represented as an ASCII numeric character. The numbers are located in the most significant byte positions of the register.

POSSIBLE VALUES

ASCII 30 to 39 (0 to 9) for each byte of the number. The rest of the register is zero.

LOADING

This register may be loaded at any time prior to receipt of a message.
It may be loaded by using S-program instructions.

REQUIRED

This register must always be specified if transmission numbers are used.

DEFAULT VALUES

System Generate/Load – NUL (00) codes.

RELATED REGISTERS

TRANSMISSION NUMBER REGISTER GROUP.

EXPECTED GROUP TRANSMISSION NUMBER REGISTER

FUNCTION

This register specifies the expected value of the transmission number in the header portion of the next group message to be received. The number of transmission numbers is determined by the setting of the NUMBER OF TRANSMISSION NUMBERS REGISTER and the precise use of the register is dependant on the line procedure.

FORMAT

The register size is 64 bits and its format is dependant on the number of transmission numbers. Each number occupies 8 bits and MUST be represented as an ASCII numeric character. The numbers are located in the most significant byte positions of the register.

POSSIBLE VALUES

ASCII 30 to 39 (0 to 9) for each byte of the number. The rest of the register is zero.

LOADING

This register may be loaded at any time prior to receipt of a group message.
It may be loaded by using S-program instructions.

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUES

System Generate/Load – NUL (00) codes.

RELATED REGISTERS

TRANSMISSION NUMBER REGISTER GROUP.

EXPECTED BROADCAST TRANSMISSION NUMBER REGISTER

FUNCTION

This register specifies the expected value of the transmission number in the header portion of the next broadcast message to be received. The number of transmission numbers is determined by the NUMBER OF TRANSMISSION NUMBERS REGISTER and the precise use of the register is dependant on the line procedure.

FORMAT

The register size is 64 bits and its format is dependant on the number of transmission numbers. Each number occupies 8 bits and MUST be represented as an ASCII numeric character. The numbers are located in the most significant byte positions of the register.

POSSIBLE VALUES

ASCII 30 to 39 (0 to 9) for each byte of the number. The rest of the register is zero.

LOADING

This register may be loaded at any time prior to receipt of a broadcast message.
It may be loaded by using S-program instructions.

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUES

System Generate/Load – NUL (00) codes.

RELATED REGISTERS

TRANSMISSION NUMBER REGISTER GROUP.

TRANSMISSION HEADER REGISTER

FUNCTION

This register contains some or all of the header portion of the last message received. The precise use of the register is dependant on the line procedure.

FORMAT

The register size is 64 bits and its format is dependant on the line procedure.

RETRIEVING

This register may be retrieved at any time using an S-program instruction.

DEFAULT VALUES

System Generate/Load – NUL (00) codes.

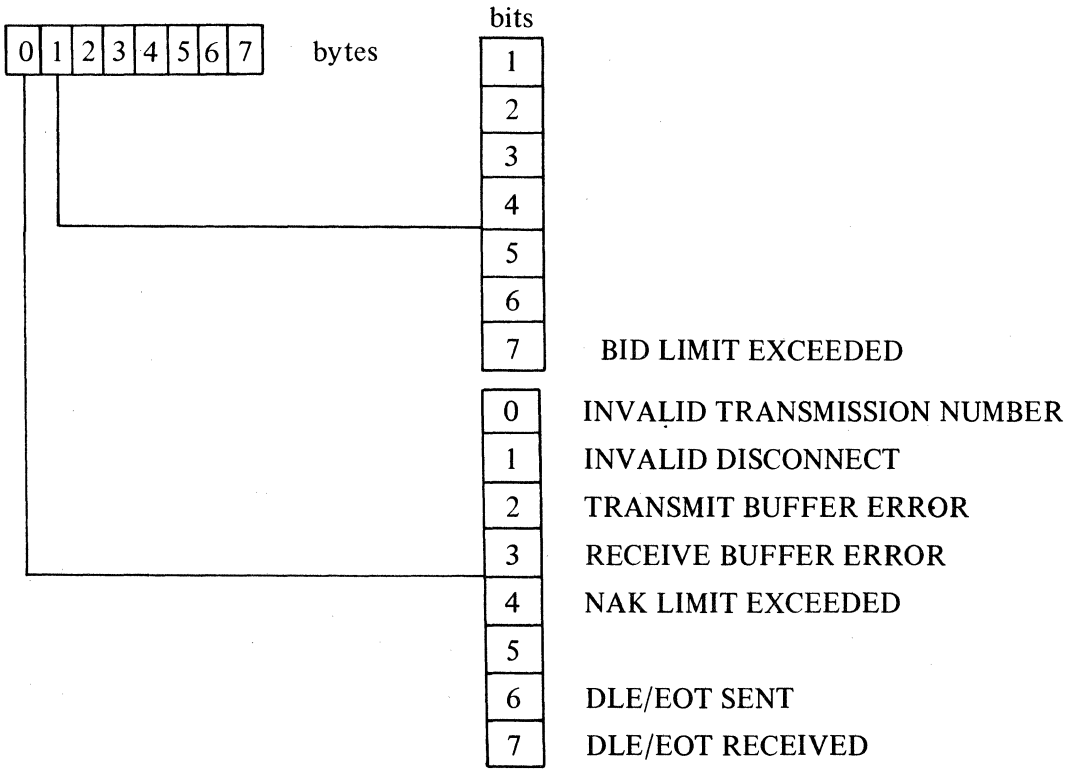
PROBLEM FLAG REGISTER

FUNCTION

This register specifies the precise error condition that exists when a Data Comm Error (D1 flag set) has been flagged. Each error condition is precisely defined under the line procedure with which it can occur.

FORMAT

The register size is 64 bits and it contains the following problem flags:



RETRIEVING/LOADING

This register may be retrieved and loaded at any time using S-program instructions.

REQUIRED

This register should be retrieved and examined immediately after a Data Comm Error has been flagged. The condition should be corrected, the register should then be reset to zero and the Data Comm Error Flag reset. The precise use of the register is dependant on the line procedure.

SPECIAL OPTIONS BYTE

FUNCTION

This register specifies, to the line procedure, any extra options that are available to the S-program.

FORMAT

The register size is 8 bits.

POSSIBLE VALUES

The values are dependant on the line procedure.

LOADING

This register is loaded at Cold Start and is defined at System Generation.

REQUIRED

The use of this register is dependant on the line procedure.

DEFAULT VALUES

System Generate – 0 (Zero)

USER REGISTERS

This section defines each of the User Registers. The registers of this group are accessed directly by the user program to give flexibility in programming. They can only be accessed using S-program instructions.

All word addresses mentioned in this section are considered to be relative to the User Memory Base Address Register except where otherwise stated.

ACCUMULATOR

FUNCTION

This register is used for temporary storage, accumulation and shifting of digits. It serves as an operand in all arithmetic instructions and may be accessed in full or by single digit.

FORMAT

The register size is 64 bits and it is split into 16 digits of 4 bits each. For the purposes of arithmetic, it is treated as having 15 data digits with the most significant digit containing the Accumulator Flags.

DEFAULT VALUES

Program Start – Indeterminate.

RELATED REGISTERS

FLAG REGISTERS.

INDEX REGISTERS

FUNCTION

These registers are used to modify, under S-program control, the parameter fields of instructions. They may also be used as programmatic counters with the ability of setting a TEST FLAG when the count has reached a limit.

FORMAT

There are 4 registers and each is 8 bits in size for SL3 and 16 bits in size for SL5.

POSSIBLE VALUES

SL3 – Hex 00 to FF (0 to 255)
SL5 – Hex 0000 to FFFF (0 to 65536)
When incrementing or decrementing the registers, if the value passes either limit, then it is looped-round i.e. 00 passes to FF and FF passes to 00.

DEFAULT VALUES

Program Start – 0 (Zero)

RELATED REGISTERS

FLAG REGISTERS.

REMAINDER REGISTER

FUNCTION

This register contains the remainder which is left after the execution of a DIVIDE instruction

FORMAT

The register size is 64 bits and it consists of 15 data digits plus a flag digit in the most significant digit position.

DEFAULT VALUES

Program Start – 0 (Zero)

SCALE REGISTER

FUNCTION

This register defines the scale factor for MULTIPLY and DIVIDE instructions. For MULTIPLY, the value of the factor is the number of digits that the product is to be right shifted and for DIVIDE it is the number of implied decimal digits in the quotient.

FORMAT

The register size is 4 bits.

POSSIBLE VALUES

Hex 0 to F (0 to 15).

DEFAULT VALUES

Program Start – 0 (Zero)

RELATED REGISTERS

SCALE REGISTER FLAGS.

PROGRAM SELECT KEY TABLE REGISTER

FUNCTION

This register specifies the absolute base memory address of the Program Select Key Table. The use of an enabled PSK at a keyboard halt will cause execution of the appropriate instruction from the table.

FORMAT

The register size is 16 bits.

POSSIBLE VALUES

The address must be located on a word boundary.

DEFAULT VALUES

Program Start – 0 (Zero)

PROGRAM SELECT KEY ENABLED REGISTER

FUNCTION

This register defines which Program Select keys will be enabled at the next keyboard instruction. After execution of the keyboard instruction, the register will be reset to zero.

FORMAT

The register size is 16 bits and it is split into two registers of 8 bits each, each representing a group of PSKs.

DEFAULT VALUES

Program Start – 0 (Zero)

KEYBOARD BASE REGISTER

FUNCTION

This register stores the destination base address for succeeding instructions which require a memory address not provided by the instruction. Addressing is to the base address, except in an SL3 data handling or SL5 environment in which case addressing is to character locations as specified by the SEND CHARACTER POINTER REGISTER.

FORMAT

The register size is 16 bits.

POSSIBLE VALUES

The base address must be located on a word boundary. If the value is zero, the register will address the DATA COMM SEND BUFFER.

DEFAULT VALUES

Program Start – 0 (referencing Data Comm buffers).

RELATED REGISTERS

SEND CHARACTER POINTER REGISTER.

SEND CHARACTER POINTER REGISTER

FUNCTION

This register is used as a modifier to the KEYBOARD BASE REGISTER in situations where character addressing is required.

FORMAT

The register size is 16 bits in SL5 and 8 bits in SL3.

POSSIBLE VALUES

SL3 – Hex 00 to FF (0 to 255)
SL5 – Hex 0000 to FFFF (0 to 65535).

DEFAULT VALUES

Program Start – 0 (Zero)

RELATED REGISTERS

KEYBOARD BASE REGISTER

RECEIVE BUFFER REGISTER

FUNCTION

This register stores the source base address for succeeding instructions which require a memory address not provided by the instruction. Addressing is to character locations as specified by the RECEIVE CHARACTER POINTER REGISTER. This register is only active if the Data Handling 1 segment is resident.

FORMAT

The register size is 16 bits.

POSSIBLE VALUES

The base address must be located on a word boundary. If the value is zero, the register will address the DATA COMM RECEIVE BUFFER.

DEFAULT VALUES

Program Start – 0 (referencing Data Comm buffers).

RELATED REGISTERS

RECEIVE CHARACTER POINTER REGISTER

RECEIVE CHARACTER POINTER REGISTER

FUNCTION

This register is used as a modifier to the RECEIVE BUFFER REGISTER in situations where character addressing is required.

FORMAT

The register size is 16 bits in SL5 and 8 bits in SL3.

POSSIBLE VALUES

SL3 – Hex 00 to FF (0 to 255)
SL5 – Hex 0000 to FFFF (0 to 65535)

DEFAULT VALUES

Program Start – 0 (Zero)

RELATED REGISTERS

RECEIVE BUFFER REGISTER

PRINT NUMERIC MASK REGISTER

FUNCTION

This register specifies the base address of the Print Mask Table. It is also used to specify the base address of the Check Digit Compute/Verify Table.

FORMAT

The register size is 16 bits.

POSSIBLE VALUES

The base address must be located on a word boundary.

DEFAULT VALUES

Program Start – Indeterminate.

POSITION REGISTER

FUNCTION

This register specifies the position to which the print head is to be moved before executing a printer or keyboard instruction.

FORMAT

The register size is 8 bits.

POSSIBLE VALUES

Hex 00 to 95 (1 to 150)

DEFAULT VALUES

Program Start – 00 (Position 1).

LEFT FORMS COUNT REGISTER

FUNCTION

This register keeps a count of each advance of the Forms Handler, whether programmatic (LEFT advance) or resulting from a keyboard operation. When the register equals the LEFT FORMS LIMIT REGISTER then the next such advance will cause the count to return to 1 and the TEST LIMIT FLAG to be set.

FORMAT

The register size is 8 bits.

POSSIBLE VALUES

Hex 00 to FF (0 to 255)

DEFAULT VALUES

Program Start – 00

RELATED REGISTERS

LEFT FORMS LIMIT REGISTER
FLAG REGISTERS

LEFT FORMS LIMIT REGISTER

FUNCTION

This register sets up a limit for the LEFT FORMS COUNT REGISTER. It is generally used to represent the number of lines per page on a continuous form.

FORMAT

The register size is 8 bits.

POSSIBLE VALUES

Hex 00 to FF (0 to 255)

DEFAULT VALUES

Program Start – FF (255).

RELATED REGISTERS

LEFT FORMS COUNT REGISTER

RIGHT FORMS COUNT REGISTER

FUNCTION

Since there is only one Forms Handler on this system, this register is only used to keep a count of each programmatic RIGHT advance. (This advance has no effect on the Forms Handler). When the register equals the RIGHT FORMS LIMIT REGISTER then the next such advance will cause the count to return to 1 and the TEST LIMIT FLAG to be set.

FORMAT

The register size is 8 bits.

POSSIBLE VALUES

Hex 00 to FF (0 to 255).

DEFAULT VALUES

Program Start – 00

RELATED REGISTERS

RIGHT FORMS LIMIT REGISTER
FLAG REGISTERS

RIGHT FORMS LIMIT REGISTER

FUNCTION

This register sets up a limit for the RIGHT FORMS COUNT REGISTER. It is generally used to represent the number of lines per page on a continuous form.

FORMAT

The register size is 8 bits.

POSSIBLE VALUES

Hex 00 to FF (0 to 255)

DEFAULT VALUES

Program Start – FF (255).

RELATED REGISTERS

RIGHT FORMS COUNT REGISTER

SUBROUTINE STACK

FUNCTION

This register stores the return addresses for use with subroutine jump and return instructions. The register can store eight return addresses for SL5 and four return addresses for SL3 but, the subroutine return instruction can only select return levels 1 to 4.

FORMAT

The register size is 128 bits for SL5 and 64 bits for SL3. It is split into addresses of 16 bits each.

