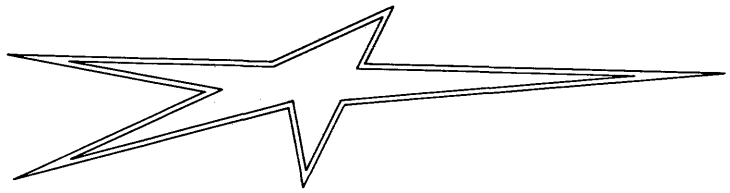
CONTROL DATA
CORPORATION



CONTROL DATA® STAR PERIPHERAL STATIONS
PAGING (DRUM) STATION
DISK STATION
SERVICE STATION
844 STORAGE STATION
MAGNETIC TAPE STATION
UNIT RECORD STATION
DISPLAY/EDIT STATION
STAR-1B SERVICE STATION
STAR-100 SERVICE STATION
STORAGE (MEDIA) STATION
STAR-100 MAINTENANCE CONTROL UNIT
STAR-65 MAINTENANCE CONTROL UNIT
HIGH CAPACITY DISK STATION

HARDWARE REFERENCE MANUAL

NOTICE

ALL REFERENCES IN THIS MANUAL TO CAPABILITIES/ PERFORMANCE OF THE STAR STATIONS AND ASSOCIATED PERIPHERAL DEVICES ARE NOMINAL VALUES AND MAY IN SOME CASES BE THE MAXIMUM CAPABILITIES OF THE DEVICES. THE ACTUAL CAPABILITIES ENCOUNTERED IN A SYSTEM ARE DEPENDENT ON THE PARTICULAR HARDWARE AND SOFTWARE CONFIGURATION USED.

REVISION RECORD

REVISION	DESCRIPTION
01	Preliminary issue.
(6-01-73)	
02	Preliminary issue. Revised to incorporate technical review comments where available.
(7-18-73)	
03	Revised preliminary issue. Pages changed; v, 2-117, 2-119, 2-123, 2-136, 2-141, 2-144,
(9-14-73)	2-148, 2-150, 2-154, 2-161, 2-168, 2-171, 2-176, 3-58 through 3-69, 3-71 through 3-74,
	3-76 through 3-80, A-1, A-3, A-4, A-7, A-13, A-26, A-31, A-53 through A-59.
	Pages added: iv, ix, xiv, 1-1 through 1-55, 3-64.1, 3-80.1 through 3-80.4, A-18.1 through
	A-18.6, C-1 through C-15.
04	Revision packet issued. Pages changed: cover (note), iii, vi, ix, xii, xiii, xvi, 1-1 through
(10-26-73)	1-5, 1-7 through 1-11, 1-18, 1-20 through 1-23, 1-25, 1-28 through 1-37, 1-39 through 1-41,
	1-45, 1-47, 1-48, 1-51 through 1-54, 2-11, 2-11.1, 2-13, 2-16 through 2-18, 2-23 through
	2-25, 2-86, 2-117, 2-157, 3-10, 3-37, A-1 through A-59 (revised and renumbered), C-3, C-4,
	C-6 through C-10, C-12 through C-15. Pages added: iv, xiii, xvi, 3-81 through 3-114,
	4-1 through 4-31, 5-1 through 5-9, B-0 through B-8.
A	Manual released
(1-07-74)	
B	Manual revised; includes ECO 034793 (publication change only). Pages changed: revision
(4-1-74)	record, iii through xii, 1-1 through 1-54, 2-1 through 2-6, 2-9, 2-10, 2-12, 2-14, 2-19,
	2-21, 2-22, 2-25, 2-31, 2-32, 2-35, 2-40, 2-42 through 2-49, 2-52, 2-53, 2-55, 2-65,
	2-67, 2-68, 2-70, 2-71 through 2-74, 2-82, 2-84, 2-85, 2-87, 2-88, 2-89, 2-91 through
	2-106, 2-108, 2-109, 2-110, 2-112, 2-113, 2-114, 2-120, 2-128, 2-135, 2-136, 2-137,
	2-146, 2-147, 2-152, 2-155, 2-156, 2-157, 2-165, 3-1, 3-3, 3-8, 3-9, 3-10, 3-13, 3-17,
	3-19, 3-20, 3-21, 3-28, 3-29, 3-31, 3-33, 3-34, 3-35, 3-38, 3-40, 3-46, 3-48, 3-51
	through 3-56, 3-60, 3-62, 3-68, 3-69, 3-79, 3-94 through 3-100, 3-103, 3-104, 3-105,
	4-1, 4-15, 5-1, 5-6, 5-8, A-1 through A-9, A-11, A-14 through A-19, A-22, A-23, A-24,
	A-27, A-30 A-31, A-32, A-34, A-46 through A-51, A-54, A-55, A-57, A-63 through A-66,
	A-70 through A-79, B-2 through B-9, C-1 through C-4, C-9 through C-15. Add the follow-
	ing pages: 1-55 through 1-59, 2-94.1 through 2-94.3, 3-106 through 3-109, 5-9 through 5-21,
	A-81 through A-90.

Publication No.
60405000

© 1973, 1974
by Control Data Corporation

Printed in the United States of America

Address comments concerning this manual to:

Control Data Corporation
Technical Publications Department
4201 North Lexington Avenue
Arden Hills, Minnesota 55112

or use Comment Sheet in the back of this manual.

PREFACE

This manual contains reference information for the system functions of the CONTROL DATA® STAR Peripheral Stations, descriptions of the station hardware, and some typical procedures necessary to exercise the programmable elements. The STAR stations are a set of peripheral equipment designed for use with a CDC® STAR 100 Computer, and as such, are implemented to some degree by common hardware and software. Other stations may be used in a STAR peripheral system; manuals providing information on this equipment are listed in this preface.

NOTE

Throughout this manual, the abbreviations NOC and NIC denote normal output channel and normal input channel, respectively. As an example of its use in text, the notation NOC-5 indicates normal output channel 5.

Control Data Publications	Publication No.
†STAR-100 Computer Reference Manual	60256000
†STAR-100 Computer Hardware Maintenance Manuals: General Description, Operation, Theory of Operation, Installation and Checkout, Maintenance	60256100
Control Data Large and Medium Scale Computer Systems Site Preparation Manual Section 1 - General Information	60275100
Control Data STAR-100 Computer System Site Preparation Manual Section 2 - System Data	60381600
†STAR-100 Computer Refrigeration System Customer Engineering Manual	60329800
†STAR-100 Peripheral Stations Customer Engineering Manual (Maintenance)	60325300
STAR-100 Peripheral Stations Customer Engineering Diagrams Manuals	
†Station Buffer Unit Core Control	60382000 and 60406700
†Station Buffer Unit Interfaces	60382100 and 60406800
†Station Control Unit	60362900
†Station Display Unit	60382500

†This manual is available on a controlled distribution basic only from:
Control Data Corporation
Advanced Systems Lab, Publications Dept.
4201 No. Lexington Ave.
Arden Hills, Minn. 5512
All manuals not having the dagger are available from:
Literature and Distribution Services
8100 34th Ave. So., P.O. Box O
Minneapolis, Minn. 55440

Control Data Publication	Publication No.
FR101B Buffer Controller Reference Manual	60275000
3000 Series Computer Systems I/O Specifications Manual	60048800
7600/CYBER 70 Model 76 Computer Systems I/O Specifications	60408700
Buffer Controller Maintenance Console Customer Engineering Manual (TF201-A03)	58032700
Buffer Controller Maintenance Console II Customer Engineering Manual (TF204-A01)	59344000
CC102-A Display Station Reference/Customer Engineering Manual	82160600
CC601-A/B Display Station Reference/Customer Engineering Manual	82134300
CA115-A Entry Keyboard Reference/Customer Engineering Manual	82160700
405 Card Reader Reference/Customer Engineering Manual	40809300
3649-A Card Reader Controller Reference/Instruction Manual	60097900
415 Card Punch Maintenance Manual	40819600
512-1 Line Printer Reference/Customer Engineering Manual	44980100
3555-1 Line Printer Controller Reference Manual	60231300
HR-600 Line Printer Customer Engineering Manual	44983100
657-1, 2, 3, 4 Magnetic Tape Transports Reference/Customer Engineering Manual	70600200
659-1, 2, 3, 4 Magnetic Tape Transports Reference/Customer Engineering Manual	70601000
841 Multiple Disk Drive Maintenance Manuals	41243500
844 Disk Storage Unit Customer Engineering Manual	70629500
854 Disk Storage Drive Maintenance Manual	41245800
865 Drum Storage Unit Customer Engineering Manual (Modified 865)	60320900
7638-1 Disk File Subsystem Reference Manual	60265500
3446-A/B Card Punch Controller Reference Manual	60332100
3447-A Card Reader Controller Reference Manual	60332300
3528 Magnetic Tape Controller Reference Manual	60287600
817 Disk File Customer Engineering Manual	41242300

CONTENTS

FIGURES

FIGURES

TABLES

TABLES

STAR PERIPHERAL STATION INTERFACE CONFIGURATION CHART

	Station / Type of Interface	Paging (Drum) Station	Storage (Media) Station	844 Storage/ Magnetic Tape Station	844 Service Station	STAR 100 Service Station	STAR 1B Service Station	Disk Station	Unit Record Station	Display/ Edit Station	STAR 100 Maintenance Control Unit	STAR 65 Maintenance Control Unit
SCU Interfaces	Microdrum	1	1	1	1	1	1	1	1	1	1	1
	Keyboard	1	1	1	1	1	1	1	1	1	1	1
	CRT	1	1	1	1	1	1	1	1	1	1	1
	SBU Coupler	1	1	1	1	1	1	1	1	1	2	2
	Normal Channels	1	1	1	1	1	1	1	1	1	1	1
	SCU Gate	1	1	1	1	1	1	1	1	1	1	1
	SDU Coupler									1		
	3000 SCU								2		1	1
	854 SCU										1	1
	HR 600										1	1
	Normal Channel Switch										1	1
	STAR-100										1	
	STAR-65											1
SBU Interfaces	3000 I/O		2	2								
	7000 I/O	2				1		2				
	841 MDD		2				2					
	844 DSD			2	1							
	STAR A Data Chans				4	8	6					
	SAC Coupler	1	2	2	2	2	2	2				
	Communication Network Trunk				2							
	Index Compare	1					1					

GENERAL DESCRIPTION 1

GENERAL

Section 1 contains an introduction to the CDC STAR Peripheral Stations. It includes a description of each station and its function, a brief description of each functional element within the station, and a physical description of the station hardware.

PERIPHERAL STATION/COMPUTER SYSTEM OVERVIEW

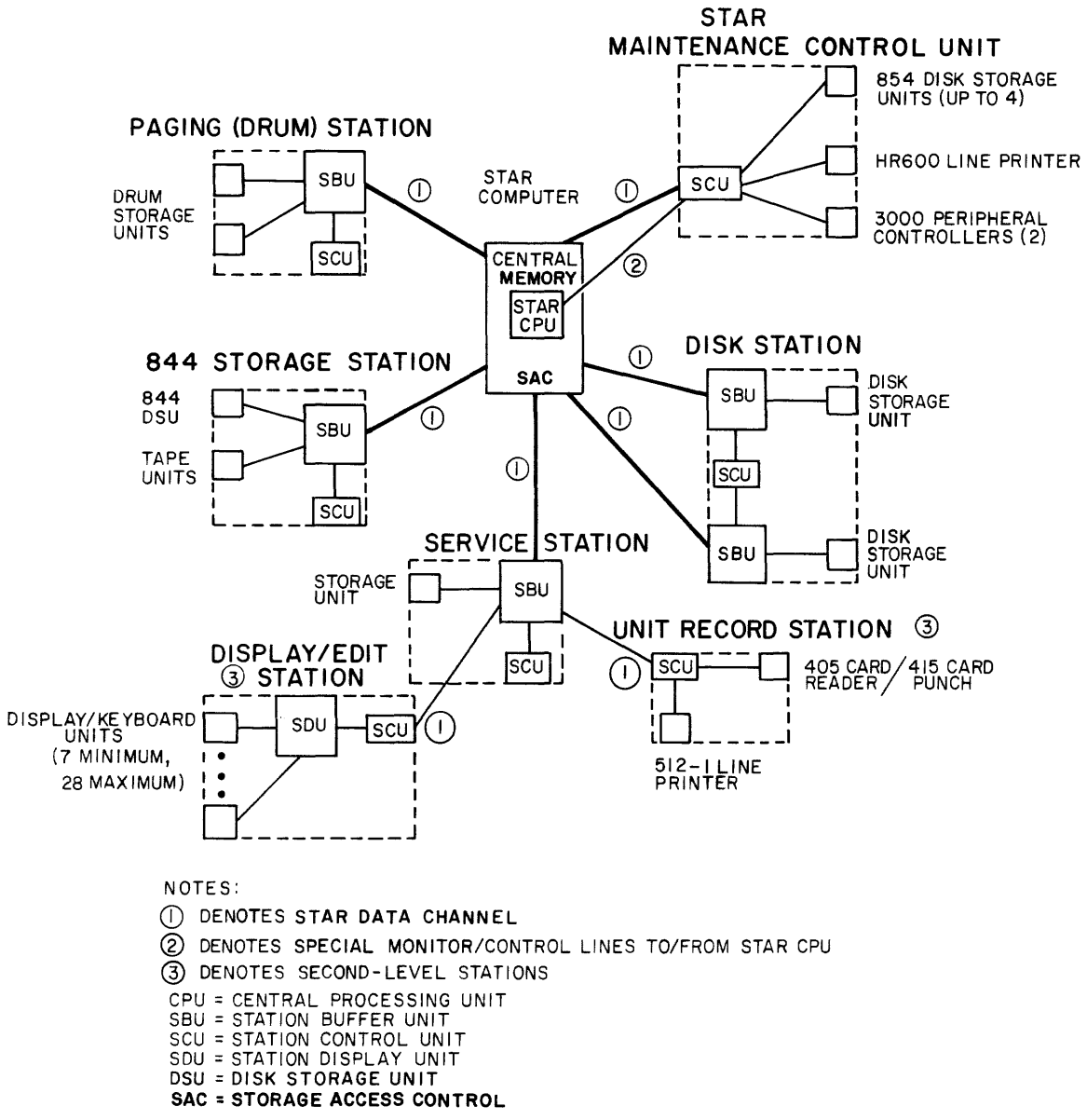
Peripheral stations perform such tasks as detailed control of peripheral equipment, recovery and retransmission of corrected data, manipulation of records and messages as to timing and routing, and the general management of available peripheral resources.

The standard peripheral stations available to handle I/O tasks in STAR computer systems are:

- Paging (Drum) Station
- Disk Station
- 844 Service Station
- STAR-100 Service Station
- STAR-1B Service Station
- 844 Storage Station
- Magnetic Tape Station
- Unit Record Station
- Display/Edit Station
- Storage (Media) Station
- STAR-100 Maintenance Control Unit
- STAR-65 Maintenance Control Unit

Figure 1-1 illustrates the relationships of some of these standard stations to the central processor and the memory of a STAR-100 Computer.