DEFAULT VALUES

Program Start – Indeterminate.

FLAG REGISTERS

FUNCTION

These are a set of registers which are used to indicate conditions. Some registers may be set or reset by the system to indicate the occurrence of certain conditions. Others are general purpose and may be used as required. The function of each flag group is defined below.

FORMAT

Each flag group occupies 4 bits.

RETRIEVING/LOADING

Each flag may be accessed at any time using S-program instructions.

NOTE

In all of the following flag descriptions, the flags are defined in ascending bit order and indicate the condition that is true when the bit is ON. The diagram below shows the values assigned to each flag of each group.

Flag Group		V	W	G	A	S	D	K	T	F	E	Y	X	B	L	P	R
Digit Posn.		F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
Bit Posn.	Bit Value	FLAGS															
0	1	*	*	*	–	*	4	4	0	*	4	4	4	4	4	4	4
1	2	*	*	*	S	*	1	1	L	*	1	1	1	1	1	1	1
2	4	*	*	*	C	*	2	2	I	*	2	2	2	2	2	2	2
3	8	*	*	*	M	*	3	3	H	*	3	3	3	3	3	3	3

The values of these flags can be displayed as part of the TRACE print-out.

* – These flags are not currently implemented on this system and must not be referenced.

ACCUMULATOR FLAGS (Group A)

These flags may be set or reset by numeric keyboard, arithmetic, Check Digit Verify or flag instructions. The flags are defined below.

–	Sign
S	Special Flag (various uses)
C	Per hundred
M	Per thousand

The Default Value at Program Start is Indeterminate.

KEYBOARD BUFFER FLAGS (Group B)

These flags are used by the interpreter to indicate the state of the keyboard buffer. The flags are defined below.

4	Soft Keyboard (SL5 only)
1	General Purpose
2	Keyboard Buffer Full (SL3)
	General Purpose (SL5)
3	Keyboard Buffer Empty.

The Default Values at Program Start are:

4	0 (Reset)
1	0 (Reset)
2	0 (Reset)
3	1 (Set)

DATA COMMUNICATIONS FLAGS (Group D)

These flags are used to indicate the state of activity in the line procedure.

The flags are:

4	Reserved		
1	Data Comm Error	–	When this is set, the S-program must interrogate the PROBLEM FLAG REGISTER to determine the exact nature of the error.
2	Message Received	–	This is set by the line procedure to indicate that a message has been properly received. It must be reset by the S-program when the contents of the Receive Buffer are no longer required.
3	Transmit Ready	–	This is set by the S-program to indicate that a message in the Transmit Buffer is ready for transmission. When the message has been successfully transmitted, the line procedure will reset the flag.

The Default Values at Program Start are Indeterminate, and at line procedure initiation are 0 (zero).

If the Data Communications Base Segment is not resident in the interpreter, the D flags (1–4) may be used as General Purpose Flags and set or reset as the programmer desires.

INPUT/OUTPUT EXCEPTION FLAGS (Group E) (SL5 only)

These flags are used to indicate exceptions that occur in Input/Output instructions. The flags are reset at the start of each I/O instruction.

The flags are 4, 1, 2 and 3.

For the exact meaning of every combination of flags, refer to Appendix F.

The Default Value at Program Start is Reset.

OPERATOR CONTROL KEY FLAGS (Group K)

These flags are used to indicate which Operator Control Key (OCK) was depressed to terminate a keyboard instruction. At the start of each keyboard instruction these flags are reset. In between keyboard instructions the flags may be used as general purpose flags. These flags can also be set by the interpreter as a result of encountering certain ASCII control codes during data handling instructions. These flag settings can be found in Appendix F.

The flags are:

4	OCK	4
1	OCK	1
2	OCK	2
3	OCK	3

The Default Value at Program Start is Reset.

SCALE REGISTER FLAGS (Group L)

These flags are a representation of the SCALE REGISTER with which they have bit correspondance. Also, for SL3 only, they are used to indicate exceptions that occur in Input/Output instructions and the flags are reset at the start of each such instruction.

The flags are 4, 1, 2 and 3.

For the exact meaning of every I/O combination of flags, refer to Appendix F.

The Default Value at Program Start is Reset.

GENERAL PURPOSE FLAGS (Group P)

These flags can be used for any purpose desired by the programmer. In addition, they turn on certain of the D-indicator lights.

The flags are:

4	turns on	D5
1	"	D6
2	"	D7
3	"	D8

The Default Value at Program Start is Reset.

GENERAL PURPOSE FLAGS (Group R)

These flags can be used for any purpose desired by the programmer. In addition, they turn on certain of the D-indicator lights.

The flags are:

4	Does not affect any D lights.	
1	turns on	D2
2	"	D3
3	"	D4

The Default Value at Program Start is Reset.

TEST FLAGS (Group T)

These flags are used to indicate exception conditions that can occur during the execution of S-programs.

The flags are:

O Overflow – This can be set under several circumstances:
  a) During shift operations when significant data digits are shifted off left.
  b) In arithmetic instructions when any resultant values overflow the word boundary.
  c) In data handling instructions when an ETX (03) code is encountered.

L Forms Limit – This is set whenever any FORMS COUNT REGISTER exceeds the relevant FORMS LIMIT REGISTER.

I Index Register – This is set whenever the incrementing or decrementing of an Index Register causes the value either to step over the desired limit or, to step over the lower or upper bounds of the register.

U Reserved.

The Default Value at Program Start is Indeterminate.

GENERAL PURPOSE FLAGS (Group X)

These flags are used for any purpose desired by the programmer.

The flags are 4, 1, 2 and 3.

The default Value at Program Start is Reset.

GENERAL PURPOSE FLAGS (Group Y)

These flags are used for any purpose desired by the programmer. In addition, they can also be set by the interpreter as a result of encountering certain ASCII control codes during data handling instructions. These flag settings can be found in Appendix F.

The flags are 4, 1, 2 and 3.

The Default Value at Program Start is Reset.

SECTION 8
DATA COMMUNICATIONS LINE PROCEDURES

This section describes the characteristics of the various line procedures that are available with this system. It also defines the interface between the S-program and the line procedure.

Each line procedure has listed every Data Comm System Register that is relevant to that procedure and how it varies from the "standard" implementation as defined in Section 7.

CONTROL CHARACTERS

This section makes many references to various control characters. All such characters are here defined for ease of reference.

ACK – ACKNOWLEDGEMENT

An affirmative response to either a normal selection (indicating "ready to receive") or a transmission (indicating "message accepted").

AD1, AD2 – ADDRESS CHARACTERS 1 AND 2

A two-character address, established as the address of a station. Used to address a station in polling or selection, in the message header to identify the station from whom the message is transmitted and for group addressing or broadcast, to identify the terminal that is to acknowledge receipt of the message.

BCC – BLOCK CHECK CHARACTER

A redundant character added to the end of a transmission block for the purposes of error detection and control. BCC immediately follows ETX in a transmission block.

BSL – BROADCAST SELECT

A character used to indicate "This is a broadcast message" to all stations. In the broadcast sequence AD1, AD2 identifies the station which will acknowledge receipt of the message. BSL is represented by the character "t" (7, 4).

CON – CONTENTION

A character used to instruct all terminals which receive the instruction to go to contention mode. CON is represented by the character "BEL" (0, 7).

DEOT – MANDATORY DISCONNECT

A communication control sequence consisting of DLE followed by EOT is used to signal that a disconnect of a switched circuit must be initiated.

DLE – DATA LINK ESCAPE

Used to change the meaning of a limited number of contiguously following characters which then become supplementary data transmission control functions.

ENQ – ENQUIRY

Control character which requests a reply. When used in establishment of communication in (a) Point-to-Point procedures, the ENQ is used to bid for master status; (b) Polling procedures, the ENQ is used as the final character of a poll or normal selection. During message transfer in Point-to-Point procedures the ENQ, when used, is a request for a repeat of the previous response.

EOT – END OF TRANSMISSION

Used to indicate the conclusion of a communication sequence. Receipt of EOT will set the station in a control state listening for a communication sequence. EOT is transmitted by a terminal as a "no traffic" response to a poll.

ETX – END OF TEXT

Used to indicate the end of a stream of characters identified as the Text of the transmission.

FSL – FAST SELECT

A character used to indicate "This is a fast select", in a selection sequence. FSL is represented by the character "s" (7, 3).

GSL – GROUP SELECT

A character used to indicate "This is a message for a group of stations". In the group select sequence, AD1, AD2 identifies the station which will acknowledge receipt of the message. GSL may be represented by any agreed on character selected from Column 2 through 6.

NAK – NEGATIVE ACKNOWLEDGEMENT

A negative response to either a selection (indicating not ready to receive) or a transmission (indicating a character parity or BCC failure).

POL – POLL

A character used to indicate "This is a poll" in a polling sequence. POL is represented by the character "p" (7, 0).

RVI – REVERSE INTERRUPT (DLE <)

Reverse Interrupt is used to request premature termination of message transmission. RVI is sent from a slave station to a master station in place of a positive acknowledgement (ACK).

SEL – SELECT

A character used to indicate "This is a normal select" in a selection sequence. SEL is represented by the character "q" (7, 1).

SOH – START OF HEADER

This is the first of a sequence of characters which form the header to a transmission block. The header is ended by STX.

STX – START OF TEXT

Precedes the sequence of characters which the Text of the transmission.

TR# – TRANSMISSION NUMBER

A number identifying in sequence, transmissions from or to a station. Each character of TR# may be represented by numerics 0–9. The number of characters in TR# can be from 0 to 3.

ASYNCHRONOUS POINT TO POINT (SWITCHED BATCH) LINE PROCEDURE

The Identification of this procedure is 02.

The Point-to-Point Switched Batch line procedure allows the system to communicate on an equal basis with another Data Communications Unit in a batch processing environment using a switched line (telephone facilities) connection. The implementation of this procedure allows for an efficient means of establishing control of the line without the overhead related to other procedures.

The Point-to-Point Switched Batch line procedure defines two basic environments: the CALLING station and the CALLED station. The CALLING station has the responsibility of initiating the phone call, over the switched network, to establish a communication link and the CALLED station has the responsibility of answering the phone call. These two terms will be used throughout the description of the procedure to help clarify actions, reactions and responsibilities.

Within the CALLING or CALLED environment, the systems may be in one of the following states:

BID STATE	– This phase of the procedure is used to establish a connection between the two communicating stations.
RECEIVE TEXT STATE	– This phase of the procedure is assigned to the station that has the responsibility of receiving the transmitted data.
TRANSMIT TEXT STATE	– This phase of the procedure is assigned to the station that has the responsibility of transmitting the data.

Figure 8–1 is a simplified flowchart of the Point-to-Point Switched Batch line procedure indicating the CALLING and CALLED stations related to the three states of the procedure. The Chart does not reflect all of the options and exceptions of the procedure; these features are documented and discussed in figure 8–2.

As indicated in figure 8–1, the states of the procedure may be entered in the following manner:

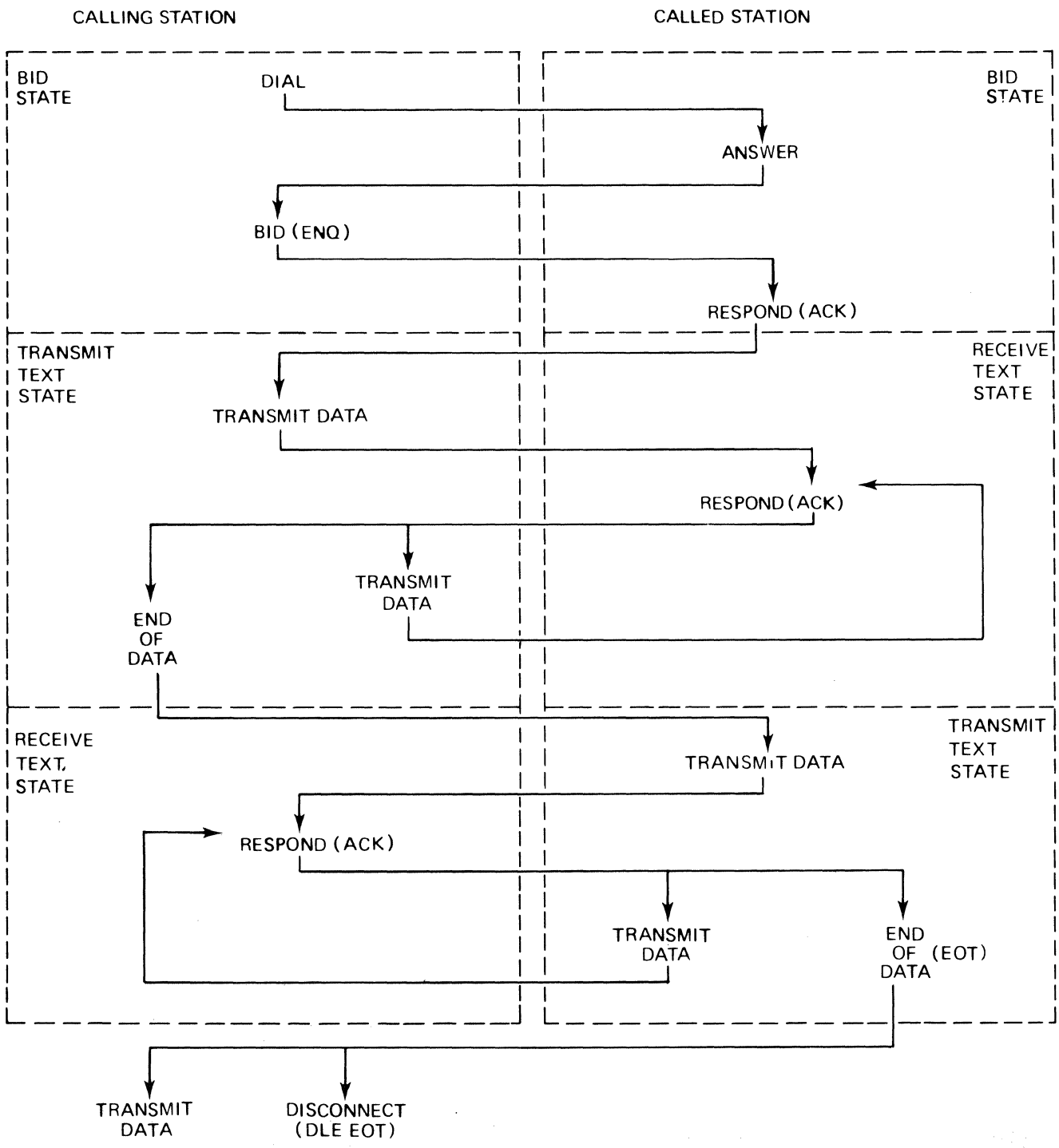


Figure 8–1. Bid, Receive and Transmit States Related to Calling/Called Station.

The BID STATE may be entered by initiation from the S-program. It may also be entered from the RECEIVE TEXT STATE or TRANSMIT TEXT STATE after a disconnect.

The RECEIVE TEXT STATE may be entered from the BID STATE for the CALLED station after a connection is made. The RECEIVE TEXT STATE may also be entered from the TRANSMIT TEXT STATE after all data has been transmitted.

The TRANSMIT TEXT STATE is entered from the BID STATE for the CALLING station after a connection is made. The TRANSMIT TEXT STATE may be entered from the RECEIVE TEXT STATE when the station receives an EOT (end of data) and has data to transmit.

BID STATE

The line procedure is in the BID STATE when the DATA COMM ERROR flag (D1) is set and the PROBLEM FLAG REGISTER is equal to zero.

The following information relates the actions that the procedure will take and how the S-program can interrogate these conditions.

CALLING STATION	CALLED STATION
The S-program must set the appropriate SYSTEM REGISTERS and the D1 flag.	The S-program must set the appropriate SYSTEM REGISTERS and the D1 flag.
Manual dial of data set phone to make connection, if not already made by Auto Dial techniques.	Manual or Auto Answer of the phone call depending on the setting of the AUTO ANSWER REGISTER.
Transmits ENQs up to BID LIMIT.	Transmits ACK if ENQ is received.
If an ACK is received, the procedure resets the D1 flag and enters the TRANSMIT TEXT STATE.	If the procedure transmits an ACK, it resets the D1 flag and enters the RECEIVE TEXT STATE.
If an invalid or no response is received to all the bids, the procedure will disconnect and set the DLE/EOT SENT and BID LIMIT EXCEEDED flags.	The procedure will look for ENQ characters for up to 30 seconds and, if none are recognised, will disconnect and set the DLE/EOT SENT flag.
If the S-program wishes to discontinue the BID STATE, it must RESET the D1 flag. The procedure will then disconnect and set the DLE/EOT SENT and D1 flags.	

Idle Loop

Whenever a disconnect takes place, the line procedure goes to an idle loop. It is the responsibility of the S-program to CLEAR the PROBLEM FLAG REGISTER and RESET the D1 flag before a new BID sequence is initiated. After this, a new BID sequence can be initiated by the S-program by setting the D1 flag, provided that the PROGRAM FLAG REGISTER is RESET.

TRANSMIT TEXT STATE

Upon entering the TRANSMIT TEXT STATE, if the TRANSMIT READY (D3) flag is reset, indicating that there are no messages to be sent, then an EOT will be transmitted and the procedure will enter the RECEIVE TEXT STATE.

If D3 is set, the message in the transmit buffer will be sent and the procedure will enter a Receive Response State (within the TRANSMIT TEXT STATE), in which it will be looking for either an ACK, NAK, RVI, DLE/EOT or an invalid or no response for 1 second. The following notes describe subsequent procedure action.

CALLING STATION

If the NAK LIMIT is exceeded and the system is in the UNATTENDED MODE the procedure will disconnect and set the DLE/EOT SENT and D1 flags.

If the NAK LIMIT is exceeded and the system is in the ATTENDED MODE then, if the REQUEST DISCONNECT REGISTER is SET (=1), the procedure will disconnect and set the DLE/EOT SENT and D1 flags. If the REQUEST DISCONNECT REGISTER is RESET (=0) the procedure will transmit an EOT and enter the RECEIVE TEXT STATE.

CALLED STATION

If the NAK LIMIT is exceeded and the REQUEST DISCONNECT REGISTER is SET (=1) the procedure will disconnect and set the DLE/EOT SENT, NAK LIMIT EXCEEDED and D1 flags.

If the NAK LIMIT is exceeded and the REQUEST DISCONNECT REGISTER is RESET (=0) the procedure will transmit an EOT, set the NAK LIMIT EXCEEDED and D1 flags, and enter the RECEIVE TEXT STATE.

CALLING OR CALLED STATION

If an ACK is received in response to the transmitted message, the procedure will reset the D3 flag and increment the transmission number. If the TRANSMIT DELAY REGISTER is SET (=1) the procedure will wait 10.5 seconds to enable the S-program to prepare any further message for transmission. The procedure will then go the start of the TRANSMIT TEXT STATE.

If an RVI (Reverse Interrupt = DLE/<) is received, the procedure will reset the D3 flag, increment the transmission number, transmit an EOT and enter the RECEIVE TEXT STATE.

If an invalid or no response condition exists, the procedure will transmit an ENQ (this causes the other station to re-transmit the last control character sent). The procedure will transmit an ENQ up to 3 times and, if the condition persists after 3 retries, the line will be disconnected and the DLE/EOT SENT and D1 flags set.

RECEIVE TEXT STATE

Upon entering the RECEIVE TEXT STATE, the procedure will be looking for a message (starting with SOH) or an EOT. The following notes describe subsequent procedure action.

If a message is received, the procedure will compare the received Block Check Character (BCC) with the calculated BCC. If unequal, the procedure will transmit a NAK and go to the start of the RECEIVE TEXT STATE. If equal, the procedure will transmit an ACK, set the D2 flag and compare the received transmission number with the EXPECTED TRANSMISSION NUMBER REGISTER. If this result is unequal, the procedure will set the INVALID TRANSMISSION NUMBER and the D1 flags. If equal, the EXPECTED TRANSMISSION NUMBER REGISTER will be incremented. If the D3 flag has been set by the S-program, the procedure will transmit an RVI and enter the TRANSMIT TEXT STATE, else the procedure will then go to the start of the RECEIVE TEXT STATE.

If an EOT is received and the D3 flag is set the procedure will enter the TRANSMIT TEXT STATE.

CALLING STATION

If an EOT is received, the D3 flag is reset and the station is in the UNATTENDED MODE, the procedure will disconnect and set the DLE/EOT SENT and D1 flags.

If an EOT is received, the D3 flag is reset and the station is in the ATTENDED MODE then, if the REQUEST DISCONNECT REGISTER is SET (=1) the procedure will disconnect and set the DLE/EOT SENT and D1 flags. If the REQUEST DISCONNECT REGISTER is RESET (=0) the procedure will transmit an EOT and go to the start of the RECEIVE TEXT STATE.

CALLED STATION

If an EOT is received and the D3 flag is reset and the REQUEST DISCONNECT REGISTER is RESET (=0) the procedure will transmit an EOT and go to the start of the RECEIVE TEXT STATE. If the REQUEST DISCONNECT REGISTER is SET (=1), the procedure will disconnect and set the DLE/EOT SENT and D1 flags.

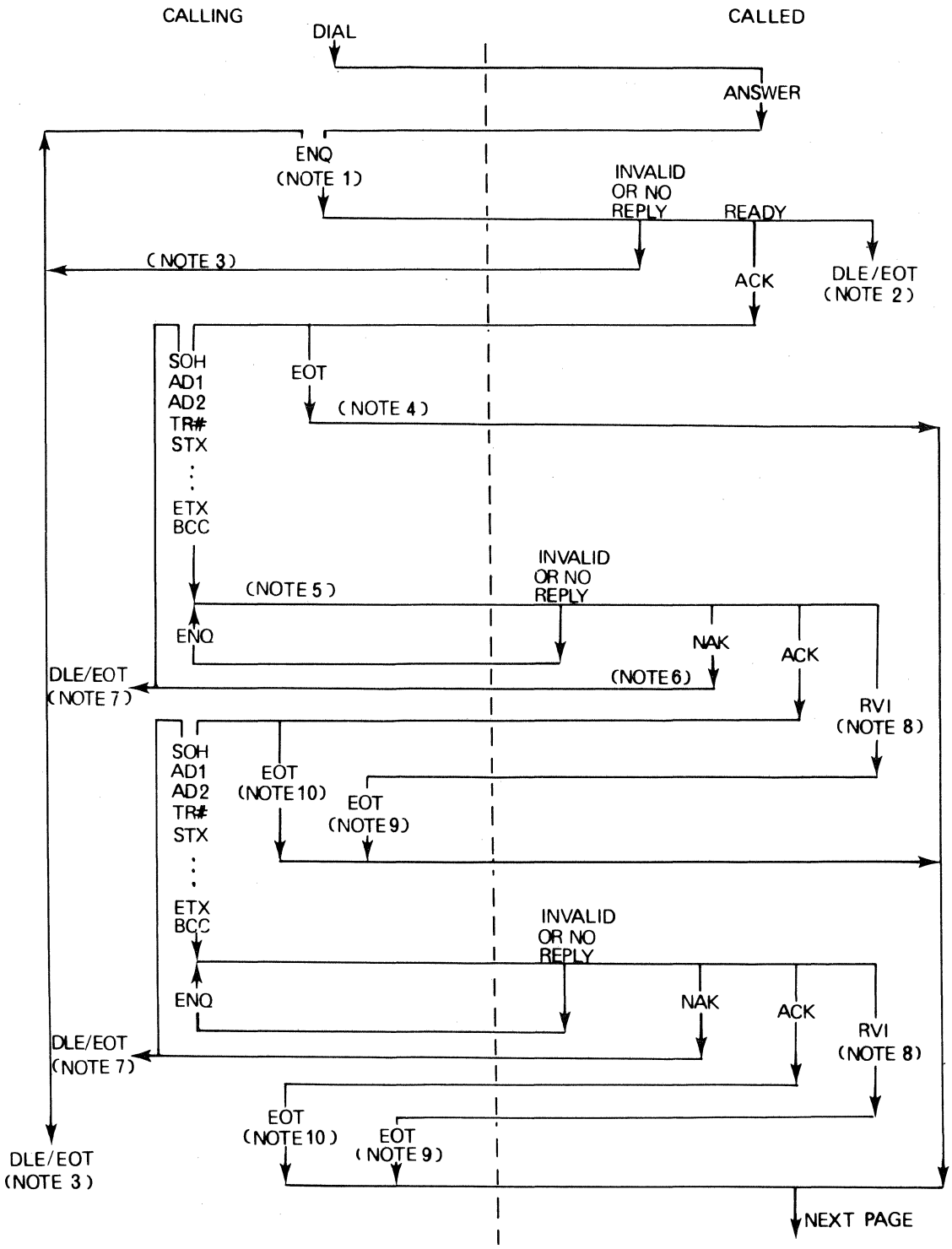


Figure 8–2. Point-to-Point Switched Batch Procedure.

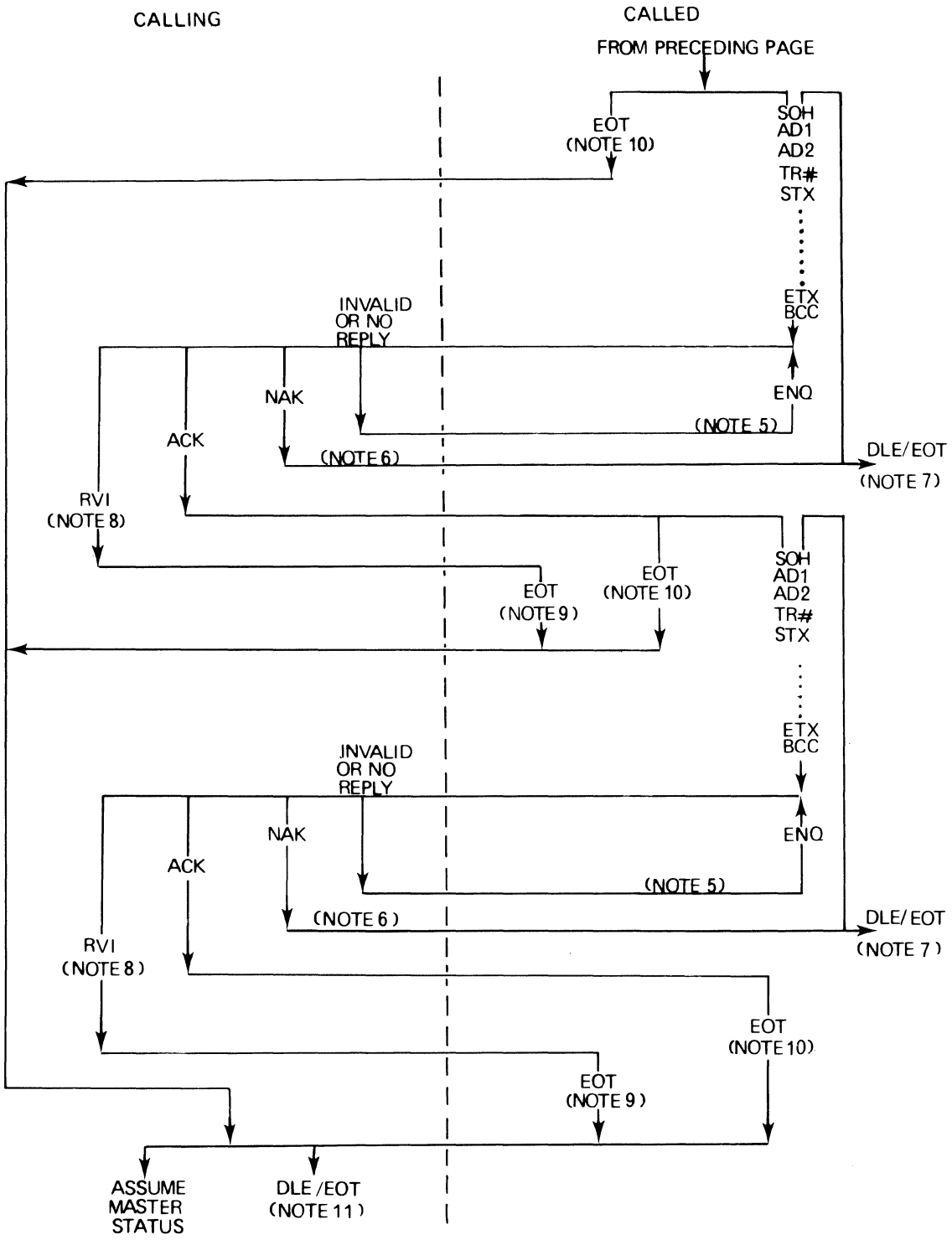


Figure 8–2. Continued

Notes to Figure 8–2.