Peripheral stations are classified as first level or second-level stations according to the position they occupy in the overall system hierarchy. This hierarchy is physical as well as logical in that an I/O data stream is staged twice from second-level stations and once from first-level stations on the way into and out of the STAR central processor. This procedure is necessary since many of the I/O operations usually required of the central processor have been reassigned to one or more levels in the hierarchy. First-level stations connect directly to central memory through STAR storage access control (SAC). Second-level stations (Figure 1-1) communicate indirectly with central memory through a first-level service station. The first-level service station acts as a concen-



NOTES:

① DENOTES STAR DATA CHANNEL

② DENOTES SPECIAL MONITOR/CONTROL LINES TO/FROM STAR CPU

③ DENOTES SECOND-LEVEL STATIONS

CPU = CENTRAL PROCESSING UNIT
SBU = STATION BUFFER UNIT
SCU = STATION CONTROL UNIT
SDU = STATION DISPLAY UNIT
DSU = DISK STORAGE UNIT
SAC = STORAGE ACCESS CONTROL

Figure 1-1. Typical System Configuration, STAR Peripheral Stations

trator which multiplexes several low-speed stations into a single central memory channel. The data path between all first-level stations and central memory is a STAR data channel. This data channel is also used between second-level stations and a service station. In the STAR-100/STAR-65 maintenance control units (MCU) only, a second group of lines connect a control interface in the MCU with the computer system's central processor.

STATION ORGANIZATION

The typical peripheral station consists of a station control unit (SCU), station buffer unit (SBU), functionally related peripheral devices, and the necessary logic to form the required interfaces. Every standard station contains an SCU but not necessarily an SBU. Normally, first-level stations contain both an SCU and an SBU while second-level stations contain only an SCU. Therefore, there are two major types of peripheral stations, SCU/SBU stations and SCU-only stations. A simplified version of the paging (drum) station (Figure 1-2) is typical of SCU/SBU stations. In this type of station, the SBU acts as a buffer for data being exchanged between I/O devices and the STAR computer or other higher level processor. The SCU controls the peripheral interfaces but I/O data does not flow through the SCU. Some SCU/SBU stations provide for the addition of an optional second SBU. Physically, the SCU and SBU are separate units.

A typical SCU-only station is illustrated in Figure 1-3. In this type of station, interface logic for the I/O devices connects directly to the SCU processor and is physically located in the SCU cabinet. The SCU provides buffer storage for the I/O data as well as controlling the peripheral interfaces.

An SCU contains a small processor called the buffer controller (BC) and a memory. A program, resident in SCU memory, governs data transfers to and from the I/O peripheral devices. The presence of a BC in each station allows the station to perform many of the system I/O-related computing functions leaving the central processor free to complete other tasks.

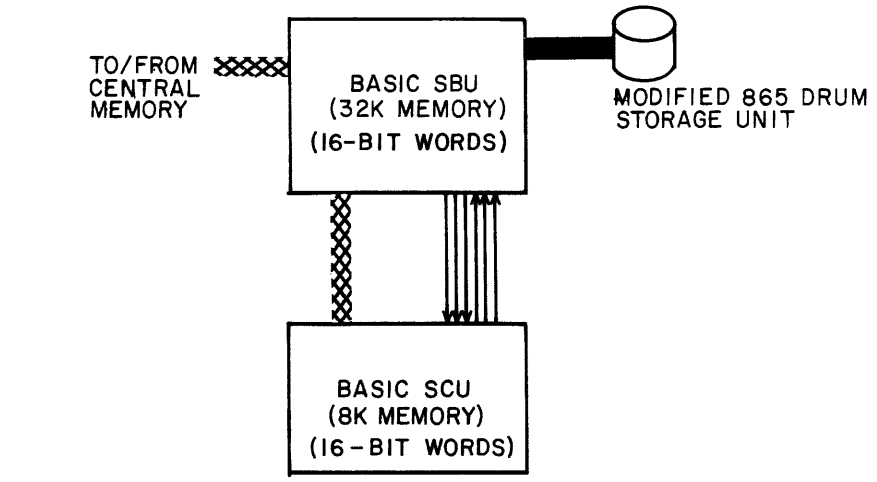


Figure 1-2. Typical SCU/SBU Station

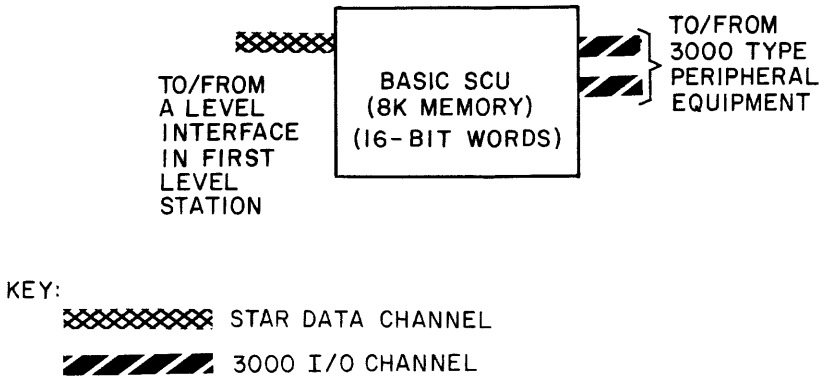


Figure 1-3. Typical SCU-Only Station

The SBU is basically a 32K- 16-bits per word-core memory with the peripheral interfaces that allow I/O devices to transfer data directly into or out of memory. Control lines from the SCU are used to activate and monitor the various peripheral interfaces. In many cases, several peripheral interfaces can transfer data concurrently. To transfer a block of data to the central processor memory from an I/O peripheral device, the data is first stored in SBU memory and then transferred to central memory via a STAR data channel. Similarly, to transfer a block of data to an I/O device, data is first stored in SBU memory and then transferred to the proper I/O device.

Differences in the varieties of SCUs and SBUs used in the various stations occur primarily in the peripheral interface areas.

STATION CONFIGURATIONS

Figures 1-4 through 1-14 illustrate the major components in each station and methods of interconnection between peripheral equipment and SBUs.

NOTE

The following illustrations contain A and B data channel interfaces, which are elements associated with the STAR data channel and are explained in detail in appendix C.

Also, the term peripheral A interface is used to differentiate the logic connected to the B interfaces of other stations from the A interface which connects to the B interface in the SCU.

STATION DESCRIPTIONS

The following paragraphs describe the functions and logical configurations (building blocks) of the various stations and their interfaces. Most of these building blocks are used in more than one station. The Configuration Chart preceding section 1 lists the types and quantities of interfaces in each station. Figures 1-4 through 1-16 illustrate their relationship in each station.

Since the basic SBU and SCU are both common to many stations, a general discussion of each follows Figure 1-14 and precedes the station descriptions.