1. The CALLING station is considered to be the Master Station for the initial transaction and will send ENQ immediately following the establishment of the connection.
2. After a connection is made, the CALLED station will await the receiption of an ENQ. If an ENQ is not received within 30 seconds, the CALLED station will transmit a DLE/EOT and disconnect the line.
3. In the event that no reply is received following the transmission of an ENQ the procedure will re-transmit ENQ until either the S-program resets the D1 flag or the procedure reaches the BID LIMIT. In either case the CALLING station will transmit DLE/EOT and disconnect the line.
4. The CALLING station can enter the receive TEXT STATE by transmitting an EOT. The S-program can accomplish this by having the D3 flag reset.
5. Failure of the TRANSMITTING station to receive an ACK or NAK following message transmission, will cause the procedure to transmit an ENQ. The reception of this ENQ by the receiving station will cause a retransmission of the last control character (ACK or NAK) sent.
6. The receiving station will transmit a NAK to a message if the message contained either bad character parity or a bad BCC character. The receiving station will also transmit a NAK if its receive buffer is still full.
7. In the event the station's NAK-LIMIT is exceeded, this station may transmit a DLE/EOT and disconnect the line.
8. If the receiving station wishes to suspend transmission so that it can transmit a high priority message it can do so by transmitting a reverse interrupt in place of an ACK.
9. The station that receives a Reverse Interrupt must reliniguish the line by transmitting an EOT.
10. If this station has no text to transmit, it will send EOT.
11. If no further communication is desired and this station is the Calling, this station will transmit DLE/EOT and disconnect the line.

SYSTEM REGISTERS

DATA SET INTERFACE REGISTER

As defined in Section 7.

CLEAR TO SEND REGISTER

As defined in Section 7.

TA 214/714 REGISTER

FUNCTION

This register specifies whether or not a TA714 dataset, if being used to automatically answer an incoming telephone call, has its switch set to AUTO and the data set strapped for Auto Dial and Auto Answer. If a TA714 data set is not being used, the register must specify No.

FORMAT

The register size is 1 bit and it is part of the DATA SET REGISTER GROUP.

POSSIBLE VALUES

0 = NO
1 = YES (switch to Auto and strapped as specified).

LOADING

See DATA SET REGISTER GROUP.

REQUIRED

This register must be specified for this line procedure.

DEFAULT VALUES

System Generate/Load – 0
Data Comm System Register Generator – 0

D/C BUFFER LENGTH REGISTER

As defined in Section 7.

PROCEDURE AREA 1

FUNCTION

This register specifies the number of bids that will be made by the CALLING station, in an effort to establish activity, during the BID STATE. In the event of no positive acknowledgement from the CALLED station, the CALLING station's line procedure will set its BID LIMIT EXCEEDED flag. Each bid consists of a transmission of an ENQ character and takes place once every second.

FORMAT

The register size is 8 bits and it contains the limit expressed as a hexadecimal number.

POSSIBLE VALUES

Hex 01 to FF (1 to 255 bids). A value of Hex 00 will give 256 bids.

LOADING

This register must be loaded before this line procedure is initiated.
The register may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions (SL5 only).

REQUIRED

This register must always be specified for the CALLING station.

DEFAULT VALUES

System Generate/Load – 0 (Zero)

RELATED REGISTERS

CALLING/CALLED REGISTER

NUMBER OF TRANSMISSION NUMBERS REGISTER

As defined in Section 7.

PARITY REGISTER

As defined in Section 7. Normally it is set to 1 (EVEN parity).

VRC REGISTER

As defined in Section 7. Normally it is set to 0 (Generate or Check Parity).

ASYNCHRONOUS RATE REGISTER

FUNCTION

This register specifies whether the transmission rate is to be High (normal) or Low (fall-back). This register is only active if the CONTROL/SENSE RATE REGISTER is set to 1 (Control).

FORMAT

The register size is 1 bit and it is part of the TRANSMISSION RATE REGISTER GROUP.

POSSIBLE VALUES

0 = High (normal) rate
1 = Low (fall-back) rate

LOADING

See the TRANSMISSION RATE REGISTER GROUP.

REQUIRED

This register must always be specified for this line procedure.

DEFAULT VALUES

System Generate/Load – 0 (High Rate).

RELATED REGISTERS

CONTROL/SENSE RATE REGISTER.

CONTROL/SENSE RATE REGISTER

FUNCTION

This register specifies that the system is to transmit and receive data at the rate specified in the ASYNCHRONOUS RATE REGISTER.

FORMAT

The register size is 1 bit and it is part of the TRANSMISSION RATE REGISTER GROUP.

POSSIBLE VALUES

1 = CONTROL Rate.

LOADING

See the TRANSMISSION RATE REGISTER GROUP.

REQUIRED

This register must always be specified for this line procedure, and set to 1.

DEFAULT VALUES

System Generate/Load – 0 (SENSE Rate).

RELATED REGISTERS

ASYNCHRONOUS RATE REGISTER

FOUR WIRE RECEIVE DELAY REGISTER

As defined in Section 7.

FOUR WIRE TRANSMIT DELAY REGISTER

As defined in Section 7.

TWO WIRE RECEIVE DELAY REGISTER

As defined in Section 7.

TWO WIRE TRANSMIT DELAY REGISTER

As defined in Section 7.

D/C SEND BUFFER ADDRESS REGISTER

As defined in Section 7.

D/C RECEIVE BUFFER ADDRESS REGISTER

As defined in Section 7.

TRANSMIT MACHINE ADDRESS REGISTER

FUNCTION

This register specifies the terminal address that is transmitted in the header portion of messages.

FORMAT

The register size is 64 bits and it is split into eight address characters. Each 8 bit character contains the column and row numbers that represent the ASCII character to be transmitted.

Only the two most significant 8 bit characters are used by this line procedure.

POSSIBLE VALUES

Any ASCII characters from columns 2, 3, 4, 5, 6 or 7 except DEL (7, 15).

LOADING

This register must be loaded before the line procedure transmits the terminal address.
The register may be loaded by one of the following methods:

a) Data Comm System Register Generator.
b) Magnetic Tape Cassette S-Loader (either as part of a file previously created by the Data Comm System Register Generator or, as part of a program or data object file).
c) S-program instructions.

REQUIRED

A two character address must always be specified if the terminal is to transmit any data.

DEFAULT VALUES

System Generate/Load – NUL (00) codes.

CALLING/CALLED REGISTER

FUNCTION

This register specifies whether the terminal has the responsibility of initiating the activity on the line (CALLING station) or of responding to the CALLING station (CALLED station).

FORMAT

The register size is 1 bit and it is part of the TWO/FOUR WIRE REGISTER GROUP.

POSSIBLE VALUES

0 = CALLING station
1 = CALLED station

LOADING

This register must be loaded before the line procedure is initiated.

See the TWO/FOUR WIRE REGISTER GROUP.

REQUIRED

This register must always be specified.

DEFAULT VALUES

System Generate/Load – 0 (CALLING)

RELATED REGISTERS

ATTENDED MODE REGISTER
AUTO ANSWER REGISTER

ATTENDED MODE REGISTER

FUNCTION

This register specifies whether the CALLING station is attended (operator present) or unattended (no operator).

FORMAT

The register size is 1 bit and it is part of the TWO/FOUR WIRE REGISTER GROUP.

POSSIBLE VALUES

0 = UNATTENDED
1 = ATTENDED

LOADING

This register must be loaded before the line procedure is initiated.

See the TWO/FOUR WIRE REGISTER GROUP.

REQUIRED

This register must always be specified for the CALLING station.

DEFAULT VALUES

System Generate/Load – 0 (UNATTENDED)

RELATED REGISTERS

CALLING/CALLED REGISTER.

RUN/ABORT REGISTER

FUNCTION

This register will permit the procedure to disconnect the line immediately when it is set to ABORT. When the disconnect takes place, the procedure will set the DLE/EOT SENT, INVALID DISCONNECT and D1 flags of the station which enforced the ABORT.

FORMAT

The register size is 1 bit and it is part of the TWO/FOUR WIRE REGISTER GROUP.

POSSIBLE VALUES

0 = ABORT mode.
1 = RUN mode.

LOADING

This register may be loaded at any time.

See the TWO/FOUR WIRE REGISTER GROUP.

REQUIRED

This register must always be specified.

DEFAULT VALUES

System Generate/Load – 0 (ABORT)

REQUEST DISCONNECT REGISTER

FUNCTION

This register specifies whether or not the line procedure should disconnect the communication line at a point in the procedure. This point can be when the NAK LIMIT EXCEEDED flag is set or, when an EOT is received and the D3 flag is reset.

FORMAT

The register size is 1 bit and it is part of the TWO/FOUR WIRE REGISTER GROUP.

POSSIBLE VALUES

0 = DO NOT DISCONNECT
1 = REQUEST VALID DISCONNECT

LOADING

This register may be loaded at any time.

See the TWO/FOUR WIRE REGISTER GROUP.

REQUIRED

This register must always be specified.

DEFAULT VALUES

System Generate/Load – 0 (NO DISCONNECT).

AUTO ANSWER REGISTER

FUNCTION

This register specifies whether the CALLED station is to automatically or manually answer an incoming telephone call.

FORMAT

The register size is 1 bit and it is part of the TWO/FOUR WIRE REGISTER GROUP.

POSSIBLE VALUES

0 = MANUAL ANSWER
1 = AUTO ANSWER

LOADING

The register must be loaded before the line procedure is initiated.

See the TWO/FOUR WIRE REGISTER GROUP.

REQUIRED

This register must always be specified for the CALLED station.

DEFAULT VALUES

System Generate/Load – 0 (MANUAL ANSWER).

RELATED REGISTERS

CALLING/CALLED REGISTER

TRANSMIT DELAY REGISTER

FUNCTION

This register specifies whether or not the line procedure will wait for 10.5 seconds before indicating that the terminal has no data to transmit.

FORMAT

The register size is 1 bit and it is part of the TWO/FOUR WIRE REGISTER GROUP.

POSSIBLE VALUES

0 = DISABLE TRANSMIT DELAY
1 = ENABLE TRANSMIT DELAY

LOADING

This register may be loaded at any time.
See the TWO/FOUR WIRE REGISTER GROUP.

REQUIRED

This register must always be specified

DEFAULT VALUES

System Generate/Load – 0 (DISABLE TRANSMIT DELAY).

NAK LIMIT REGISTER

FUNCTION

This register specifies the number of times that a station is willing to receive a NAK in response to a transmitted message before aborting that transmission and taking further corrective action. The corrective action depends on many factors and is described in the TRANSMIT TEXT STATE.

FORMAT

The register size is 8 bits and it is part of the TWO/FOUR WIRE REGISTER GROUP. It contains the value, expressed in hexadecimal format.

POSSIBLE VALUES

Hex 00 to FF (0 to 255 times)

LOADING

This register must be loaded before the line procedure is initiated.
See the TWO/FOUR WIRE REGISTER GROUP.

REQUIRED

This register must always be specified.

DEFAULT VALUES

System Generate/Load – 00.

TWO/FOUR WIRE REGISTER

As defined in Section 7.

SEND TRANSMISSION NUMBER REGISTER

As defined in Section 7.

EXPECTED TRANSMISSION NUMBER REGISTER

As defined in Section 7.

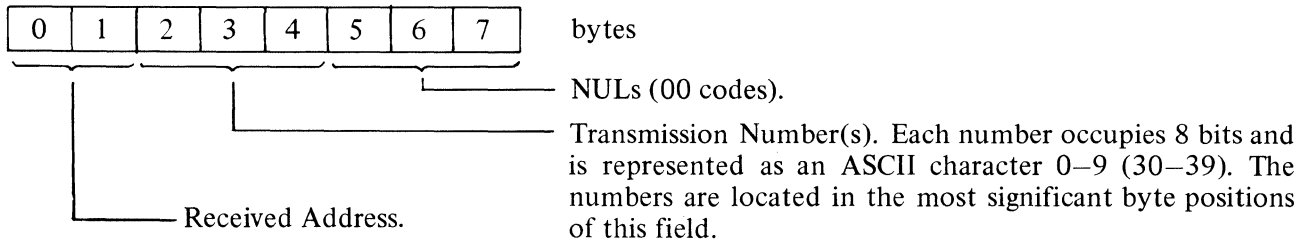
TRANSMISSION HEADER REGISTER

FUNCTION

This register contains the address and transmission numbers from the header portion of the last received message.

FORMAT

The register size is 64 bits and its format is as follows:



RETRIEVING

This register may be retrieved at any time using an S-program instruction.

DEFAULT VALUES

System Generate/Load – NUL (00) codes.

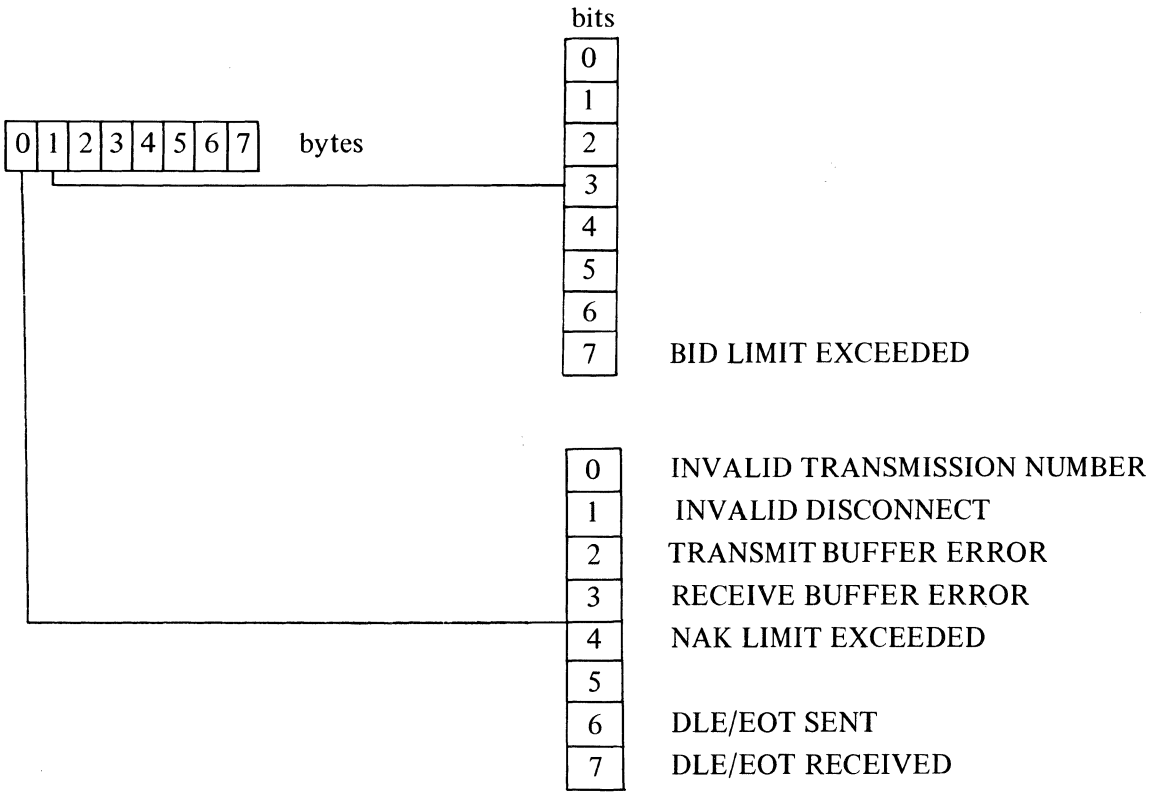
PROBLEM FLAG REGISTER

FUNCTION

This register specifies the precise error condition that exists when a Data Comm Error (D1 flag set) has been flagged.

FORMAT

The register size is 64 bits and it contains the following problem flags:



RETRIEVING/LOADING

This register may be retrieved and loaded at any time using S-program instructions.

REQUIRED

This register should be retrieved and examined immediately after a Data Comm Error has been flagged. The condition should be corrected, the register should then be reset to zero and the Data Comm Error Flag reset.

BID LIMIT EXCEEDED FLAG

This flag is set when, in the BID STATE, the CALLING station is unable to get an affirmative response to its bids.

INVALID TRANSMISSION NUMBER FLAG

This flag is set when the EXPECTED TRANSMISSION NUMBER is not equal to the actual received transmission number as located in the TRANSMISSION HEADER REGISTER.

INVALID DISCONNECT FLAG

This flag is set whenever the procedure disconnects as a result of the ABORT MODE being active.

TRANSMIT BUFFER ERROR FLAG

When transmitting data, if the procedure cannot find a valid termination code (NUL or ETX) within the buffer limit, as defined by the D/C BUFFER LENGTH REGISTER, then this flag is set. The procedure will then transmit an ETX followed by the complement of the calculated BCC in order to force the other station to respond with a NAK.

RECEIVE BUFFER ERROR FLAG

When receiving data, if the procedure does not receive a valid termination code (ETX) within the buffer limit, as defined by the D/C BUFFER LENGTH REGISTER, then this flag is set. The procedure will then continue to accept data from this line, but to ignore such data until an ETX followed by a BCC is received. A negative acknowledgement (NAK) will then be transmitted back to the other station.

NAK LIMIT EXCEEDED FLAG

This flag is set if a station has received more NAKs in response to a transmitted message than are allowed for in the NAK LIMIT REGISTER.

DLE/EOT SENT FLAG

This flag is set when the procedure transmits a DLE/EOT sequence to disconnect the line.

DLE/EOT RECEIVED FLAG

This flag is set when the procedure receives a DLE/EOT sequence. The procedure will then disconnect the line.

DATA COMMUNICATIONS FLAGS

The use of these flags is standard except for the D1 flag which has a secondary function in the BID STATE.

With the PROBLEM FLAG REGISTER zero, the D1 flag must be set by the S-program to enable the procedure to enter the main loop of the BID STATE.

DATA COMM INDICATOR LIGHTS

The CARR (Carrier Frequency Detect) light is not dependant on the line procedure and is explained in Section 3.

DTA R (DATA RECEIVED)

This light is illuminated by the system after it has verified that the Block Check Character (BCC) is correct. It will remain on until the MESSAGE RECEIVED (D2) flag is reset by the S-program.

DTA S (DATA TRANSMIT READY)

This light is illuminated when the S-program sets the TRANSMIT READY (D3) flag. It will remain on until the procedure receives a positive acknowledgement (ACK) in response to the transmission of the message.

CTL R (CONTROL CHARACTER RECEIVED)

This light is illuminated by the procedure as soon as it has received a control character. It will remain on until immediately prior to the transmission of a control character in response to the one(s) received.

The control characters are ACK, NAK, EOT, ENQ, RVI (DLE <), SOH and DLE/EOT.

CTL S (CONTROL CHARACTER SENT)

This light is illuminated by the procedure immediately prior to the transmission of the first control character of a transmission string. It will remain on until the final character of that string is sent.

The control characters are ACK, NAK, EOT, ENQ, RVI (DLE <), SOH and DLE/EOT.

HARDWARE CONSIDERATIONS

The Asynchronous Point-to-Point Switched Batch Line Procedure requires that the structure of the characters to be transmitted and received is defined by the settings on the Asynchronous Data Comm Controller (ADC) board.

Mandatory settings on the ADC board are:

Data Bits	–	7
Parity Odd/Soft Select	–	SOFT SELECT

Optional settings on the ADC board are:

Stop Bits	–	1 or 2 (normally set to 1).

SECTION 9
UTILITIES

CONCEPTS

Utilities are general purpose routines that can be used on a system independent of applicational environment. Three types are available; MEMORY LOAD ROUTINES, INTRINSICS and S-LEVEL UTILITIES. Intrinsics can be considered as a integral part of the system and reside in the interpreter area of memory and can be used independent of the user program. S-level utilities are S-language dependent routines that reside in the user memory and must be loaded using the Memory Load routines.

MEMORY LOAD ROUTINE

This utility provides the means of loading data in object format to memory.

MAGNETIC TAPE CASSETTE S-LOADER

This utility is used to load data in SL3/SL5 object format from magnetic tape cassette to memory. This data must have been dumped previously using the magnetic tape cassette S-dump or on another SL3/SL5 system using a compatible dump routine. The object file can be loaded from a cassette in drive unit 1 or 2. If included in the object file, the Basic and Data Communications System Registers will also be loaded at the same time.

INTRINSICS

Intrinsics include a memory dump routine (Magnetic Tape Cassette S-Dump), routines that determine virtual machine configuration (Basic System Register Utility and Data Communications System Register Utility) and routines that access memory locations (Trace and Memory List).

MAGNETIC TAPE CASSETTE S-DUMP

This utility is used to dump data in SL3/SL5 object format from memory to magnetic tape cassette. The data which has been dumped may be reloaded using the magnetic tape cassette S-loader or loaded to another SL3/SL5 system using a compatible load routine. The dump can be directed to a magnetic tape cassette in cassette drive unit 1 or 2. In each execution of this utility, the sections of memory dumped to a cassette will form an object file preceded by a file identification (up to 5 characters). The areas of memory to be dumped (word orientated) and the file identification is determined by the user. The Basic System Registers (and Data Communications System Registers if Data Communications Base segment is present) may be dumped as part of the object file. These system registers are not transportable to any other SL3/SL5 system. The utility has the capability of adding object files to multiple file magnetic tape cassettes.

BASIC SYSTEM REGISTER UTILITY

This utility routine will generate Basic System Registers which can be loaded directly to memory and/or dumped to a magnetic tape cassette. If the dump option is selected, the system registers are dumped with a file identification (up to 5 characters) to a new or existing (multi-file) magnetic tape cassette. This file can then be loaded to the system using the magnetic tape cassette S-loader. The System Register values are determined by machine configuration and certain decisions made by the operator. A detailed description of the system register values is given in Section 7.

DATA COMMUNICATIONS SYSTEM REGISTER UTILITY

This utility will generate Data Communication System Registers which can be loaded directly to memory and/or dumped to a magnetic tape cassette. If the option is selected, the Data Communication System Registers are dumped with a file identification (up to 5 characters) to a new or existing (multi-file) magnetic tape cassette. This file can then be loaded to the system using the magnetic tape cassette S-loader. The system register values are determined by the operators response to printed questions. A detailed description of the Data Communication System Registers is given in Section 7.

TRACE

This utility provides the ability to list on the console printer the contents of certain user registers before the execution of each program instruction. The hexadecimal form of the instruction, and its memory location in decimal can also be printed. The utility will operate with SL3 and SL5 object programs. A program can be TRACED from its beginning, or from any instruction.

There are four parts of TRACE printout, which can be selected independently, or in any combination:

1. Memory Address and Instruction.
2. Flags.
3. Accumulator.
4. Index Registers.

MEMORY LIST

This utility provides the ability to list, on the console printer, any part of S-Program Memory. The listing of memory is in hexadecimal format (16 digits per word of memory) or in alphanumeric format (8 characters per word). A heading of up to 130 characters may be typed by the operator for information. The contents of the memory selected is printed six words to a line if hexadecimal, or ten words to a line if alphanumeric.

S-LEVEL UTILITIES

S-level utilities are those utilities which are written in SL5 and which have to be loaded into memory (using a Memory Load Routine) before execution. The utility routines that are used to set up Virtual Machine configurations and to set up optional segments for the system generation are as follows:

Basic System Register Utility
Data Communications System Register Utility
Translation Table Generator

The utility routines that are used to transfer data or object files from an existing cassette to an output media (cassette, console printer) are as follows:

Cassette to Cassette/Print (Object)
Cassette to Cassette/Print (Data)
Cassette Copy (Object/Data)
Cassette to Console Printer (Data)

The utility routine that is used to transfer data from the keyboard to an output media (cassette/ console printer) is:

Keyboard to Cassette/Print (Data)

BASIC SYSTEM REGISTER UTILITY

This utility is functionally identical to the Basic System Register Utility described under Intrinsics in this section except that as an S-level utility it must be loaded using a memory load routine and consideration given to the requirement of 4K bytes of user memory necessary for execution.

DATA COMMUNICATIONS SYSTEM REGISTER UTILITY

This utility is functionally identical to the Data Communications System Register Utility described under Intrinsics in this section except that as an S-level utility it must be loaded using a memory load routine and consideration given to the requirement of 4K bytes of User Memory necessary for execution.

TRANSLATION TABLE GENERATOR

This utility routine is used to create a magnetic tape cassette containing the cassette input or output translation table for use at system generation. Only one file may be written to a given magnetic tape cassette; therefore, two tapes are required if both input and output translation tables are required.

The size of the translation table is 16 x 16 = 256 bytes. The translation table is created by indexing the required values for each position in the table.

CASSETTE TO CASSETTE/PRINT (OBJECT)

This utility routine is used to copy selected object files from one magnetic tape cassette to another. Both magnetic tape cassettes can be multi-file tapes. This routine can be used to create object program library tapes or to add files to existing tapes. The specified file can also be printed on the console printer in hexadecimal format. The 5 character file identification can be printed on the console printer for each file in a multi-file magnetic tape cassette.

CASSETTE TO CASSETTE/PRINT (DATA)

This utility routine reads data files from cassette unit 1 and copies multiple files or a single file to cassette unit 2. Data records may be up to 256 bytes in length and must be fixed length records. The file(s) may also be printed on the console printer if required.

CASSETTE COPY (DATA/OBJECT)

This utility routine copies a magnetic tape cassette in cassette drive 1 to a magnetic tape cassette in cassette drive 2. The entire tape is copied from beginning of tape to the double tape mark. This utility can be used to make duplicate data or object file magnetic tape cassettes.

CASSETTE TO CONSOLE (DATA)

This utility routine reads fixed length data records of up to 256 characters from a magnetic tape cassette in cassette drive 1 and prints the records (80 characters per line) on the console printer. If the record size exceeds 160 characters, every third line may have up to 96 characters. The records may be printed on 66–line or 51–line paper and will give correct spacing on the start of each page. The utility can process all files on the magnetic tape cassette and terminate at the end of data, signified by two consecutive tape marks or the physical end of tape. The utility can also process selected files on the cassette, either labelled or unlabelled. The end of a single file is determined by a single tape mark.

KEYBOARD TO CASSETTE/PRINT (DATA)

This utility routine accepts data from the alphanumeric keyboard and writes this data to a magnetic tape cassette in cassette drive 1. The data is formatted in 80 character blocks; if less than 80 characters are indexed, the buffer will be filled with spaces to equal 80 characters. The records can have a blocking factor of 1, 2 or 3. Blocked 1 will give 80 character records, blocked 2 and 3 will give 160 and 240 character records respectively. A magnetic tape cassette will store 880 blocks of 240 characters (2640,80–character records), but can store only 1580 blocks of 80 characters. This difference is due to the greater number of inter-record gaps for the 80 character records. The utility has the capability of creating multi-file, labelled or unlabelled, data magnetic tape cassettes.

CUSTOMER CONFIDENCE PROGRAM

This program is designed to perform a confidence check on the system. It can be used to establish whether the system is functioning correctly and to spotlight degradation of components before they fail. The printed output from this program should always be kept and shown to a Field Engineer who can interpret the results and thus be better informed as to the systems performance.

The program is contained in the MTR magnetic tape cassette, and is started from the Cold Start with the System Test Switch in the up position and the MTR tape in any cassette drive unit. A check of the Read Only Memory (ROM) is performed before the beginning of the main program. The main program on cassette will then perform an extensive test and evaluation of the Processor Memory Keyboard, Console Printer and Peripherals. Any faults that occur will be printed on the right hand side of the printout. These fault codes will allow a Field Engineer to quickly determine the area of trouble.

SECTION 10
ENVIRONMENTAL SPECIFICATIONS

INTRODUCTION

This section describes the power supply required to run a AE500 system and its peripherals, and the environmental conditions under which it will operate.

POWER SUPPLY

Provisions are made to accommodate any of the following voltages (+5%, – 10%), and frequencies.

100	110	115	120	127
200	208	220	230	240 volts 250v
50 or	60 Hz ± 1%			

The total power required depending on printing and forms handling power requirements will be approximately 1.4 KVA with a power factor of 0.75. The power supply is housed in the processor cabinet adjacent to the console. The base system supply uses a normal source transformer with a series regulator on all outputs.

PROTECTION

All power supplies have a short circuit current limiting facility, and an over-voltage protection crowbar is provided in the base power supply.

Any base power supply voltage which goes over or under its safe range causes a main circuit breaker to operate and isolate the machine (base and printer power supplies) from the main input. The machine can also be isolated from main power by using the Cutout switch inside the front door of the processor cabinet.

All power supply output voltages remain within specifications during a main supply drop out of one full cycle (i.e. 20 m sec. on 50 Hz supply).

INTERNATIONAL STANDARDS

The power supplies and electrical specifications conform to the following International Standards and specifications:

British Standards Institution	(B.S. 3861 Part 1, 1965)
Canadian Standards Association	(C22.2 No. 68)
Underwriters Laboratories	(U.L. 114 OCT, 1971)
	(U.L. 478 DEC, 1972)
C.E.E. Publications 10	(Part 1 and 2)
NEMKO Specification	662/55
	661/67
	502/68
German Electrical Specifications	VDE 0730 Part 1 and 2P
	VDE 0871
	VDE 0874
	VDE 0875 (R.F.I.)