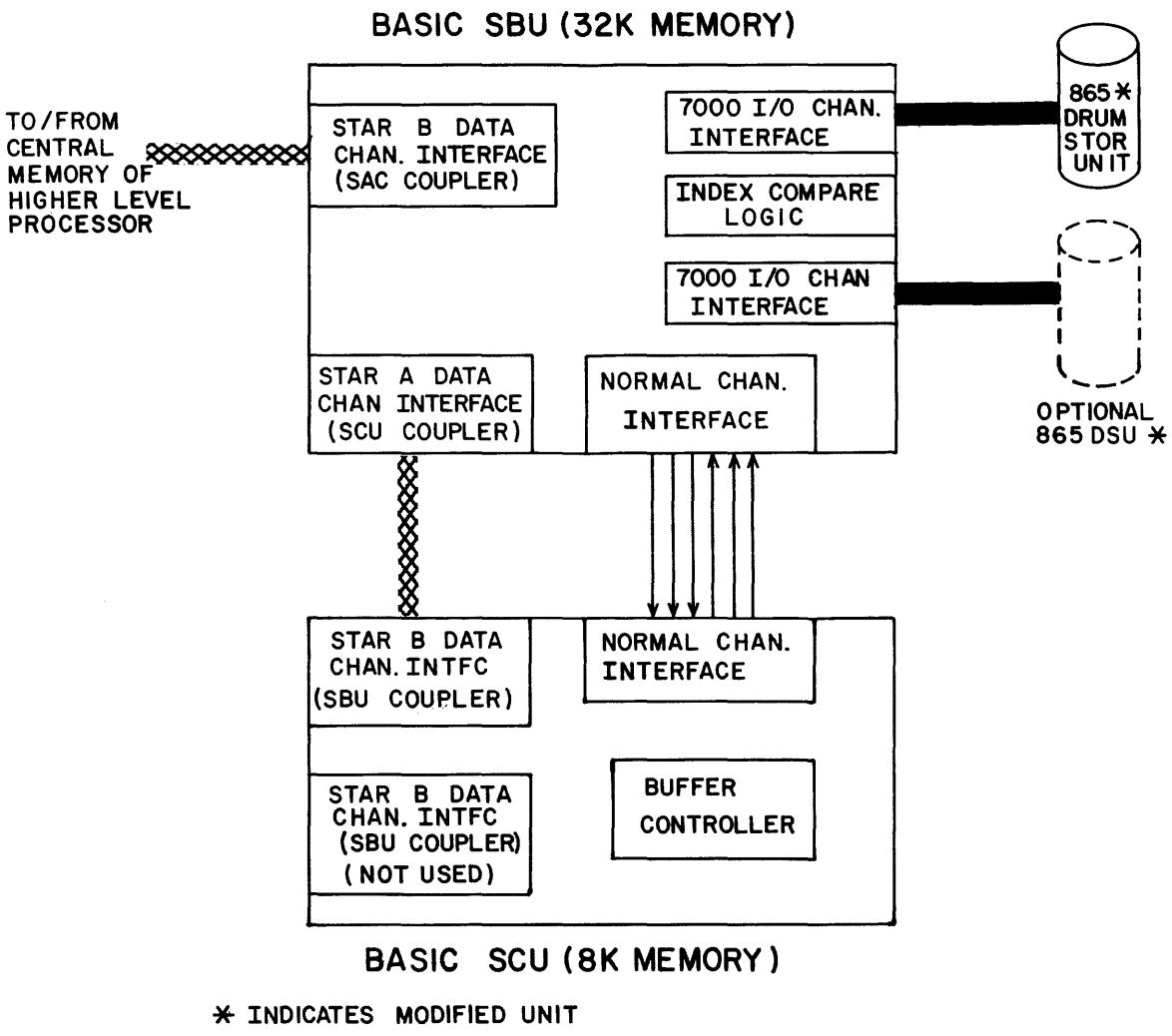


Figure 1-4. Paging (Drum) Station, Typical Configuration

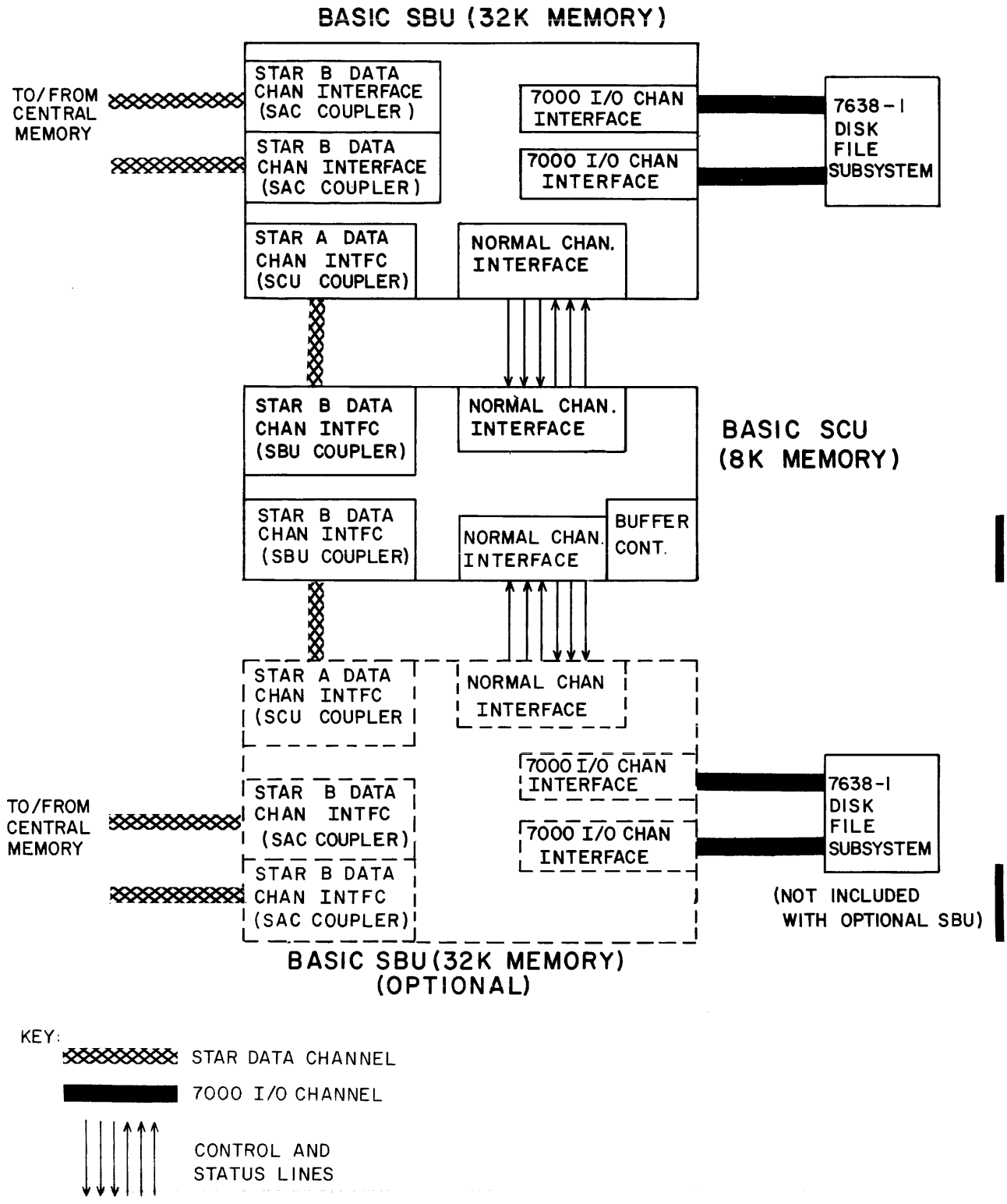


Figure 1-5. Disk Station, Typical Configuration

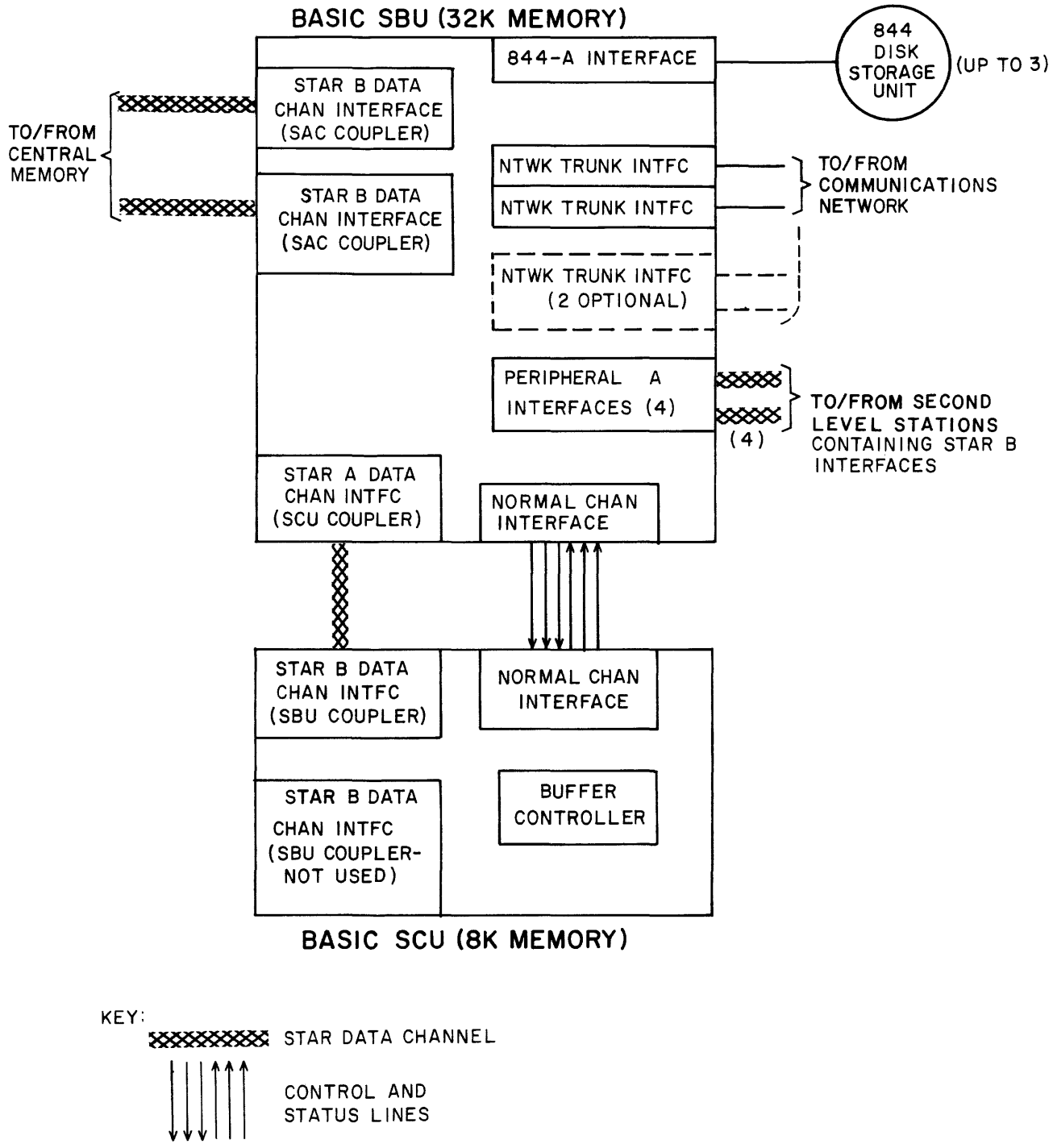
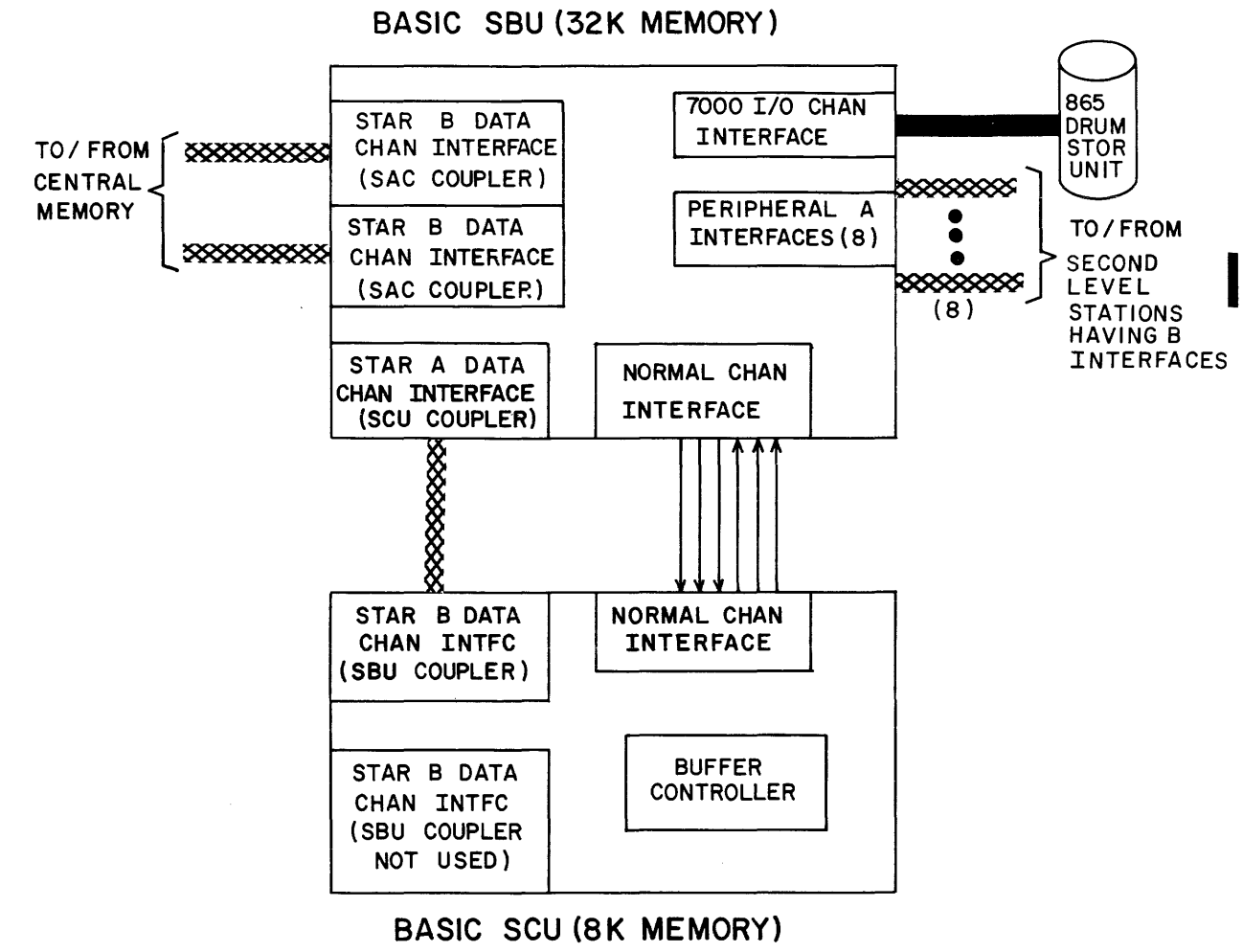


Figure 1-6. 844 Service Station, Typical Configuration



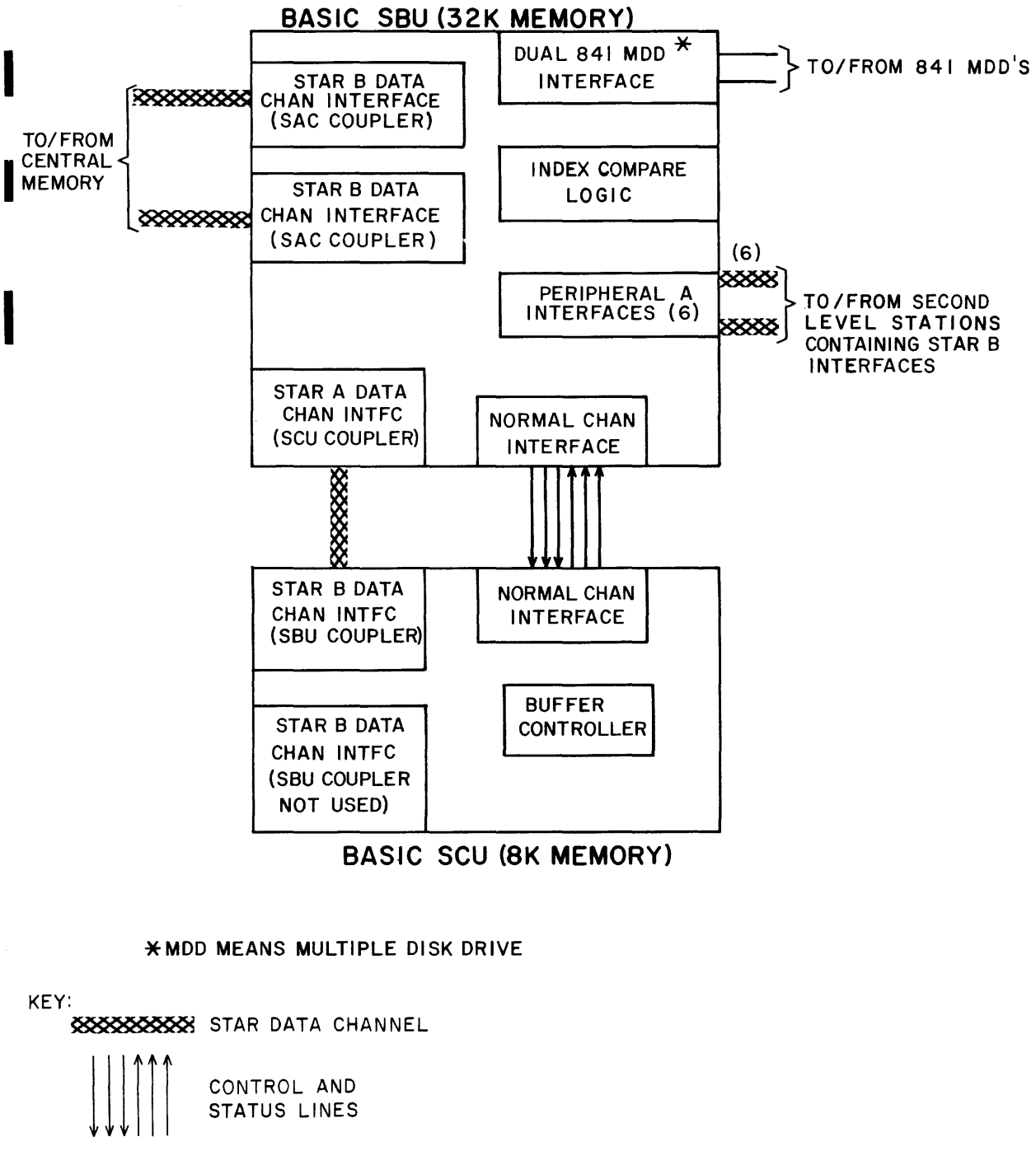
KEY:

STAR DATA CHANNEL

7000 I/O CHANNEL

CONTROL AND STATUS LINES

Figure 1-7. STAR-100 Service Station, Typical Configuration



*MDD MEANS MULTIPLE DISK DRIVE

KEY:

STAR DATA CHANNEL

CONTROL AND
STATUS LINES

Figure 1-8. STAR-1B Service Station, Typical Configuration

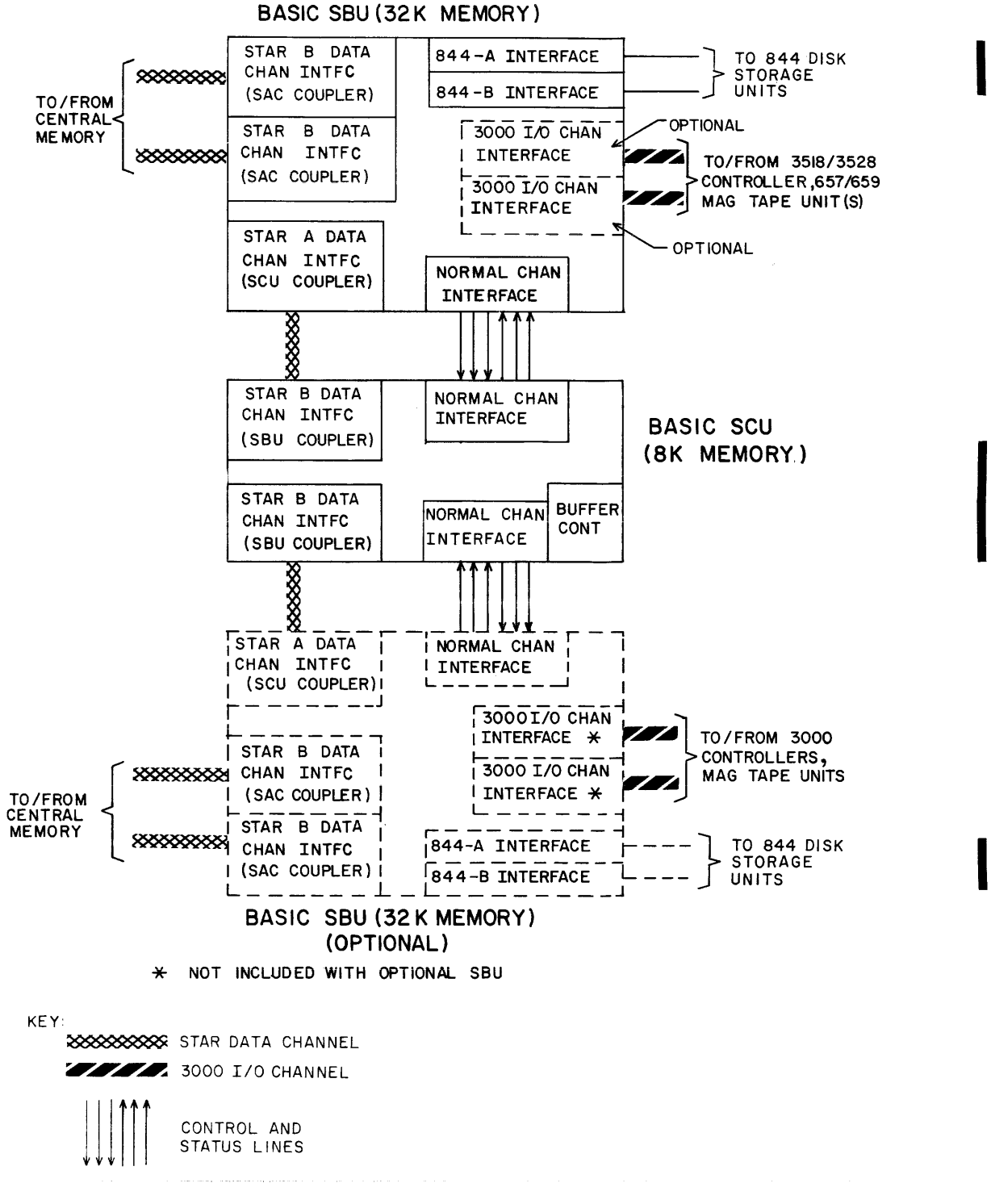


Figure 1-9. 844 Storage Station, Typical Configuration

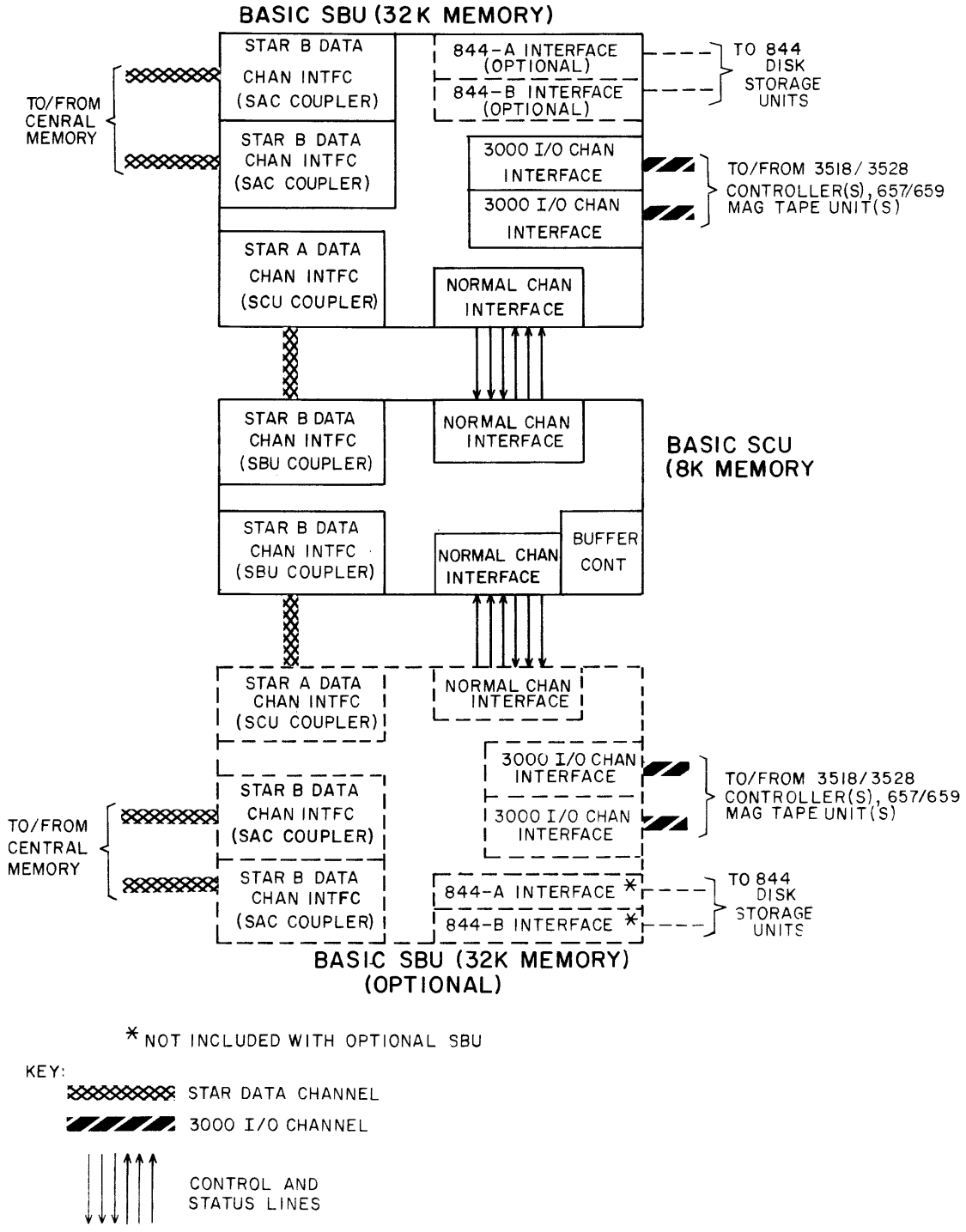
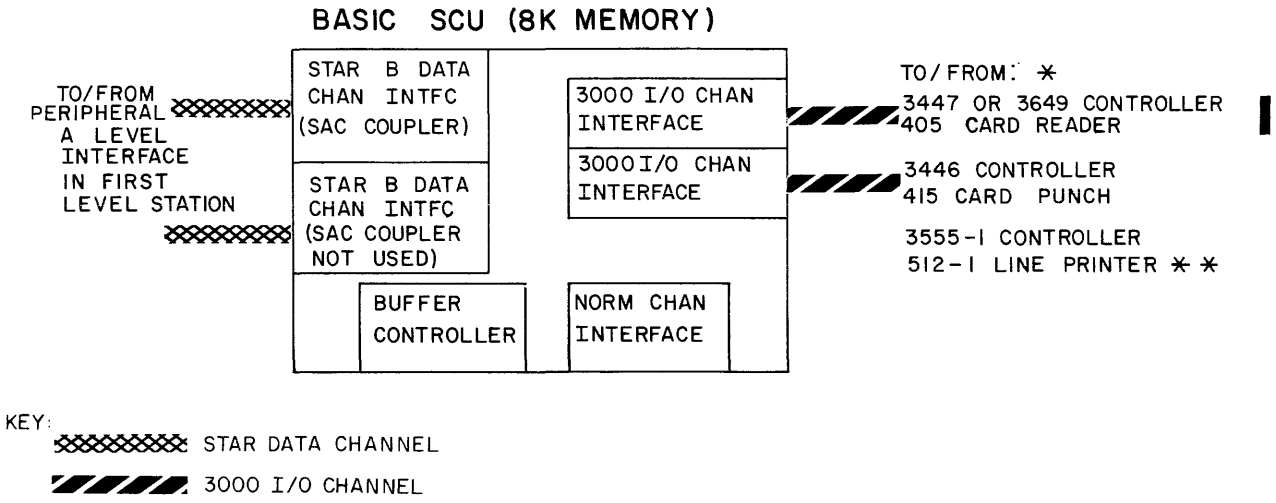


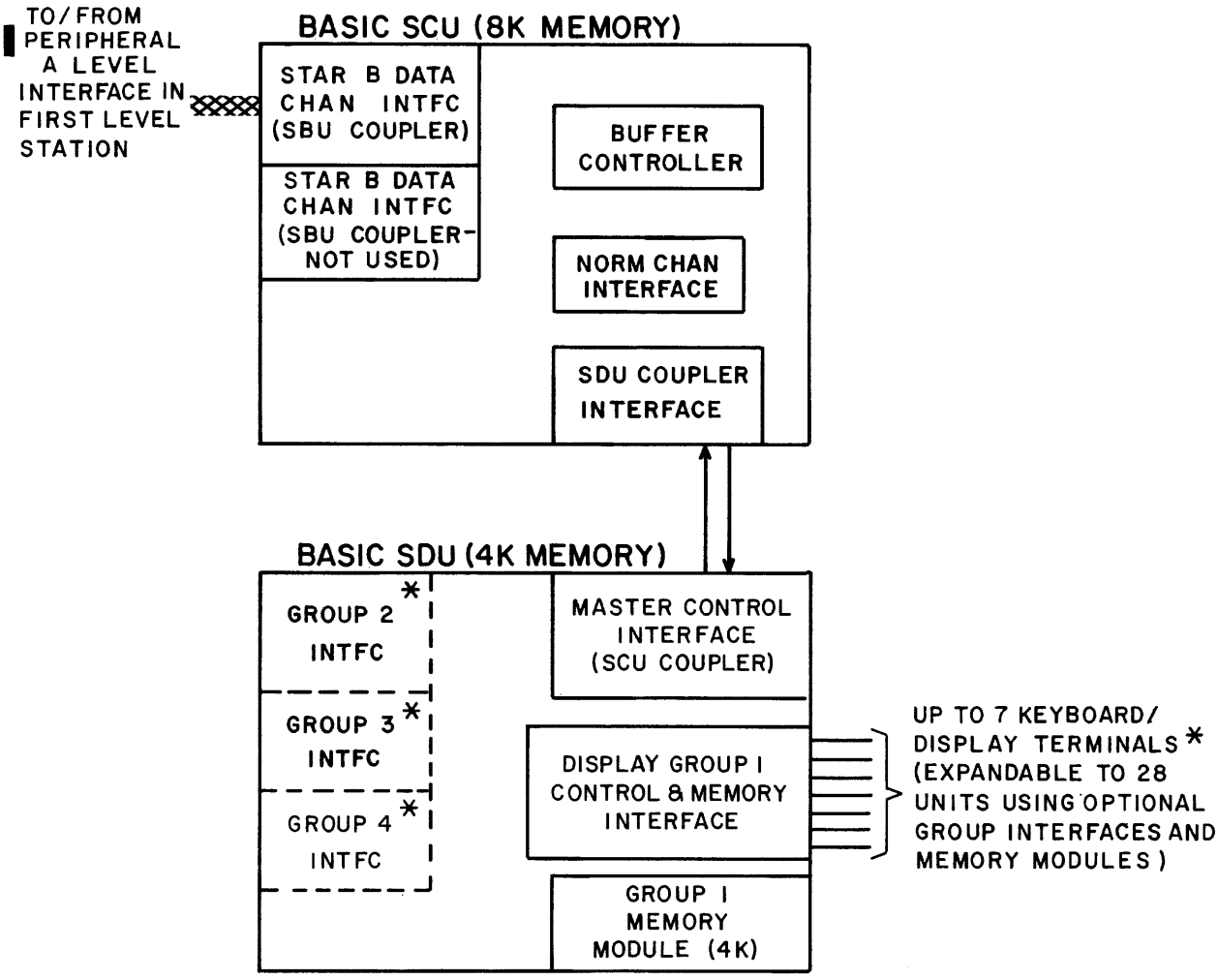
Figure 1-10. Magnetic Tape Station, Typical Configuration



* PERIPHERAL EQUIPMENT NOT INCLUDED WITH BASIC SCU

** SPECIAL 96-CHARACTER ASCII TRAIN CARTRIDGE AVAILABLE FOR LINE PRINTER.

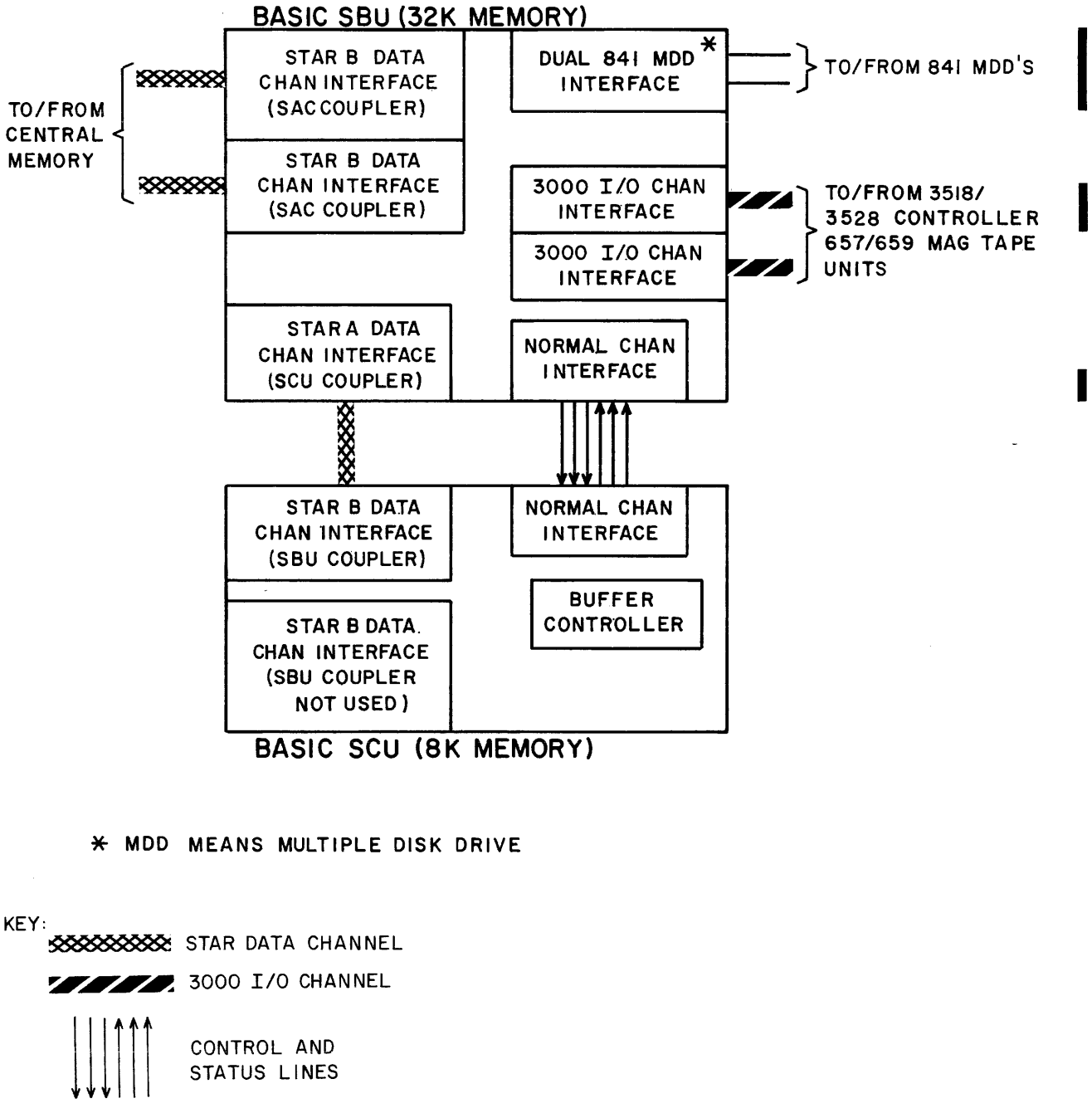
Figure 1-11. Unit Record Station, Typical Configuration



* THESE ITEMS NOT SUPPLIED WITH BASIC SDU.
OPTIONAL KEYBOARD/DISPLAY TERMINALS FURNISHED WITH 200-FOOT SIGNAL CABLES.