LINE CORD

A three wire non-detachable line cord, 10'6" (267 cm) (external length) is normally supplied. Optional 10'6" line cords with moulded plugs are supplied for some countries including the U.S. The line cord is attached to the machine at the bottom of the right hand rear corner of the processor box.

ELECTRICAL SPECIFICATION PLATE

The electrical specification plate specifies the place of manufacture, series number and electrical specifications for the machine. The electrical specifications give all the information required to make connections for the various voltages for the transformer. The specifications plate is attached to the machine so that it is visible without the removal of case panels.

CASSETTE DRIVE UNIT

The cassette drive unit takes its power from the AE500 processor and is powered on and off with the processor.

ENVIRONMENTAL SPECIFICATIONS

This section describes the dimensions of the AE500 system, and the conditions under which it will operate.

DIMENSIONS

The dimensions of the AE500 console and processor and the peripherals that can be attached to it are described below:

Console and Processor Cabinet

Overall dimensions of the assembled console and processor cabinet with their table tops.

Height	30 inches	(76.2 cm)
Width	43.67 inches	(110.9 cm)
Depth	29 inches	(73.7 cm)
Weight	430 lbs	(195 Kg)

Clearances

Operational	3 feet (91.44 cm) at front.
Maintenance	3 feet (91.44 cm) on all sides

Note: The machine may be rolled away from obstructions to obtain the above clearances.

Maximum Heat Output

3,58K BTU/hour (903 Kcal/hour)

ENVIRONMENTAL CONDITIONS

The following environmental conditions are divided into two types: operating environment and non-operating environment. The operating environment specifications describe the range of conditions which allow the machine to operate with the required reliability. The non-operating environment specifications describe the range of conditions which will not damage the machine. All specifications given here also apply to the cassette tape subsystem.

Temperature

	Operating Environment		Non-Operating Environment	
Minimum	1.7°C	35°F	–45°C	–50°F
Maximum	40.5°C	105°F	71.1°C	160°F

When operating the machine, changes in temperature should be as small as possible. A temperature change from the minimum allowed to the maximum allowed must not be allowed to occur in a period of less than eight hours.

Relative Humidity

	Operating Environment	Non-Operating Environment
Minimum	5%	5%
Maximum	95%	100%

The relative humidity may vary from a minimum to a maximum within the temperature range shown, but humidity changes from an extreme of the operating humidity to the other must not be allowed to occur in a period of less than four hours.

Barometric Pressure

Operating Environment	Sea level to approximately 10,000 feet altitude (3,048m)
Non-Operating Environment	Sea level to approximately 50,000 feet altitude (15,340m)

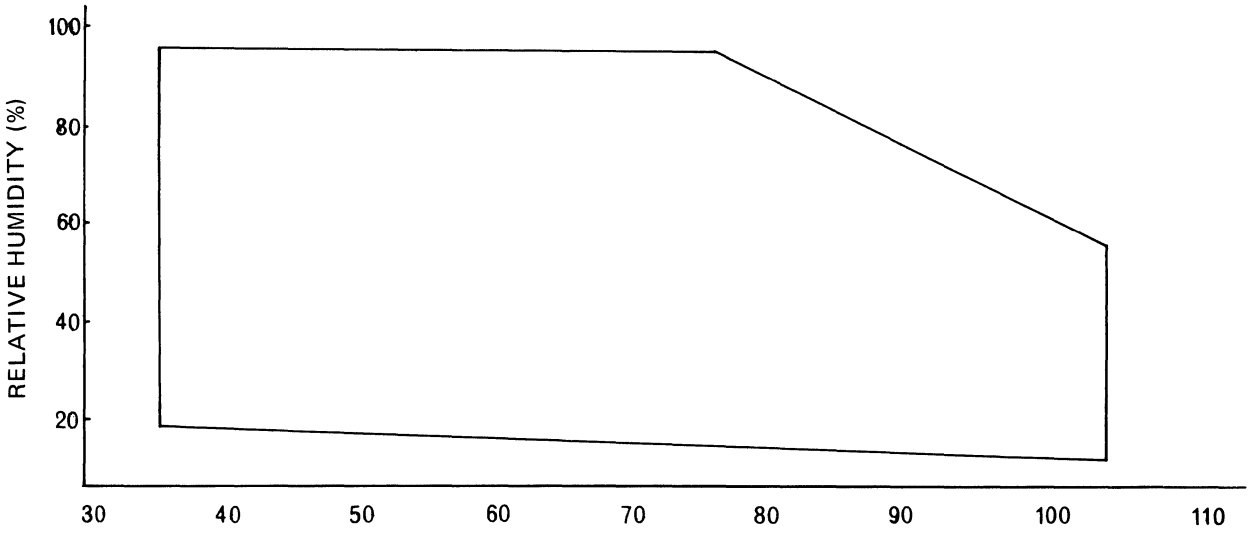


Figure 10–1. Temperature and Humidity Operating Range

APPENDIX A
GLOSSARY OF TERMS

ABSOLUTE ADDRESS

An address in memory which corresponds to a physical location, and is the address used by the hardware to gain access to a memory location. (See RELATIVE ADDRESS)

ABSOLUTE UPPER LIMIT OF MEMORY

The hardware address of the highest available memory location. The absolute upper limit of memory depends upon the number of Random Access Memory boards installed in the system.

ALGEBRAIC OPERATIONS

The normal arithmetic operations which are used to perform calculations (ADD, SUBTRACT, DIVIDE, MULTIPLY).

ALPHA CHARACTER

Any printable character. An alpha character may be the letters A–Z, the numerals 0–9, and any additional special characters. Alpha characters are represented within AE500 systems using the ASCII code set and occupy one byte each. Arithmetic may not be performed on alpha characters.

ASCII/ANSCII

See USASCII

ASYNCHRONOUS DATA COMMUNICATION

A data communication mode of transmission in which a predetermined number of bits are enclosed by "start" and "stop" bits. The time interval between start and stop bits is precise, but the interval between consecutive start bits is arbitrary.

AUTOMATIC READ RETRIES

In the event of a cassette read error, system software will backspace and re-read the record containing the error. The re-read is alternatively performed with the read amplifier gain high and low. A Basic System Register specifies the number of read retries to be performed.

AUTOMATIC WRITE RETRIES

In the event of a cassette write error, system software will backspace the tape, erase the record containing the error, and re-write the record approximately 5 inches further along the tape. A Basic System Register specifies the numbe of write retries to be performed.
Note: Since each write retry extends the inter-record gap by approximately 5 inches, more than 4 write retries will cause a timeout error when the tape is subsequently read.

BANDWIDTH

The difference between the lowest and highest frequencies a circuit/communication link may use. Bandwidth is a limiting factor in the data transmission rate which may be employed.

BASIC REGISTERS

Reserved areas of memory which define non data communication aspects of system operation. These registers may be altered by the user to suit applicational requirement.

BDI

Burroughs Direct-connect Interface. BDI permits Burroughs data communication systems to communicate without datasets (direct connect) at distances up to 15,000 feet and at data transfer rates up to 64,000 bits per second.

BDLC

Burroughs Data Link Control. An advanced data communication method in which the line procedure is bit oriented for greater flexibility.

BIT

Binary Digit. A bit may have one of two values, 0 or 1.

BLOCK CHECK CHARACTER (BCC)

A character added at the end of a transmitted or stored data block, to provide for error checking. The value of the BCC is calculated from the values of all the data characters in the block, and checked using the same calculation when the data is received or retrieved.

BOOLEAN OPERATIONS

The binary logical connectives of AND, OR, and NEGATE.

BOOTSTRAP

A means of loading System Software into a system when System Software normally provides the load routine. In AE500 systems the bootstrap consists of a specified load routine in Read Only Memory.

BOT

Beginning of Tape. A marker is provided on computer grade magnetic tape to designate the start of the useable tape.

BUFFER

A means of compensating for the various data transfer rates of different parts of the system. A dedicated physical buffer may be provided (hard buffer), or memory space may be reserved (soft buffer), to store data arriving at one rate and to subsequently emit the data at a different rate.

BUS

An electrical connection common to many circuits.

BYTE

A field of 8 bits. Usually, a bit positional weighting is assigned such that each bit position represents a power of 2; 2^0 to 2^7. A byte can represent 1 of 256 possible values.

CALL

Cause a subroutine to be executed. A subroutine may be "called" from different points in a program, and control will be returned to the instruction immediately after the "calling" instruction, after execution of the subroutine.

CALLED STATION

In a Point to Point environment, the data communication system which is responsible for answering the phone call is known as the CALLED STATION.

CALLING STATION

In a Point to Point environment, the data communication system which is responsbile for initiating the phone call and establishing the communication link is the Calling Station.

CARRIER FREQUENCY

The basic oscillation rate of a carrier wave.

CARRIER WAVE

An electrical transmission which is altered in some way (phase, amplitude, frequency) to carry an analogue of a digital signal.

CDC/CDV

Check Digit Compute/Verify. A method of checking the integrity of numeric fields. CDC attaches a redundant digit to numeric fields. The value of the digit is computed from the values of all the other digits in the field. CDV performs the same calculations, and verifies that the check digit is correct.

CHARACTER

A field of 8 bits; identical to a byte.

CLOCK

A train of timing pulses which synchronises the operation of the machine logic circuits.

COLD START

The loading procedure which must be used when the system memory contents are considered to be invalid. Cold Start loads the system software into system memory, and sets the system into a "ready to use" (WARM) condition. Since Cold Start loads system software only, the first use of the system will normally be to load user software and data.

COLD STATE

The initial system state which is entered from power on or from pushing the Cold Start button.

COMMON CARRIER ENVIRONMENT

A data communications environment in which the digital signals from the data communication system are used to modulate a carrier wave. This permits the digital signal (dc voltage levels) to be propagated along a communication link which will only accept limited bandwidth ac signals. The generation and modulation/demodulation of the carrier wave is performed by the data set.

COMMUNICATION LINK

The data transmission medium which connects remote data communication systems.

COMPILER

A program which translates a source program into an object program.

CONCATENTATION

The linking together of data communication terminals to permit them to share a single data set/communication link.

CONTROL PROGRAM

A program with supervisory functions, usually associated with resource management.

CRC

Cyclic Redundancy Check. An error checking mechanism similar to Longitudinal redundancy check (LRC), but permitting greater integrity and longer data blocks. It is a two byte value computed by dividing the entire message by a polynomial and taking the remainder as the CRC value. The polynomial used is $X^{16} + X^{15} + X^2 + X^0$.

DATA COMMUNICATION REGISTERS

Reserved areas of memory which define the data communication aspects of system operation. These registers may be altered by the user to suit applicational requirements.

DATA COMMUNICATION SYSTEM

Used to refer to any Computer, Video Display, Terminal etc., which is capable of communicating with other data communication systems via a communication link.

DATA HIGHWAY

Eight parallel conductors (busses) which carry information around the system. A data highway may be able to pass information in either direction, in which case it is called a bi-directional data highway.

DATA SET

A device which permits a data communication system to utilize various types of communication links. Data sets provide a standard machine/communication link interface and convert signals from the machine to an analogue version suitable for transmission via the particular communication link. Data sets are frequently referred to as MODEMS since they Modulate and Demodulate a carrier signal.

DEFAULT VALUES

The value to which a variable is initialized, and at which it will remain, unless altered by the user.

DESTINATION FIELD

See RESULT FIELD

DIGIT

A field of four bits. Usually, a bit positional significance is assigned such that each bit position represents a power of 2, from 2^0 to 2^3. A digit can represent 1 of 16 possible values. For arithmetic purposes, digits may be used in BCD (binary coded decimal) mode, and will represent the values 0–9.

DIRECT CONNECT

A communication link, between data communication systems, which does not use data sets.

DOWNSTREAM

In a concatenated group of data communication terminals, the direction away from the data set is referred to as downstream.

DUPLEX

1. Half Duplex. A data communication mode of operation which permits communication in both directions, but not simultaneously.
2. Full Duplex. A data communication mode of operation which permits communication in both directions simultaneously.

EBCDIC

Extended Binary Coded Decimal Interchange Code. This is a well defined code for the representation of data.

EOT

1. End Of Tape. A marker is provided on computer grade magnetic tape to designate the end of the usable tape.

2. End of Transmission. A data communication control character which terminates a transmission sequence.

EXECUTE STATE

The SL3/5 system state in which all processing work is done.

FETCH

The mechanism which extracts the next instruction in execution sequence from memory. Two types of fetch are required in a Virtual Machine environment; the next micro instruction to be executed is brought from memory through a hardware controlled fetch cycle; the next S-instruction to be executed is brought from memory by a micro programmed fetch routine.

FIRMWARE

A collective term used to describe permanently resident micro programmed interpreters and systems software.

FLAG

A one bit register. The flag can be set ON or OFF (1 or 0) by system conditions and/or the user program, and can be examined by the user program to allow alternative routines to be executed.

FUNCTIONAL CODE

The codes emitted by certain keys on the console keyboard are interpreted by system software as control information. Key Codes which are interpreted as functional codes include the PK's the OCK's and the line advance keys.

GATE

An electronic circuit with several inputs and one output. The output may be set to one of two voltages (representing 0 and 1) in response to a predetermined input pattern of voltages representing 0's and 1's. The predetermined pattern which causes a 1 to appear at the output may be chosen to reflect the Boolean functions of AND, OR and NEGATE.

HARDWARE

The physical components of a system.

HARD(WARE) INTERRUPT

A signal which forces a micro program subroutine jump to a fixed location. The signal is generated by various systems conditions, usually associated with I/O functions.

HEADER

The portion of a data communication message between the SOH (Start of Header) and the STX (Start of Text) characters. The header contains address and transmission number characters.

HERTZ (Hz)

The unit of frequency. 1Hz = 1 cycle per second.

ID

Identity.

INPUT/OUTPUT (I/O)

A general term referring to all peripheral devices and their associated control circuits.

INTERRUPT

See HARDWARE INTERRUPT

INTERPRETER

The portion of System Software which provides the micro programs to implement user program instructions.

INTRINSIC

System Software utility routines which can be permanently resident in the system concurrently with other intrinsics.

JUMPER

See STRAP

K BYTE (KB, Kilobyte)

1024 (2^{10}) Bytes

LARGE SCALE INTEGRATED CIRCUITRY (LSIC)

Advanced technology components which permit major subsystems of the machine electronics to be fabricated in single packages.

LEASED LINES

Dedicated connections through the public telephone networks.

LINE PROCEDURE

An exactly defined protocol for data communication message transfers.

LOAD ROUTINE

A program (provided by System Software) which brings "data" into memory via a peripheral device. "Data" may be System Software, user programs, or data required by User Programs.

LOGIC

Hardware circuits which perform Boolean functions.

LOOP

A programming term which refers to the repeated execution of the same series of instructions until a predetermined condition is sensed. If the loop has no exit (either deliberately, or through faulty programming) it is known as an infinite loop.

LRC

Longitudinal Redundancy Check. A block check character (BCC) formed by taking the EXCLUSIVE OR of all corresponding bit positions in the data block to form the respective bit position in the BCC.

MEDIUM SCALE INTEGRATED CIRCUITRY (MSIC)

Advanced technology components identical in structure to LSIC but of lesser complexity.

MICRO INSTRUCTIONS

The machine language instructions which are interpreted directly by the hardware. Micro instructions are sometimes known as Primary Instructions.

MICRO PROGRAM HANDLERS

System Software micro programs which exist for each type of peripheral device. An interrupt from a particular peripheral device will be processed by the appropriate micro program handler.

MICRO PROGRAMS (MICRO CODED)

Micro Programs are programs of micro instructions, and as such are the only programs which the system can physically execute. All other programs are executed through micro program interpretation of the program instructions.

MODEM

See DATA SET

MODULATE

Alter a carrier wave in amplitude, frequency, or phase to provide an analogue of a digital signal.

MONETARY SAFEGUARD SYMBOL

The character printed immediately to the left of the most significant digit of monetary fields to prevent fraudulent addition of further digits.

NON-DESTRUCTIVE MEMORY READ

AE500 random access memory contents are not changed by a memory read cycle. In some memory systems (viz core memory), the memory read cycle causes the memory location to be cleared, requiring a further write cycle to replace the contents.

NUMERIC DIGIT

The numerals 0–9. Numeric digits are represented within AE500 systems in Binary Coded Decimal, and may be used in arithmetic.

OBJECT CODE

Coded instructions which are interpreted dynamically by System Software to perform the functions required by the programmer.

OBJECT FILE

A recorded version of an S-language (object) program, which may be loaded using a load routine appropriate to the recording medium, and is then available for execution.

OBJECT PROGRAM

A program which may be executed interpretirely by a computer. Programs for AE500 systems are written in COBOL source language and compiled into AE500 object code.

OCK

Operation Control Key. Four keys (marked I, II, III, IIII) on the numeric keyboard and two keys (marked I/III, II/IIII) on the alpha keyboard, which are used to terminate keyboard entries.

OSCILLATOR

An electronic circuit which produces a train of regular voltage fluctuations.

OVERLAPPED FETCH AND EXECUTE

A function of the timing machine state logic which permits the next micro instruction in execution sequence to be brought from memory and partially decoded, at the same time as the current instruction is executing.

PARITY

A means of checking the integrity of stored or transmitted data. A redundant bit is attached to each data item. The value of the bit (0 or 1) is selected by the parity mechanism such that the total number of bits of value 1 in the data item (including parity) is even (even parity) or odd (odd parity).

PERIPHERAL (DEVICE)

System components which provide input and/or output of information to/from the system.

PK (PSK)

Program Key. The program keys are the top row of blue keys on the console keyboard. PK's may be used to select alternative routines within a user program, or to control system functions by selecting alternative routines within a micro program.

PRINTED CIRCUIT BOARD (PCB)

A flat board of non-conducting material bearing etched copper conductors which connect the various electronic components mounted on the board.

PROCESS

A series of continuous actions which are carried out one at a time.

PROCESSOR

The section of the machine which interprets machine language instructions (micro instructions).

PROGRAM

A set of step by step instructions designed to solve a particular problem.

PROGRAM START

The point in time at which SL3/5 user program execution is initiated through depression of PKI from the Ready State.

RANDOM ACCESS MEMORY (RAM, READ/WRITE MEMORY)

A type of memory which can be written to and read from. RAM contents are lost when power is removed from the system.

READ ONLY MEMORY (ROM)

A type of memory whose contents are fixed at the time of manufacture. ROM contents will not be altered by a write operation, and are retained with power removed from the system.

READY STATE

The central SL3/5 System state, from which all other states may be entered, and to which all other states exit.

RELATIVE ADDRESS

An address in memory which is defined by its offset from a base address. System Software computes and uses the correct absolute address. Relative addressing permits programs to reside, unaltered, anywhere in memory. (see ABSOLUTE ADDRESS)

RELOCATABLE CODE

Programs which use Relative Addressing, and may be executed without modification anywhere in memory. System Software micro programs have this characteristic, through the use of a special register/mechanism in the AE500 processor architecture.

RESULT FIELD

Registers or memory areas to which data is directed after manipulation.

SIMPLEX

A data communication mode of operation in which communication is in one direction only.

S-INSTRUCTION

Secondary level instruction, or object instruction. S-instructions are interpreted by strings of instructions (primary instructions) in system software.

S-LANGUAGE

The set of S-instructions which are interpreted by systems software to perform functions required by the programmer.

SL3, SL5

System Language 3 and System Language 5. These are user program languages which may be interpreted by AE500 systems. Programs may be written in COBOL and compiled to SL3 or SL5 object code, or existing L8000/TC3500/TC5000 SL3 or SL5 programs may be used.

SOFT DEFINITION

To provide maximum flexibility in the application of AE500 systems, many functions of the system can be altered before or during program execution. This is accomplished through reserved locations in memory, called the Basic and the Data Communication Registers, to which System Software refers during program interpretation. This concept is known as "Soft Definition" of the configuration.

SOURCE FIELD

Registers or memory areas from which data is taken for manipulation.

SOURCE PROGRAM

A program written in a problem oriented "high level language". Such a program requires translating into object code to permit its execution. The translation process is performed by a special program called a compiler. The high level language used by AE500 systems is COBOL.

S-PROGRAM

A program of S-instructions such as the object program intepreted by AE500 systems.

STACK

A series of registers (physical or reserved areas of memory) which may store quantities on a "first in last out" (FILO) basis.

STANDBY STATE

The SL3/5 system state in which the system is powered on but inactive. This state simulates a powered off condition, but retains power to the processor and memory, since a full power off would necessitate a Cold Start.

STRAP

An electrical connection between two points which may be installed or removed in order to alter some characteristic of circuit operation.

SWITCHED LINES

"Dial up" connections through the public telephone exchange networks.

SYLLABLE

A field of 16 bits (1/4 word). A syllable is required to store each instruction in an SL3 program.

SYNCHRONOUS DATA COMMUNICATION

A data communication mode of transmission in which synchronization between transmitter and receiver is achieved at the beginning of a transmission, and accurately maintained during the entire transmission.

SYSTEM

Integrated collection of hardware and system software which is operable as a unit.

SYSTEM ERROR

Any error which system software cannot correct automatically constitutes a system error.
Examples are – attempted execution of invalid S-instruction; attempted access of non-present peripheral device; attempted access of non-present memory.
All systems errors return the system to the Ready State, and display a code pattern on the keyboard indicators to designate the error.

SYSTEM REGISTERS

Reserved areas of memory which have special significance. System registers describe the hardware configuration and operational requirements to the virtual machine, and also provide the user registers which are available to the S-program.

SYSTEM SOFTWARE

Special programs, written by Burroughs which simplify system operation and increase efficiency. System software includes the basic control programs and interpreters intrinsic to the system, and the optional utilities which enhance system operation.

TAPE MARK

A control record which is written onto cassette tape. The tape mark can be recognised by the cassette drive unit control logic while the tape is moving at high speed. The use of tape marks permits user programs to quickly locate specific records on the tape.

TASK

See PROCESS

TERMINAL

Generally used to refer to a data communication system which does not have responsibility for network control.

TDI

Two Wire Direct-connect Interface. TDI permits Burroughs data communication systems to communicate without data sets (direct-connect) at distances up to 1000 feet.

TIMING MACHINE STATE LOGIC

The central control circuitry of the processor. The timing machine states control the fetch and execution of micro instructions.

TRANSLATION TABLE

Translation tables are used to convert one representation of data into another representation, for example ASCII into EBCDIC.

TRANSMISSION NUMBER

Transmission numbers may be included in data communication messages to permit the user program to check that no messages have been missed. Transmission numbers may be decimal, in which case there may be one (0–9), two (00–99) or three (000–999) transmission numbers, or may be Even/Odd type in which case the single number is alternatively 0 and 1.

TRANSPARENT

1. Transparent to the user. Certain aspects of the machine/system software architecture are not accessible by the user, and are described in this manual for interest only. These aspects are referred to as transparent to the user.

2. Transparent data. The data communication hardware of AE500 systems is capable of transmitting data with or without parity generation and checking. When parity is not used, the transmission is said to be in transparent mode.

3. Transparent text. Certain data communication line procedures provide the ability to transmit/receive blocks of data containing control characters, and treat the control characters as data only. These data blocks are known as transparent text.

UPSTREAM

In a concatenated group of data communication terminals, the direction toward the data set is referred to as upstream.

USASCII

United States of America Standard Code for Information Interchange, often abbreviated to ASCII. This is a well defined code for the representation of data. The internal representation of data in AE500 systems is in USASCII code.

USER REGISTERS

Reserved areas of memory which are altered through interpretation of user programs to provide the results required by the programmer.

VIRTUAL MACHINE

The Virtual Machine is the machine the user "sees". The physical machine requires complex programming, especially for the input/output aspects. To simplify the operation and programming of the machine, a micro programmed interface is provided between the user and the physical machine. The micro programs treat reserved areas of memory as "user registers", and perform the intimate control required by I/O devices. User program instructions, consisting of arbitrary codes representing an operation, are examined by the micro programs, and the required operation is performed. It is this interaction between the physical machine and the micro programs which is referred to as a Virtual Machine.

VISIBILITY MODE

Applies to the systems software controlled action which alters the print head/printed line alignment to permit full visibility of the printed information.

VOICE GRADE

Refers to a communication link which is designed for vocal transmissions, for example, normal public telephone lines. Data sets are required to permit voice grade lines to carry digital information.

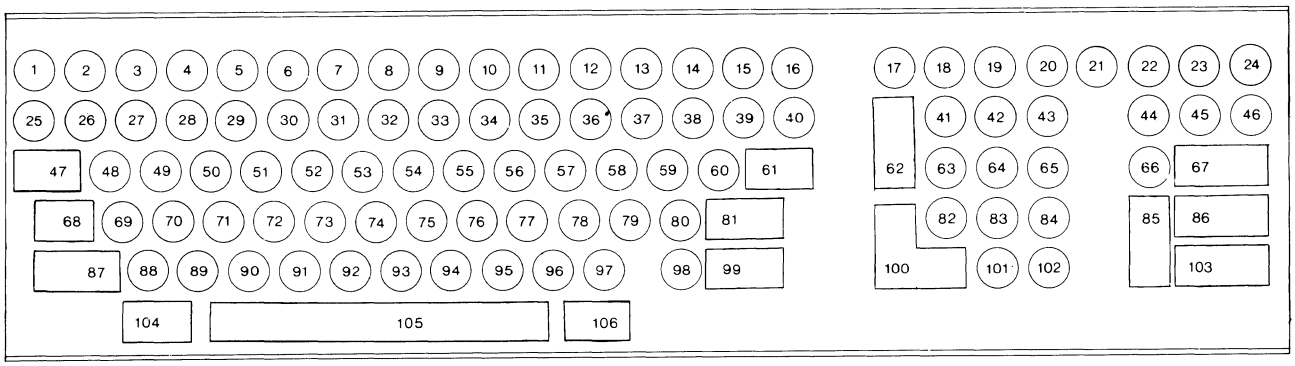
WARM START

The procedure which makes the system "ready to use" when system software has already been loaded into system memory. Any user software/data previously loaded will still be valid.

WORD

A field of 64 bits. A word is treated as 16 digits for arithmetic purposes, or as 8 bytes (characters) for alphanumeric storage.

APPENDIX B
ALPHA AND NUMERIC KEYBOARD VARIANTS



KEY POSITION	KEYTOP LEGEND		INTERNAL REPRESENTATION		LANGUAGE									
					1	2	3	4	5	6	7	8	9	10
	LOWER CASE	UPPER (*) CASE	LOWER CASE	UPPER (*) CASE	U.S.A.	UNITED KINGDOM	FRENCH	ITALIAN	GERMAN	PORTUGUESE	SPANISH	SCANDINAVIAN	CROATIAN	FINNISH/SWEDISH
1 thru 24			– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
25	O/C		– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
26	1	!	31	21	X	X	X	X	X	X	X	X	X	X
27	2	"	32	22	X	X	X	X	X	X	X	X	X	X
28	3	#	33	23	X					X	X			
28	3	£	33	23		X	X	X	X			X	X	X
29	4	$	34	24	X	X	X	X	X	X	X	X	X	X
30	5	%	35	25	X	X	X	X	X	X	X	X	X	X
31	6	&	36	26	X	X	X	X	X	X	X	X	X	X
32	7	'	37	27	X	X	X	X	X	X	X	X	X	X
33	8	(	38	28	X	X	X	X	X	X	X	X	X	X
34	9	)	39	29	X	X	X	X	X	X	X	X	X	X
35	0	—	30	5F	X	X	X	X	X	X	X	X	X	X
36	–	=	2D	3D	X	X	X	X	X	X	X	X	X	X
37	^		5E		X	X	X	X	X	X	X			
37	Ü		5E									X		X
37	Ž		5E										X	

KEY POSITION	KEYTOP LEGEND		INTERNAL REPRESENTATION		LANGUAGE									
					1	2	3	4	5	6	7	8	9	10
	LOWER CASE	UPPER (*) CASE	LOWER CASE	UPPER (*) CASE										
38	@		40		X	X	X	X		X	X			X
38	§		40						X				X	
38	Ö		40									X		
39	←		– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
40	O/C		– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
41	7		7 (2)		X	X	X	X	X	X	X	X	X	X
42	8		8 (2)		X	X	X	X	X	X	X	X	X	X
43	9		9 (2)		X	X	X	X	X	X	X	X	X	X
44	RESET		– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
45	C		(3)		X	X	X	X	X	X	X	X	X	X
46	M		(3)		X	X	X	X	X	X	X	X	X	X
47	↑		– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
48	Q		51		X	X		X	X	X	X	X	X	X
48	A		41				X							
49	W		57		X	X			X	X	X	X	X	X
49	Z		5A				X	X						
50	E		45		X	X	X	X	X	X	X	X	X	X
51	R		52		X	X	X	X	X	X	X	X	X	X
52	T		54		X	X	X	X	X	X	X	X	X	X
53	Y		59		X	X	X	X		X	X	X		X
53	Z		5A						X				X	
54	U		55		X	X	X	X	X	X	X	X	X	X
55	I		49		X	X	X	X	X	X	X	X	X	X
56	O		4F		X	X	X	X	X	X	X	X	X	X
57	P		50		X	X	X	X	X	X	X	X	X	X
58	]		5D		X	X	X	X			X			
58	Ü		5D						X					
58	Ã		5D							X				
58	Ä		5D									X		X
58	Š		5D										X	
59	;	+	3B	2B	X	X	X	X	X	X	X	X	X	X
60	—		5F		X	X	X	X	X	X	X		X	X
60	Ä		5F									X		