KEY:
STAR DATA CHANNEL

Figure 1-12. Display/Edit Station, Typical Configuration



* MDD MEANS MULTIPLE DISK DRIVE

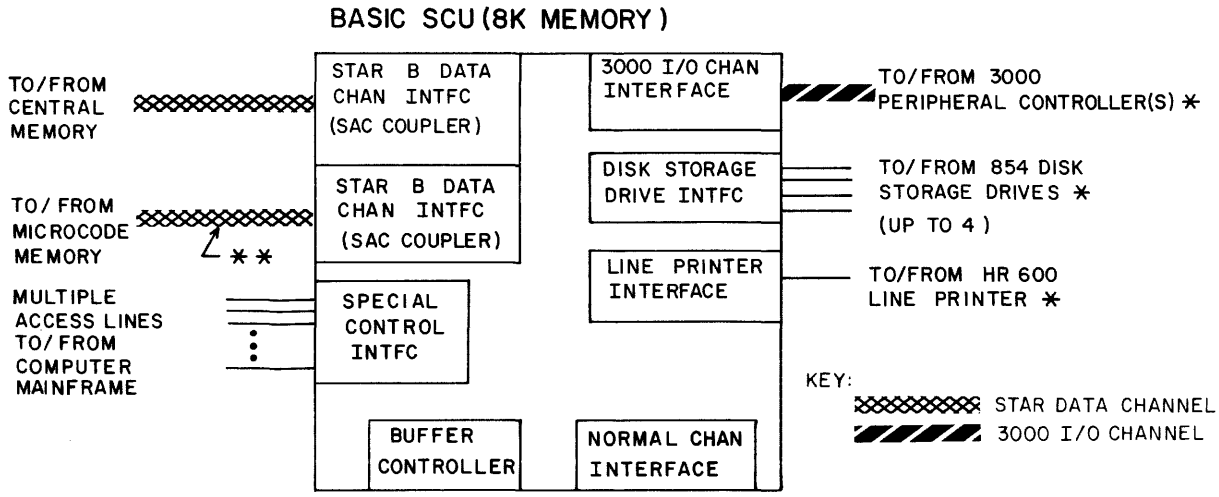
KEY:

STAR DATA CHANNEL

3000 I/O CHANNEL

CONTROL AND STATUS LINES

Figure 1-13. Storage (Media) Station, Typical Configuration



* THESE ITEMS NOT SUPPLIED WITH THE BASIC SCU.
** DENOTES A 16-BIT CHANNEL SIMILAR TO A STAR DATA CHANNEL.

Figure 1-14. STAR-100 Maintenance Control Unit, Typical Configuration

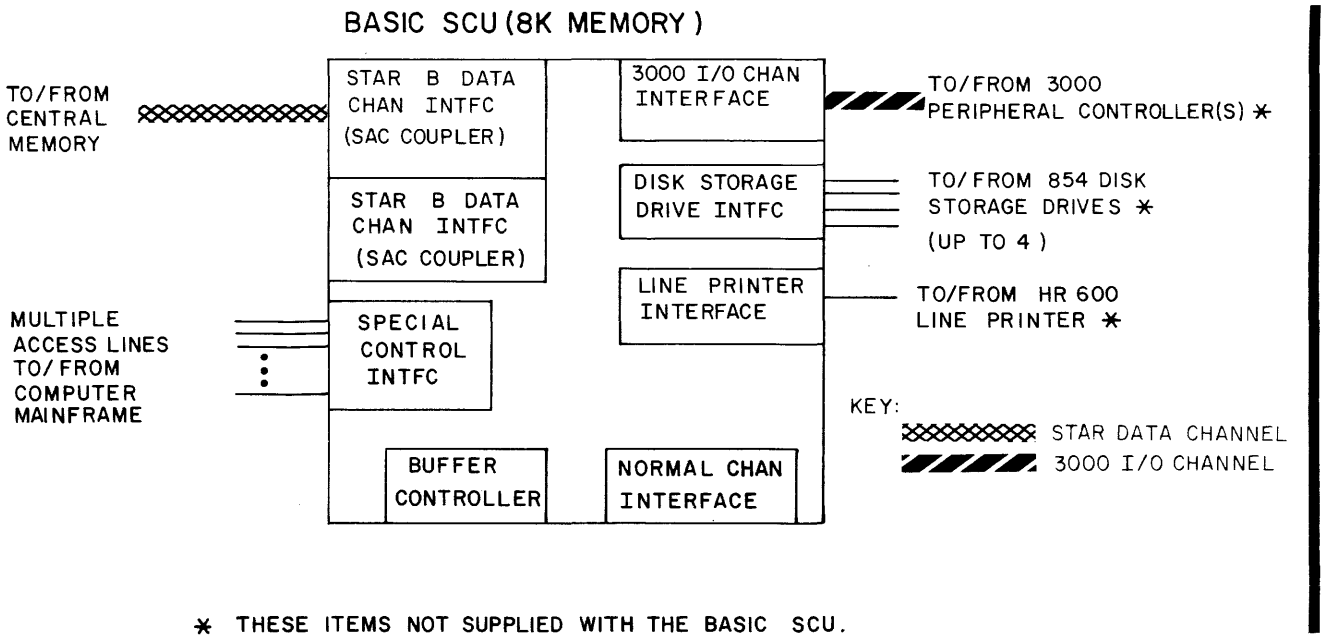


Figure 1-15. STAR-65 Maintenance Control Unit, Typical Configuration

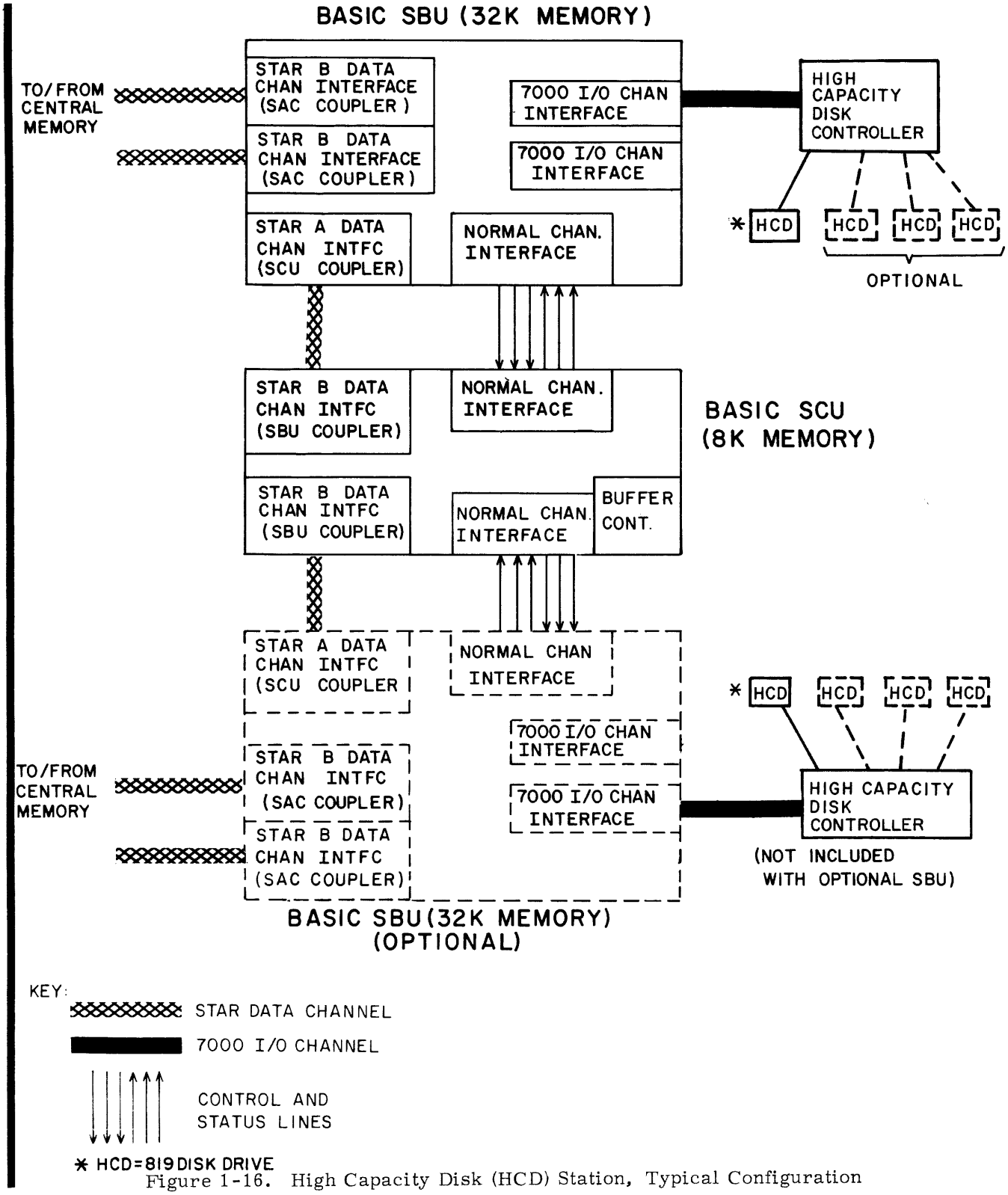


Figure 1-16. High Capacity Disk (HCD) Station, Typical Configuration

STATION BUFFER UNIT

The SBU (Figure 1-17) consists primarily of core memory for intermediate storage of data between the STAR-100 CPU and peripheral devices or other peripheral stations. It also contains core control logic which provides 12 access points in and out of core memory for the various peripheral interfaces. This core memory contains 32,768 16-bit words organized into eight phased banks. The cycle time of each bank is 1.1 microseconds.

Typically, the SBU is used for transferring large amounts of data at high speeds, staging of files and jobs, temporary storage of data and messages, and expansion of the system to subordinate level stations. Therefore, the SCU is relieved of heavy data transfer tasks and is available for higher level control activities even though it controls the SBU and monitors all of its data flow. The SBU is treated as an external memory by the SCU and is not directly addressable by the BC within the SCU. Each functional section within the SBU is discussed in the order listed.

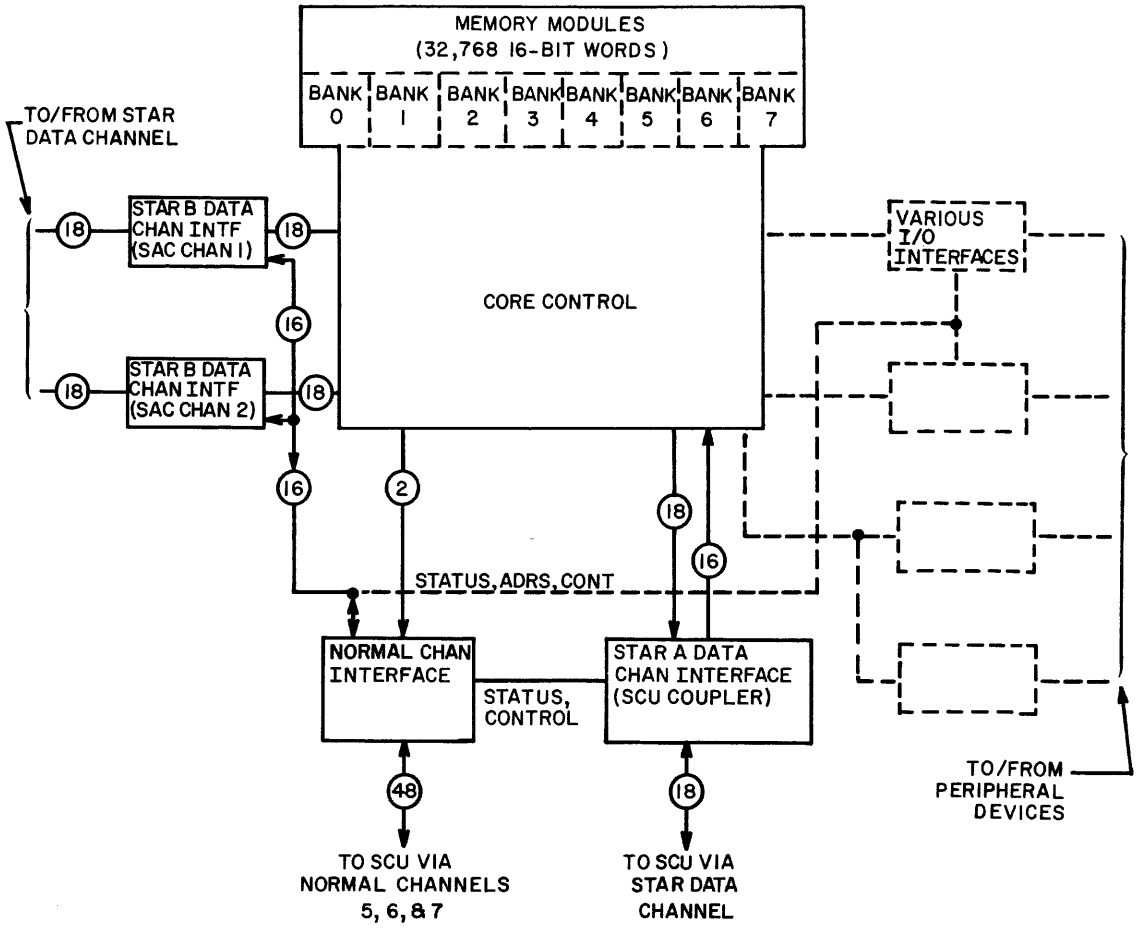
- STAR A data channel interface (SCU coupler)
- STAR B data channel interface (SAC coupler)
- Normal channel interface
- Core control
- Various I/O interfaces

STAR A DATA CHANNEL INTERFACE (SCU COUPLER)

The STAR A data channel interface provides an interface between an access to SBU core and an SCU via the STAR data channel. The A data channel interface is passive, with transfer, control, and initiation being directed by the STAR B data channel interface (active end) channel.

STAR B DATA CHANNEL INTERFACE (SAC COUPLER)

The STAR B data channel interface provides the connection between an access to SBU core and a STAR data channel B end. The B end of the STAR data channel is the controlling end. Normally, the B end is connected to a STAR data channel provided by the SAC of a STAR computer. The B end controls transfers, specifies addresses for the A end, and handles fault conditions.



NOTES:
INTF = INTERFACE
SAC CHAN = STORAGE ACCESS CONTROL CHANNEL

Figure 1-17. Station Buffer Unit Functional Block Diagram

NORMAL CHANNEL INTERFACE

The normal channel interface logic provides an interface in the SBU between the SCU buffer controller normal channel bits and the various interface logic blocks in the SBU. The normal channel bits are used to transmit data, control information to the SBU interfaces, and read in status bits from the SBU interfaces.

This logic block interfaces six 16-bit BC normal channels, three input channels, and three output channels. It contains transmitters, receivers, data fanin and fanout logic, and a decoding section.

CORE CONTROL

The core control logic provides 12 access channels to eight modules of core memory having a total of 32,768 16-bit words of memory. Core memory is phased between modules, each module having independent read/write capability. The maximum memory reference rate is eight references per full memory scan of 1.1 microsecond. The core control logic determines channel priorities, increments channel addresses, checks for page boundaries, controls transfer of data and addresses to memory modules, and provides external timing signals.