KEY POSITION	KEYTOP LEGEND		INTERNAL REPRESENTATION		LANGUAGE									
					1	2	3	4	5	6	7	8	9	10
	LOWER CASE	UPPER (*) CASE	LOWER CASE	UPPER (*) CASE										
61	↑		– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
62			– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
63	4		4 (2)		X	X	X	X	X	X	X	X	X	X
64	5		5 (2)		X	X	X	X	X	X	X	X	X	X
65	6		6 (2)		X	X	X	X	X	X	X	X	X	X
66	RE		(3)		X	X	X	X	X	X	X	X	X	X
67	⊞		– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
68	SHIFT LOCK		– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
69	A		41		X	X		X	X	X	X	X	X	X
69	Q		51				X							
70	S		53		X	X	X	X	X	X	X	X	X	X
71	D		44		X	X	X	X	X	X	X	X	X	X
72	F		46		X	X	X	X	X	X	X	X	X	X
73	G		47		X	X	X	X	X	X	X	X	X	X
74	H		48		X	X	X	X	X	X	X	X	X	X
75	J		4A		X	X	X	X	X	X	X	X	X	X
76	K		4B		X	X	X	X	X	X	X	X	X	X
77	L		4C		X	X	X	X	X	X	X	X	X	X
78	\		5C		X	X								
78	M		4D				X	X						
78	ö		5C						X					X
78	Ç		5C							X				
78	ñ		5C								X			
78	æ		5C									X		
78	č		5C										X	
79	[		5B		X	X	X	X			X			
79	ä		5B						X					X
79	õ		5B							X				
79	ø		5B									X		
79	ć		5B										X	
80		*	3A	2A	X	X	X	X	X	X	X	X	X	X

KEY POSITION	KEYTOP LEGEND		INTERNAL REPRESENTATION		LANGUAGE									
					1	2	3	4	5	6	7	8	9	10
	LOWER CASE	UPPER (*) CASE	LOWER CASE	UPPER (*) CASE										
81	II	IIII	– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
82	1		1 (2)		X	X	X	X	X	X	X	X	X	X
83	2		2 (2)		X	X	X	X	X	X	X	X	X	X
84	3		3 (2)		X	X	X	X	X	X	X	X	X	X
85	I		– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
86	III		– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
87	SHIFT		– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
88	Z		5 A		X	X				X	X	X		X
88	W		57				X	X						
88	Y		59						X				X	
89	X		58		X	X	X	X	X	X	X	X	X	X
90	C		43		X	X	X	X	X	X	X	X	X	X
91	V		56		X	X	X	X	X	X	X	X	X	X
92	B		42		X	X	X	X	X	X	X	X	X	X
93	N		4E		X	X	X	X	X	X	X	X	X	X
94	M		4D		X	X			X	X	X	X	X	X
94	\		5C				X	X						
95	,	<	2C	3C	X	X	X	X	X	X	X	X	X	X
96	.	>	2E	3E	X	X	X	X	X	X	X	X	X	X
97	/	?	2F	3F	X	X	X	X	X	X	X	X	X	X
98	SHIFT		– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
99	I	III	– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
100	0		0 (2)		X	X	X	X	X	X	X	X	X	X
101	00		00 (2)		X	X	X	X	X	X	X	X	X	X
102	000		000 (2)		X	X	X	X	X	X	X	X	X	X
103	II		– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
104	REPEAT		– (1)	– (1)	X	X	X	X	X	X	X	X	X	X
105	SPACE		20	20	X	X	X	X	X	X	X	X	X	X
106	REPEAT		– (1)	– (1)	X	X	X	X	X	X	X	X	X	X

(1) These keys are control and special function keys and have no internal representation.
(2) These numeric keys produce 4–bit (BCD) representation only.
(3) These special keys set certain flags only.
(*) Where no upper case legend appears, the internal representation is as for lower case.

APPENDIX C
EXCEPTION CONDITION INDICATORS

This appendix describes the possible patterns of the D indicators when they are used to show Input/Output Errors or System conditions.

During the EXECUTE STATE, the lights D2 to D8 may be illuminated or extinguished programmatically but the D1 light can only be affected under interpreter control.

INPUT/OUTPUT ERRORS

These error patterns will be set up by the interpreter whenever an I/O error has occurred during the execute state. They are extinguished immediately prior to the execution of the next I/O instruction.

The D1 light will always be illuminated for this type of exception and the READY light will be extinguished.

MAGNETIC TAPE CASSETTE

U U in the positions of D7 and D8 indicate which cassette drive has the error specified, as follows:

(neither) – drive 1
D8 – " 2
D7 – " 3
D7 D8 – " 4

Light Pattern							Error
D1	D3	D4			U	U	Not Ready – Drive may be open or there may not be a cassette in the drive.
D1	D3	D4		D6	U	U	File protected – The write enable plugs are not covering the write protect holes.
D1	D3	D4	D5		U	U	Read Error – A read error has been declared after the interpreter-controlled read retries have been unsuccessful.
D1	D3	D4	D5	D6	U	U	Write Error – A write error has been declared after the interpreter-controlled write retries have been unsuccessful.

SYSTEM CONDITIONS

These light patterns will be set up by the interpreter whenever the system enters the Ready State from the Execute State. These patterns will show the reason why the system entered the Ready State. The READY light will always be illuminated.

Light Pattern							Condition
			(none)				Terminated by Ready Key.
						D8	Orderly termination by use of a STOP S-instruction.
				D5			Data Error on Object Program File – An attempt was made to load an invalid object file.
			D4				Non-present peripheral – An S-program attempted to use a peripheral which was not attached to the system.
		D3					Invalid S–code.
		D3		D5			Division by zero.
	D2						S-program exceeds available memory. An attempt was made to access non-present memory.
	D2	D3					Non-present Interpreter – S-program attempted to execute an S-instruction which is interpreted by an optional segment which was not declared at System Generation.
D1	D2	D3	D4	D5			Serial Printer not available (no power).
D1	D2	D3	D4	D5		D8	Serial Printer Forms Motor Jam (paper could be jammed).
D1	D2	D3	D4	D5	D7		Serial Printer Carrier Motor Jam.

APPENDIX D
USA STANDARD CODE FOR INFORMATION INTERCHANGE

STANDARD CODE

b_7 b_6 b_5 / Bits					$0\,0\,0$	$0\,0\,1$	$0\,1\,0$	$0\,1\,1$	$1\,0\,0$	$1\,0\,1$	$1\,1\,0$	$1\,1\,1$
b_4	b_3	b_2	b_1	COLUMN → / ROW ↓	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	DLE	SP	0	@	P	'	p
0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	BS	CAN	(	8	H	X	h	x
1	0	0	1	9	HT	EM	)	9	I	Y	i	y
1	0	1	0	10	LF	SUB	*	:	J	Z	j	z
1	0	1	1	11	VT	ESC	!	;	K	[	k	{
1	1	0	0	12	FF	FS	'	<	L	\	l	¦
1	1	0	1	13	CR	GS	–	·	M	]	m	}
1	1	1	0	14	SO	RS	·	>	N	^	n	~
1	1	1	1	15	SI	US	/	?	O	—	o	DEL

APPENDIX E
S INSTRUCTIONS

This appendix lists which S-instructions are implemented in each segment of the SL3/5 Interpreter.

BASIC SEGMENT

SL3

ADA	ADIR	ADK	ADM	AL	ALARM	ALR
ALTO	AR	ARTO	BRU	CC	CHG	CLA
CLM	CPA	DIR	DIV	EAM	EX	EXE
EXL	EXZ	IIR	INK	LIR	LKBR	LLCR
LLLR	LOD	LPKR	LPNR	LRCR	LRLR	LSR
MOD	MUL	MULR	NK	NKCM	NKR	NKRCM
NOP	OC	OFF	PA	PAB	PC	PC+
PC–	PCP	PEND	PKA	PKB	PKC	PN
PNS+	PNS–	POF	PON	POS	REM	RR
RST	SET	SK	SKE	SKL	SKZ	SLRO
SLROS	SRJ	SRR	STOP	SUA	SUK	SUM
TAIR	TIRA	TK	TKM	TRA	TRM	

SL5

ADA	ADIR	ADK	ADM	AL	ALARM	ALR
ALTO	AR	ARTO	BRU	BRUIA	CC	CHG
CLA	CLM	CPA	CPL	CPLA	DIR	DIV
EAM	EDIT	EX	EXE	EXL	EXZ	IIR
INK	LIR	LKBR	LLCR	LLLR	LOD	LPKR
LPNR	LRCR	LRLR	LSR	MOD	MUL	MULR
NK	NKCM	NKR	NKRCM	NOP	OC	OFF
PA	PAB	PC	PC+	PC–	PCP	PEND
PKA	PKB	PKC	PN	PNS+	PNS–	POF
POFF	PON	POS	REM	RR	RST	SET
SK	SKE	SKL	SKZ	SLRO	SLROS	SRJ
SRR	STOP	SUA	SUK	SUM	TAIR	TIRA
TK	TKM	TRA	TRM			

DATA HANDLING 1 SEGMENT

SL3

IRCP	LRBR	RCP	SCP	TRAB	TRBA	TRCB
TRF						

SL5

IRCP	LRBR	RCP	SCP	SPRD	TNAB	TNBA
TRAB	TRABS	TRBA	TRABN	TRBAS	TRCB	TRF
TRFL	TRFN					

DATA COMM BASE SEGMENT

SL3

LBN	LGN	LPF	LPR	LRA	LSA	LSN
LTF	LTH	LTN	RBN	RGN	RPF	RPR
RRA	RSA	RSN	RTF	RTH	RTN	TRB
TSB						

SL5

ANDSR	DIAL	LBN	LCR	LDCL	LDRB	LDSB
LGN	LPF	LPR	LRA	LSA	LSN	LSYR
LTF	LTH	LTN	ORRSR	RAUTO	RBN	RETDC
RGN	RPF	RPR	RRA	RSA	RSN	RSYR
RTF	RTH	RTN	SPDC	STDC	STLDC	TRB
TSB	XORSR					

CASSETTE I/O SEGMENT

ERASE	FTM	RC	RTM	RWND	RWNDI	SC
SPBK	WC	WCTM	WCV			

CHECK DIGIT COMPUTE/VERIFY SEGMENT

CDV CDC

SOFTWARE SPECIALS

SL5 ONLY – Special routines for S-Utilities

APPENDIX F
FLAG SETTINGS

This appendix explains cause and effects of certain flag group settings within an SL3/5 Environment.

Y AND K FLAGS

The following flags within the flag groups Y and K are set/reset when a (DATA HANDLING 1) transfer of data from the receive buffer to memory is completed and the data transferred is terminated (last character in data string) by the code indicated in the left hand column. These flags can be interrogated by the user program.

1 = SET, 0 = RESET

TERMINATION CODE	Y FLAG GROUP 3	2	1	4
00	NO FLAG ALTERED			
01	0	0	0	1
02	0	0	1	0
03	NO FLAG ALTERED (*)			
04	0	1	0	0
05	0	1	0	1
06	0	1	1	0
07	0	1	1	1
08	1	0	0	0
09	1	0	0	1
0A	1	0	1	0
0B	1	0	1	1
0C	1	1	0	0
0D	1	1	0	1
0E	1	1	1	0
0F	1	1	1	1

TERMINATION CODE	K FLAG GROUP 3	2	1	4
10	0	0	0	0
11	0	0	0	1
12	0	0	1	0
13	0	0	1	1
14	0	1	0	0
15	0	1	0	1
16	0	1	1	0
17	0	1	1	1
18	1	0	0	0
19	1	0	0	1
1A	1	0	1	0
1B	1	0	1	1
1C	1	1	0	0
1D	1	1	0	1
1E	1	1	1	0
1F	1	1	1	1

* T FLAG 0 SET

E(L) FLAGS

The E flag group in a SL5 environment (L flag group in SL3) is 'used' to indicate to the user program error/exception conditions that arise when the user program (via the interpreter) attempts to access a peripheral device (i.e. magnetic tape cassette) with an I/O command.

MAGNETIC TAPE CASSETTE

After an I/O command is given and if an error/exception condition occurs the following flags are set/reset to indicate this condition. The E (L in SL3) flag group should be interrogated by the user program after each I/O command. (See Magnetic Tape Cassette Program Handler for Cobol Programs – Form No. 2006458)

EXCEPTION CONDITION	FLAG SETTING E (L) FLAGS (1 = Set, 0 = Reset)			
	1	2	3	4
WRITE ERROR FATAL	1	1	0	1
NOT READY	0	1	0	1
WRITE ERROR AND EOT	0	1	1	0
WRITE INHIBIT	0	0	1	1
EOT AND TAPE MARK	0	0	1	1
ACCIDENTAL REWIND	1	1	0	0
OP – ALERT	1	0	0	1
READ ERROR AND EOT	1	0	1	0
READ ERROR	1	0	0	0
WRITE ERROR	0	1	0	0
EOT	0	0	1	0
TAPE MARK	0	0	0	1

APPENDIX G
RELATED DOCUMENTATION

The following manuals are related to equipment discussed in this manual:

	Form Numbers
COBOL for Mini-Computers Reference Manual	1068160
AE500 System Software Operation Guide	
AE500 Operators Manual	
Series L/TC Mini-Computers Magnetic Tape Cassette Program Handler for COBOL Programs	2006458

Burroughs

International Technical Information Organization
Publications Remarks Form

Title: Series AE500 Equipment
Reference Manual

Form: 2007332

Date: February 1976

Check Type of Suggestion:

☐ Addition ☐ Deletion ☐ Revision ☐ Error

General Comments and/or Suggestions for Improvement of Publication:

From:

Name _____ Date _____

Title _____

Company _____

Address _____

Cut Along Dotted Line

Staple

Fold Down Second Fold Down

From

Apply Postage

Manager, Technical Documentation
International Technical Information Organization
Burroughs Machines Limited
Cumbernauld G68 OBN
Glasgow, Scotland

Fold Up First Fold Up