I/O INTERFACES

Various types of I/O interfaces are used to enable the transfer of data to or from peripheral devices and an HLP or between second-level stations having B interfaces and an HLP. With one exception (the network trunk interface) all I/O interfaces are controlled by the BC in the SCU and provide status conditions to the BC upon request.

STATION CONTROL UNIT

The SCU (Figure 1-18) is composed of a number of functionally related devices described in the order listed.

- Buffer controller and 1.1-microsecond memory
- Normal channels
- Normal channel interface
- SCU gate
- STAR B data channel interface (SBU coupler A/B)
- Drum/display logic, keyboard, and microdrum

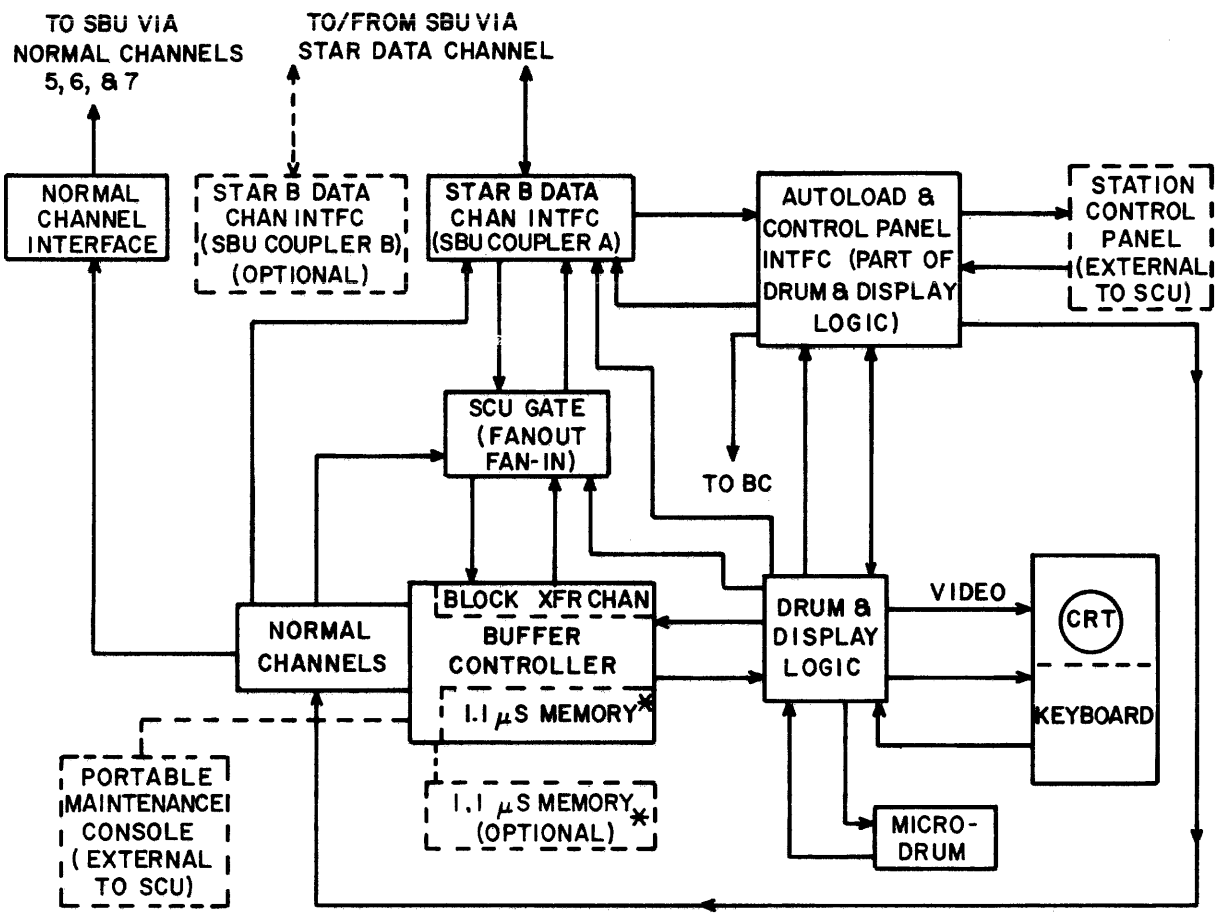
BUFFER CONTROLLER

The BC is the primary control element of any station. It is an internally programmed parallel mode, digital computer having 4096 16-bit words of core memory with a cycle time of 1.1 microseconds. The core memory is expandable to 8K words. The instruction set of the BC was specifically chosen to be compatible with peripheral device-related tasks. The BC provides one 16-bit parallel block transfer channel for high speed data transfer. It also provides up to 512 individually programmed normal channel lines for lower speed data transfer and for peripheral device and station control.

An optional 8K semiconductor memory with a 200-nanosecond cycle time is available in place of the standard 1.1 microsecond core memory in the STAR station control units but is software supported only in the paging and disk stations.

NORMAL CHANNELS

The SCU contains eight normal input channels and eight normal output channels, each of which is 16 bits wide. Channels 0 through 4 are used for the transfer of data and control functions between the programmable elements of the SBU and SCU. Channels 5 through 7 are used for similar functions (data and control) between the various I/O SBU interfaces and the SCU. All normal channel bit assignments for all channels appear in appendix A of this manual.



* 200 NANOSECOND SEMICONDUCTOR MEMORY ALSO AVAILABLE AS AN OPTION ON SPECIAL REQUEST FOR PAGING AND DISK STATIONS.

Figure 1-18. Station Control Unit Functional Block Diagram

NORMAL CHANNEL INTERFACE TO THE SBU

The normal channel interface transmits and receives the BC normal channel bits to and from the SBU. This block contains only receivers and transmitters; no logical operations are performed at this interface. It interfaces three 16-bit output channels and three 16-bit input channels.

SCU GATE

The SCU gate provides a fanin/fanout of four on the BC block transfer channel and permits up to four couplers to be connected to the BC. Two BC normal channel bits control the selection.

STAR B DATA CHANNEL INTERFACE ((SBU) COUPLER A/B)

The SBU STAR data channel coupler matches the BC block transfer channel to a STAR data channel. It is the B end of the STAR channel. This coupler provides features for drum and autoloading (refer to Drum/Display Logic). Only one coupler in a station is used for autoloading and is actually connected to the autoload controls and the drum data path.

The operation of the coupler is controlled by BC normal channel bits, or during autoload, by the autoload logic.

DRUM/DISPLAY LOGIC

This logic is the standard control logic provided in each SCU. It contains several sections: microdrum logic, display logic, keyboard interface, autoload package, and deadman logic.

MICRODRUM LOGIC

Data is recorded serially on the microdrum. The data read/write circuits contain the logic to encode the data, a write driver for recording on the drum, a read preamplifier, and a discriminator to decode the data. Write data is transmitted to the microdrum logic as 16 parallel bits from a BC normal channel. This data is received by a 16-bit shift register which shifts it serially to the read/write electronics.

Read data is transmitted serially from the read/write electronics to the same 16-bit shift register. It is then transmitted as 16 parallel bits via a holding register to a BC normal channel. The output of this holding register is also routed to the coupler used for autoload. Read data from the shift register is also transmitted as 16 parallel bits to a holding register for display output.

KEYBOARD TRANSMIT/RECEIVE LOGIC

The interface between the SCU keyboard and the BC normal channel bits consists of transmitters and receivers. The only exception is the additional input to the keyboard alert buzzer and indicator from the deadman logic.

During a write operation on a display track, the data being written is shifted end-around in the shift register and then transmitted to the display holding register. This procedure allows a display image to be modified and immediately displayed without disturbance or flicker.

DISPLAY LOGIC

The SCU display unit is refreshed from the microdrum. It receives a composite video signal from the display logic. The selected microdrum data track is always being read and displayed when the display mode is selected. When the data mode signal is transmitted from the BC normal channel bit, the display screen is blanked.

AUTOLOAD PACKAGE

A section of the SCU drum/display logic controls autoload operations. The SCU is autoloaded from the microdrum. The autoload sequence causes a track of microdrum data (one of four selected by the switches on the SCU control panel) to be transmitted to BC core and executed. The autoload sequence is initiated from the control panel or by the receipt of an external flag signal from a higher level data channel interface.

DEADMAN LOGIC

The deadman logic provides detection of hang-up conditions in the BC instruction execution or a block data transfer. During each revolution of the microdrum, an index mark sets a flip-flop (FF) in the deadman logic. This FF must be cleared by an output from a BC normal channel bit. If a drum index mark is received and the FF is still set, the BC is stopped, and the alert indicator and buzzer on the SCU display are activated.

A start block transfer signal from the active coupler starts a variable delay timeout in the deadman logic. If the block transfer does not terminate in the allotted delay time, a force terminate signal is sent via the coupler to the BC which terminates the block transfer allowing instruction execution to continue. This terminate condition can be sensed by status.

PORTABLE MAINTENANCE CONSOLE

Two similar portable maintenance consoles are available for maintenance and control of the BC in the SCU. Each console has its own method of accessing a BC, of which there are two basic types. General characteristics and basic differences of the two consoles are discussed in the following paragraphs.

One console (the TF201) provides paper tape input (with error checking) and manual switch operation as two means of accessing the BC. Various signals produced by the BC are monitored by indicators on the console, thereby enabling the operator to determine internal operating conditions of the BC. The TF201 provides the following capabilities.

- Data input via punched paper tape
- Register entry and display
- Sweep mode, variable speed
- Step mode by instruction or cycle
- Breakpoint by instruction, operand, or both

Information contained on the paper tape is normally transmitted to the BC under the conditions of the data entry mode. Continuous checking of the data contained on the tape is performed by the error checking logic. Using the reader test function, the data on a paper tape may also be used to test the error checking and tape control logic, with outputs to the BC being inhibited. Therefore, deliberate errors may be inserted into a tape for test purposes without affecting BC operation.

The second console (the TF204) has the same capabilities as the TF201 but uses a different medium for the transfer of information. It uses a cassette magnetic tape unit through which recorded information can be transmitted to the BC. Also, data stored in BC memory can be retrieved and recorded on the cassette. A second means of accessing the BC is through a hexadecimal keyboard. Entries made via the keyboard are displayed in a four-stage hexadecimal readout. Register contents and various signals within the BC are monitored by indicators, and a second four-stage hexadecimal readout allows the operator to determine internal operating conditions of the BC.

PAGING (DRUM) STATION

The paging (drum) station (Figure 1-4) provides memory extension for higher level computer central storage and is dedicated to one central storage/central processing unit (CPU). Storage in the drum station is addressed by the CPU in terms of central memory virtual addresses.

STATION FUNCTIONAL OPERATION

Typically, the drum station may expect the higher level processor (HLP) to issue either a request for a page (2048 16-bit words) of data or request a storage area for a page of data. The SCU polls CPU core storage for the CPU request. When the request is recognized, the SCU processes it to determine the action to be taken. If the requested page resides in SBU memory, the SCU program sends the page boundaries to the SAC coupler interface via the normal output channels. The SCU then initiates the proper type of data transfer (read or write) to or from the HLP. If the requested page is located on the drum, the SCU receives the CPU request in the form of a user key or a virtual address. To translate this information to a physical data page on the 865 drum requires an index compare operation. A table of entries residing in SBU memory consists of user keys, virtual addresses, and physical locations relating to the drum. The index compare interface compares the requested CPU user key or virtual address with the table entries. When a compare is made, the entry is saved and the physical address is read from that entry. This address is used to initiate a data transfer to/from the drum via the 7000 interface.

If the request from the CPU is to store a page on the 865 drum, the SCU program provides the page boundaries relative to where the page may be stored in SBU memory. The SCU then provides the same set of page boundaries to the drum interface after the data is in the SBU, and initiates the data transfer operation to the drum.

The drum station interfaces one or two modified 865 drum storage units via 7000 I/O channel interfaces in the SBU. Each functional section of the drum station that is not a part of either the basic SBU or SCU discussed previously, is discussed in the following paragraphs in the order listed.

- Station buffer unit (SBU)
  - STAR A data channel interface (SCU coupler)
  - STAR B data channel interface (SAC coupler)
  - Normal channel interface
  - Core control
  - Index compare
  - 7000 I/O channel interfaces (channels 0 and 1)
- Station control unit (SCU)
  - Buffer controller and 1.1-microsecond memory
  - Normal channel interface
  - Block transfer channel
  - SCU gate
  - STAR B data channel interface (SBU couplers A and B)
  - Drum/display logic, keyboard, and microdrum
- Portable maintenance console
- Drum storage unit (modified 865)

INDEX COMPARE

The index compare logic provides a hardware search of a virtual address table contained in SBU core and uses an access to SBU core. It can search for up to two addresses simultaneously. The table contains a list of four 16-bit entries. Each entry has a key and virtual address which relates to a physical device address. The interface is controlled by normal channel bits and provides simultaneous multiple searches for specific keys and/or virtual addresses.

7000 I/O CHANNEL INTERFACE

This interface provides communication between the SBU core control via a 7000 I/O channel and an external peripheral device such as a modified 865 drum storage unit. The 7000 I/O interface provides data buffering and data conversion to and from the drum unit. The 7000 I/O interface has two channels, one for data and one for control. Each channel is 12 bits wide and of parallel format. The control channel is used to transmit function codes to the peripheral device and to receive status signals from the device. The data channel exclusively handles data both to and from the peripheral system. The SCU supplies the function codes and monitors status of the SBU via the normal channels.

The 7000 I/O interface provides a stacking feature which allows starting and terminating addresses to be held until needed so the SCU can be free to complete more important control functions.

DRUM STORAGE UNIT

The DSU used in the drum station is a modified version of the CDC standard 865 DSU. The DSU provides the I/O channel with 12-head parallel format accessibility and a standard storage interface. The DSU uses 768 data tracks and four control tracks in the process of accomplishing its storage function. Two of the control tracks are used as master tracks; one is used for clock pulses and one for index marks. The remaining two are spares. The data heads are organized into 64 head groups with 12 read/write heads per group. Data bits are recorded on the drum via the phase modulation method at a clock rate of 2 megabits per second.

DISK STATION

The disk station (Figure 1-5) provides an extension of mass memory for higher level computer storage. Although the drum and disk stations have many similarities, the major difference in configuration of the disk station is that a second SBU with two 7000 I/O channels is available as an option. In the optional SBU, the 7638-1 disk file subsystem is not included as a part of the option but is available by separate order.

STATION FUNCTIONAL OPERATION

The SBU in a disk station receives functions from the HLP. Depending on the function, the SCU initiates operation of the STAR B interface to send or receive data from the HLP. Data is transferred in page increments, each page having 2048 hexadecimal words. After data is stored in SBU core memory, the SCU initiates operation of one of the 7000 I/O interfaces to transfer a page of data to the disk itself. The disk's surface is divided into one-page increments so that one page of data from memory may be stored on a one-page sector on the disk. When the HLP requests data, the SCU commands

the disk to send data to SBU core via a 7000 I/O interface. Once the data is in SBU core, the SCU initiates an operation to transfer the data from SBU core to the HLP via a STAR B interface. The normal channel interface in the SBU provides the path used by the SCU to control the various SBU interfaces.

The disk station interfaces one or two 7638-1 disk file subsystems via 7000 I/O channel interfaces in the SBU. The discussion given for the SBU in the drum station also applies to the disk station except that the disk station does not use the index compare operation. The SCUs for the drum and disk stations are identical. Components of the disk station are listed as follows:

- SBU
  - STAR A data channel interface (SCU coupler)
  - STAR B data channel interface (SAC coupler; channels 1 and 2)
  - Normal channel interface
  - Core control
  - 7000 I/O channel interfaces (channels 0 and 1)

- SCU
  - Buffer controller and 1.1-microsecond memory†
  - Normal channel interface
  - SCU gate
  - Block transfer channel
  - STAR B data channel interface (SBU couplers A and B)
  - Drum/display logic, keyboard, and microdrum

- Portable maintenance console

- 7638-1 disk file

†200-nanosecond memory is optional.

844 SERVICE STATION

The 844 service station (Figure 1-6) consists of several interfaces which service different equipment attached to various interfaces. In the case of the peripheral A interfaces, the station provides the higher level processor access to second-level stations and vice versa. Typically, a second-level station could be a unit record station or a display/edit station. The network trunk interface(s) services a transceiver, which in turn is connected to a communications link. The only option available for the station is the addition of two more network trunk interfaces, bringing the total of that type of interface to four. The 844 interface provides access to a maximum of three 844 DSUs. The 844 DSUs provide mass storage for data blocks in SBU core memory.

STATION FUNCTIONAL OPERATION

The 844 service station and its interfaces are controlled by programs in the SCU. Normal output channels carry control information to each SBU interface except the network trunk interfaces. The network trunk interfaces receive their commands from the modems, data sets†, or transceivers to which they are attached and transfer their data words to SBU memory.

The peripheral A interfaces react to SCU instructions by transmitting or receiving functions and data words to and from the second-level stations. The 844 interface communicates with up to three 844 DSUs for storage operation with SBU memory. Both the network trunk interface and the 844 interface provide serial to parallel conversion (and vice versa) of data passed between the interface and its respective peripheral device(s). The SAC coupler interface provides access to a higher level processor such as that contained in the STAR 100 Computer System. It receives function commands and data information from the SCU via the normal output channels. Status information is available to the SCU controlling program via the normal input channels. Both the SAC coupler and the peripheral A interfaces transfer data to and from SBU memory in 16-bit parallel word formats.

There are several possible modes of operation for the service station. The first mode of operation exists if the CPU has requested information via the STAR data channel, the station could respond with the proper data from the 844 DSU.

†Registered trademark of AT & T Co.

A second mode of operation for the service station is servicing the peripheral A interfaces connected to second-level stations such as a display/edit station or unit record station. The inputs received from these two stations are data for display purposes or data read from punched cards. Data transmitted to these second-level stations via the peripheral A interface is display information , card punch data, or line printer data for printout.

The third mode of operation for the service station is servicing the network trunk interface. Since the remote terminal rather than the BC controls this interface, the data received or transmitted must be processed and stored in SBU memory and then routed to/from the CPU.

If more than one interface is being used in the SBU, they are forced to share memory. Therefore, the controlling programs in the SCU must schedule operations to maintain the most efficient use of memory. The SCU does not schedule operation of the network trunk interface.

The functional sections comprising the 844 service station are given in the following paragraphs. Many of them have been discussed previously in this section; therefore, only the different ones are discussed.

- SBU
  - STAR A data channel interface (SCU coupler)
  - STAR B data channel interface (SAC couplers; channels 1 and 2)
  - Normal channel interface
  - Core control
  - 844 interface
  - Network trunk interfaces (2; two more are optional)
  - Peripheral A interfaces (4)

- SCU
  - Buffer controller and 1.1-microsecond memory
  - Normal channel interface
  - Block transfer channel
  - SCU gate
  - STAR B data channel interface (SBU couplers A and B)
  - Drum/display logic, keyboard, and microdrum

- Portable maintenance console
- 844 DSU
- Communications transceivers, modems, or data sets
- Second-level stations.

SBU INTERFACES

Since a number of the 844 service station interfaces have been discussed in previous paragraphs relating to other similar stations, only those interfaces which are different from other SBU interfaces are discussed in the following paragraphs.

844 INTERFACE

The 844 interface regulates data transfer operations between the 844 DSUs and SBU memory. Data is transferred serially to the DSU at a rate of 6,800,000 bits per second and in page increments of 2048 hexadecimal words. Mass storage data capacity of the DSU is 23,427 pages.

NETWORK TRUNK INTERFACE

The network trunk interface (NTI) connects an SBU memory access channel to a current switching mode type transceiver, modem, or data set such as the 300 series AT&T Data Set. Depending upon the type of communications device used, the NTI may transfer a serial stream of data of up to 5 megabits per second. Data at the SBU end of the interface is in 16-bit parallel form and in serial form at the modem end of the interface.

The NTI mode of operation is governed completely by information coming from the remote terminal via a modem. There are no normal channels connected from the NTI to the SCU. The NTI connects to SBU memory using a memory access channel.

PERIPHERAL A INTERFACE

The STAR A data channel interface (A interface) connects an SBU memory access channel to a STAR data channel. In addition to being a data transfer link, the A interface responds to function codes sent to it by the SCU; the SCU can obtain status information from it via normal channel bits. Initiation of data transfers is accomplished from the B end of the STAR data channel or from the B interface of a second-level station. On write operations, data flows from the STAR data channel to a memory access channel in the SBU via the A interface. On read operations, just the opposite procedure takes place.

844 DISK STORAGE UNIT

The 844 DSU is a high speed, random access, disk storage device which records data on a removable disk pack. Each pack consists of 12 disks; 10 are used for recording and two (top and bottom) are protective nonrecording disks. Each surface is divided into three equal-sized sectors and has 404 tracks per surface. Each sector contains 2048 16-bit words. The individual bits of these words are recorded serially, bit by bit, on the disk. A sector is the smallest addressable unit on the disk.

COMMUNICATIONS TRANSCEIVERS

A number of different types of communications devices are available and can be used with the NTI. One such type is the transceiver which obtains its name from the fact that is can transmit and receive data simultaneously if necessary. Other types are the modem (modulator/demodulator) and the data set (such as an AT&T 300 series Data Set). Regardless of which devices are used with the NTI interface, they all must be compatible with the current mode interface as defined by AT&T's Bell System Technical Reference for 300 series Data Sets. The NTI activates the transceiver, and once the transceiver is synchronized and all other required signals are present, data transmission may begin. Parallel data from SBU memory is transferred in 16-bit words to or from the NTI interface and then bit by bit, in serial form, to or from the transceiver.

STAR-100 SERVICE STATION

The STAR-100 service station (Figure 1-7), similar to other service stations in the peripheral station family, is used to service different equipment attached to its interfaces. This station has the similarities of at least two other stations in that it may communicate with the same type of DSU used in the paging station, and it services second-level stations via peripheral A interfaces just as the 844 service station does. Therefore, the station functional operation that follows is similar to those two stations previously mentioned. No options are offered for the STAR-100 service station.

STATION FUNCTIONAL OPERATION

The modified 865 DSU is attached to a 7000 I/O channel interface which in turn is controlled by the SCU program. The SCU also controls the eight peripheral A interfaces, which in turn provide access to second-level stations such as the unit record station or the display/edit station.

The STAR-100 service station may receive from an HLP (such as the STAR-100 Computer) requests for data from SBU memory or requests for a storage area in which to store data. In the case of the DSU, one page of data is 2048 16-bit words. In display/edit stations where data is being retrieved from the microdrum, one track of data is 1152 16-bit words.

The SAC coupler interfaces provide two access paths via STAR data channels to an HLP (such as the CPU in the STAR-100 Computers). They recieve function commands and data from the SCU via the normal output channels. Status information is available to the SCU operating program via the normal input channels. All interfaces in this station provide parity checking and transfer data to and from SBU memory in 16-bit parallel word formats.

The functional sections comprising the STAR-100 service station are given in the following paragraphs. Since all of them have been discussed previously in this manual, refer to the appropriate description for details of the interface in question.

- SBU
  - STAR A data channel interface (SCU coupler)
  - STAR B data channel interface (SAC couplers; channels 1 and 2)
  - Normal channel interface
  - Core control
  - 7000 I/O channel interface
  - Peripheral A interfaces (8)

- SCU
  - Buffer controller and 1.1-microsecond memory
  - Normal channel interface
  - Block transfer channel
  - SCU gate
  - STAR B data channel interface (SBU couplers A and B)
  - Drum/display logic, keyboard, and microdrum

- Portable maintenance console

- DSU (modified 865)

- Second-level stations

STAR 1B SERVICE STATION

The STAR 1B service station (Figure 1-8) provides servicing of second-level stations having B type interfaces. Primarily, however, the station is dedicated to servicing a STAR 1B Computer System and its second-level station functions.

The STAR 1B service station has two 841 MDD interfaces plus its peripheral A interfaces, and its index compare logic. No options are available for this station.

STATION FUNCTIONAL OPERATION

Like the other service stations in the peripheral station family, the peripheral A interfaces are controlled by the SCU program(s) and connect to second-level stations having a compatible STAR B data channel interface. Similar to other stations, the A interface responds to function codes it receives from the SCU, and the SCU can obtain status information from it via normal channel bits. Initiation of data transfers is begun from the B interface of the STAR data channel or from the B interface of a second-level station.

The functional sections comprising the STAR 1B service station are given in the following paragraphs. Since all of them except the 841 MDD interface in the SBU have been discussed previously in this section, only the 841 MDD interface is discussed.

- SBU
  - STAR A data channel interface (SCU coupler)
  - STAR B data channel interface (SAC couplers; channels 1 and 2)
  - Normal channel interface
  - Core control
  - 841 MDD interfaces (2)
  - Index compare logic
  - Peripheral A interfaces (6)

- SCU
  - Buffer controller and 1.1-microsecond memory
  - Normal channel interface
  - Block transfer channel
  - SCU gate
  - STAR B data channel interface (SBU couplers A and B)
  - Drum/display logic, keyboard, and microdrum

- Portable maintenance console

- Second-level stations

- 841 MDDs

STATION INTERFACES

Since all of the SCU interfaces and functional sections have been discussed previously for other stations, refer to the appropriate paragraphs for details. In the SBU, only the 841 MDD interface is different from other stations and is therefore discussed in the following paragraphs.

841 MDD INTERFACE

The 841 MDD interface regulates data transfer operations between the 841 MDDs and SBU memory. The two 841 MDD interfaces in the SBU may connect to the same 841 MDD units. The two interfaces are completely independent of one another and can perform simultaneous data transfer operations as long as the two procedures involve two different MDDs.

844 STORAGE AND MAGNETIC TAPE STATIONS

These two stations (Figure 1-9 and 1-10) share common elements and are therefore discussed together. Foundation elements for either station consists of one SBU and one SCU with a second SBU offered as an option. Differences in the two SBUs of the two stations are that the two 844 DSU interfaces in the 844 storage station are standard while the two 3000 I/O channel interfaces are offered as optional elements. In the magnetic tape station, the opposite is true. The two 844 DSU interfaces may control up to eight disk storage units each.

STATION FUNCTIONAL OPERATION

The CPU of the STAR-100 Computer System initiates station operation by either requesting data from the station or requesting a memory location in which to store data. All data transfers to and from the CPU are performed in 16-bit word sizes of 2048 words each (one page). Serial data can be transferred to the 844 DSU at a rate of 6,800,000 bits per second (recording frequency of the 844 DSU). If the requested page is stored in SBU memory, the SCU program sends the page boundaries to the SAC coupler interface via normal output channel bits. The SCU then initiates the data transfer to the CPU. If the page requested is located on tape or disk, the SCU program connects to the 3000 I/O channel or DSU interface and sends the page boundaries to the SBU. The data transfer is then initiated by the SCU program.

Since all interfaces (except the 3000 I/O channel interface) have been explained in previous paragraphs of this manual, only the 3000 I/O channel interface is discussed. Prior to this discussion, however, the two stations are compared in a listing as follows:

844 Storage Station

- SBU
  - 844-A interface
  - 844-B interface
  - 3000 I/O channel 0 interface (optional)
  - 3000 I/O channel 1 interface (optional)
  - STAR A data channel interface (SCU coupler)
  - STAR B data channel interface (SAC coupler; channels 1 and 2)
  - Normal channel interface
  - Core control
- SBU No. 2 (optional)

  NOTE

  3000 I/O channel interfaces are not included with optional SBU.
- SCU
  - Buffer controller and 1.1-microsecond memory
  - Normal channel interface
  - Block transfer channel
  - SCU gate
  - STAR B data channel interface (SCU couplers A and B)
  - Drum/display logic, keyboard, and microdrum
- 844 DSUs
- 3000 peripheral controllers (optional)

Magnetic Tape Station

- SBU
  - 844-A interface (optional)
  - 844-B interface (optional)
  - 3000 I/O channel 0 interface
  - 3000 I/O channel 1 interface
  - STAR A data channel interface (SCU coupler)
  - STAR B data channel interface (SAC coupler)
  - Normal channel interface
  - Core control
- SBU No. 2 (optional)

  NOTE

  844-A and B interfaces are not included with optional SBU.
- SCU
  - Buffer controller and 1.1-microsecond memory
  - Normal channel interface
  - Block transfer channel
  - SCU gate
  - STAR B data channel interface (SCU couplers A and B)
  - Drum/display logic, keyboard, and microdrum
- 844 DSUs (optional)
- 3000 peripheral controllers

3000 I/O CHANNEL INTERFACE

The 3000 I/O channel interface processes data transfers and functional control between SBU memory and a 3000 magnetic tape controller. The SBU data lines transmit 16-bit words plus two parity bits. The 3000 I/O channel transfers 12-bit words for storage on magnetic tape units in six- or eight-bit characters. Using normal channel bits, the SCU controls data transfers and sends various functions to the tape controller via the 3000 I/O channel interface. A major difference between the 3000 I/O channel interface used in the SBU and a standard 3000 series I/O channel is that the standard 3000 channel can handle eight interrupt lines and the SBU 3000 I/O channel is limited to four interrupt lines. Therefore, no more than four peripheral controllers can be connected to the 3000 I/O channel interface if the interrupt feature is used.

UNIT RECORD STATION

The unit record station (Figure 1-11) is one of the second-level stations and therefore connects to the HLP's central memory indirectly through the A level interface of a first-level station, such as the STAR-100 service station. Similar to its use in other stations, the SCU controls operations between peripheral controllers/devices and the A level interface in the first level station.

The SCU in the unit record station contains two STAR B data channel interfaces, two 3000 I/O channel interfaces, a normal channel interface, and a BC and all of its associated channels, etc. The peripheral controllers and devices to which the 3000 I/O channel interfaces could be connected are a 3447 or 3649 card reader controller and 405 card reader, a 3446 card punch controller and 415 card punch, and a 3555-1 line printer controller and 512-1 line printer. The SCU controlling program permits the connection of a second line printer.

All peripheral devices and their associated controllers must be ordered separately and are not included with the SCU used in this station. A listing of the unit record station's major components is as follows:

- SCU
  - Buffer controller and 1.1-microsecond memory
  - Normal channel interface
  - Block transfer channel
  - SCU gate
  - Drum/display logic, keyboard, and microdrum
  - 3000 I/O channels 0 and 1
  - STAR B data channel interface (SAC channels 1 and 2)

DISPLAY/EDIT STATION

The display/edit station (Figure 1-12) is a second-level station that drives a maximum of 28 keyboard/display terminals and provides for editing data externally from a higher level CPU or manually at the station level via the keyboard. This station also processes control entries, provides contiguous character editing, formats the data, and maintains the display network. This station is composed of an SCU and an SDU which is described in the following paragraphs. The SCU in this station is similar to the SCUs of other stations already described. The only difference is the addition of an SDU coupler interface. The SDU coupler is discussed in section 4 of this manual.

STATION DISPLAY UNIT

The basic SCU is a 4096 16-bit word memory used for holding messages and data displayed on a group of seven displays. Three additional 4K memory modules and associated logic can be added to serve a maximum of 28 displays. A second purpose of the SDU is to provide a data path from the keyboards to the SCU normal channels. Although all data and control words are exchanged between the SDU and SCU coupler via receiver/transmitter circuits, the SDU is composed of four main functional sections. They are described in the order listed:

Master control logic
Display group logic
Memory control logic
Memory module(s)

MASTER CONTROL LOGIC

In general, the master control logic provides an interface for data transfer between the keyboard/display units and other functional sections of the SDU and the receiver/transmitter circuits. Its main functions are:

- Produces a master timing signal for starting the 45-bit holding and shift register and timing chain in the display group logic module.
- On read-from-SDU-memory operations, transmits data from its registers through the SDU coupler to the block transfer channel in the SCU.

- On write-into-SDU-memory operations, gates data from the block transfer channel through the SDU coupler into its registers to the appropriate memory module.

- Enables offset operations to be performed on either a read or write operation. Offset operations permit an operator to change the position of a character during message editing functions.

- Addresses all four memory modules simultaneously, if necessary (either block transfer channel address or streaming address).

- Provides a unit selection counter for selecting a specific keyboard/display unit.

DISPLAY GROUP LOGIC

This functional section of the SDU contains for the most part the controlling logic which acts in response to the master control logic to regulate data to and from its seven keyboard/display units. If more than seven keyboard/display units are required, one display group logic module is added for each increment of seven units added.

A streaming counter, used with the unit counter in the master control logic, is used in addressing memory during the streaming mode of operation. Streaming simply means the continuous uninterrupted flow of data. The first five counts of the stream counter enables one line (32 words) to be displayed on a CRT. As the count progresses through a count of eight, 16 lines or 512 words may appear on a CRT. Address locations in memory are sectioned such that a controlling flip flop in the display group logic determines when the A or B field is displayed on a CRT.

The display group logic decodes and generates the 96-character set of ASCII-coded characters using six modules, 16 characters per module. The seven holding and shift registers per display group logic temporarily store this data until ready for display on the CRT(s). The cursor is also displayed under the appropriate character after a valid comparison is made of the contents of bits 2 through 6 in the cursor shift register and five bits of the stream counter (in master control logic).

MEMORY CONTROL LOGIC

The memory control section of the SDU regulates the accesses to memory modules and controls the video signals used to run the CRT displays. During read operations, this logic receives 16-bit data words from the memory module sense amplifiers. During write operations, it receives 16-bit words from the SDU coupler in the SCU.

A switch on this module permits the selection of either of two display formats of the A and B fields displayed on a CRT (refer to Figure 4-2). The A field is used for the display of data and the B field is used for displaying control and command words and other static displays.

MEMORY MODULE

Each memory module contains 4096 16-bit words of storage, or enough word locations to service seven keyboard/display units. The character buffer area in each module has seven A data fields (of 512 words each) and seven B fields (of 64 words each). The cursor address area in each module has seven A fields (of 56 words each) and seven B fields (of one word each); one word location in memory is not used. For each increment of seven keyboard/display units added to the system, one memory module is added.

STORAGE (MEDIA) STATION

The storage/media station (Figure 1-13) provides for the attachment and control of MDDs using the 841 MDDs and magnetic tape transports via the proper 3000 controller and series 65X magnetic tape units. It provides file storage and performs file management functions. Besides performing disk station type functions for the attached 841 MDDs, it accommodates read/write tape records, provides translation and assembly/disassembly functions, and communicates with the operator.

STATION FUNCTIONAL OPERATION

The storage (media) station is an intermediate storage station for a higher level processor. The SBU in this storage station receives functions from the HLP. Depending upon the function, the BC in the SCU initiates operation of the B interface preparatory to data transfer operations. Data is transferred by pages, with each page containing 2048 hexadecimal words. Once data is in SBU memory, the SCU program transfers the data either to an MDD or to a magnetic tape unit for storage.

When the HLP requests data, the SCU program initiates operation of the proper interface to transfer the requested data into SBU core memory. Once in core, the SCU initiates operation of the B interface to transfer the data to the HLP. The normal channel interface provides the SCU with the required control and status paths to the various SBU interfaces.

The functional sections comprising the storage (media) station are listed below. Since all interfaces of the station have been discussed previously, refer to the appropriate paragraph for details.

- Station buffer unit

  STAR A data channel interface (SCU coupler)
  STAR B data channel interface (SAC couplers 1 and 2)
  Normal channel interface
  Core control
  3000 I/O channel interfaces (channels 0 and 1)
  841 MDD interface

- Station control unit

  Buffer controller and 1.1-microsecond memory
  Normal channel interface
  Block transfer channel
  SCU gate
  STAR B data channel interface (SBU couplers A and B)
  Drum/display logic, keyboard, and microdrum

- Portable maintenance console

- 841 Multiple Disk Drive

STAR-100 MAINTENANCE CONTROL UNIT

The STAR-100 maintenance control unit (Figure 1-14) consists primarily of the basic SCU which provides two SAC channel couplers to the central memory of a STAR-100 computer. The SCU also contains a special control interface which is connected to multiple access lines to and from the computer's mainframe. These lines serve as the means for measuring and monitoring functions of the STAR-100 Computer. The maintenance control unit also has a 3000 I/O channel interface, line printer interface, and a disk storage drive interface. A listing of the maintenance control unit's major components is as follows:

- STAR B data channel interface (SAC couplers; channels 1 and 2)
- Special control interface
- Buffer controller
- 3000 I/O channel interface
- 854 disk storage drive interface
- Line printer interface
- Normal channel interface
- Portable maintenance console (external to MCU)

STATION FUNCTIONAL OPERATION

The two SAC coupler interfaces and the special control interface allow the maintenance control unit to regulate information flow, control pulses, and monitor performance of the STAR-100 Computer. The maintenance control unit provides system deadstart capability via the portable maintenance console and system monitoring. The three general modes of operation relative to maintenance control unit/computer program functions are:

- Operation of a computer diagnostic maintenance program used to locate faults and malfunctions within the maintenance control unit.
- Running diagnostic routines on the computer. The maintenance control unit loads diagnostics into the computer via the portable maintenance console, controls and monitors operations of the diagnostics, and indicates the results of the tests on either the line printer or other appropriate peripheral device.

- The maintenance control unit loads the program into the computer via the portable maintenance console and controls and monitors its operation. In this mode of operation, the maintenance control unit is responsible for autoloading the CPU and the BC of first-level stations, running on-line diagnostics, and restarting the CPU.

A special control interface connects to the microcode memory of the STAR-100 Computer. This interface provides the maintenance control unit with the ability to load and store microcode and to load microcode diagnostic routines. The interface also provides control to run the routines under maintenance control unit control.

PERIPHERAL EQUIPMENT

Three types of interfaces are supplied with the MCU for connecting to 3000 peripheral controllers, to a disk storage driver interface, and to a line printer interface. The 854 DSD interface accommodates up to four disk drives. The line printer interface services one line printer.

HIGH CAPACITY DISK (HCD) STATION

(To be supplied.)

HARDWARE ELEMENT DESCRIPTIONS

The following paragraphs contain the physical descriptions of the various units comprising the peripheral stations. In the case of the SCU, the instruction set of its BC is also described because of its importance and extensive usage in the peripheral stations.

STATION BUFFER UNIT

The SBU is a cabinet containing power supplies, logic chassis, eight core memory mdoules, core control logic, SCU interface logic, and additional logic as required by each station. Each SBU has an associated SCU for control purposes. Figure 1-19 shows the layout of the SBU.

CORE MEMORY

The eight core memories used in the SBU are the same as the 1.1-microsecond memory modules in the SCU. Each module contains 4096 18-bit words (16 data bits and 2 parity bits). Total capacity of the SBU memory modules is 32,768 words (65,536 8-bit characters).

CORE CONTROL

The SBU core control logic provides 12 accesses to SBU core. The eight memory modules are phased by eight, allowing eight memory references to separate modules during the 1.1-microsecond memory cycle time.

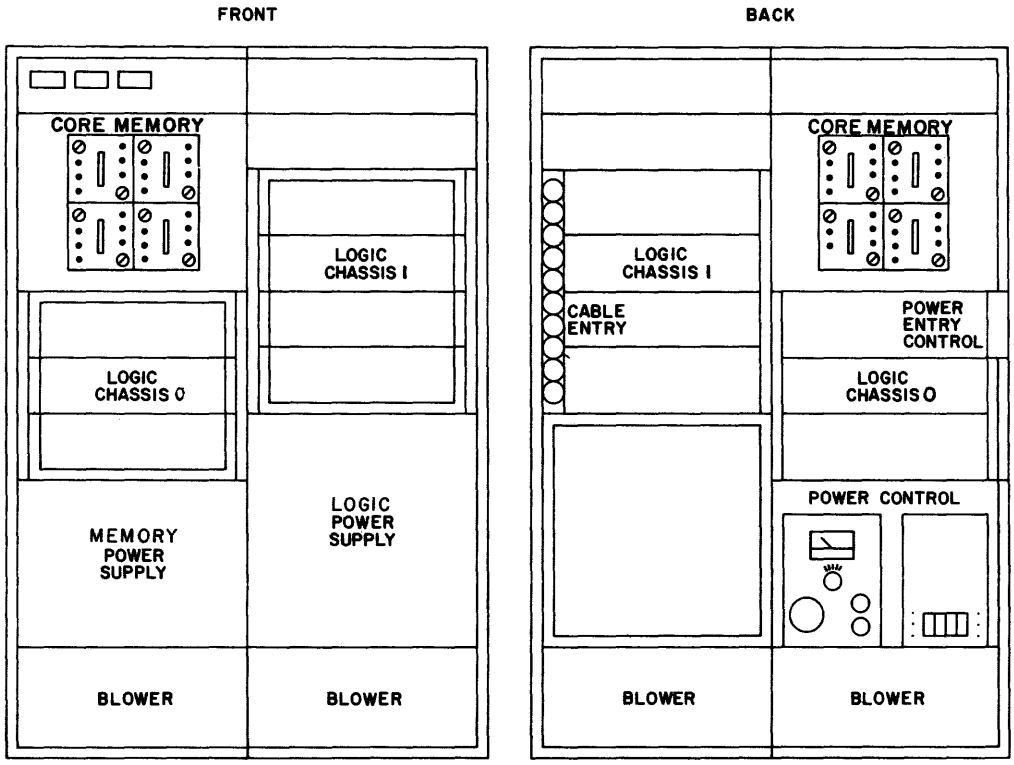


Figure 1-19. Station Buffer Unit Component Locations

SCU INTERFACE

The interface between the SBU and its associated SCU consists of an interface between the SCU block transfer channel and an SBU core access, and an interface between SCU normal channels and the SBU. The normal channel bits are used in the SBU to control the various logic interfaces on the 12 accesses to SBU core.

STATION CONTROL UNIT

The SCU is a cabinet containing a power supply, control panels, logic chassis, BC with memory module, microdrum, display/keyboard, SCU logic, and additional logic as required by each station (Figure 1-20).

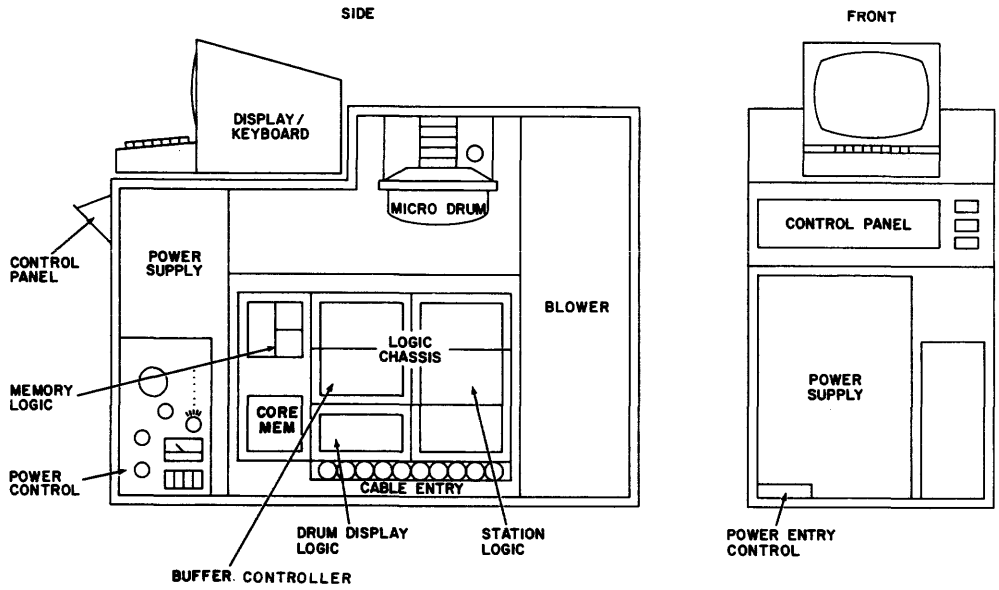


Figure 1-20. Station Control Unit Component Locations

BUFFER CONTROLLER

The BC is an internally programmed, parallel-mode digital device using a coincident current magnetic core memory. Cycle time of the core memory is 1.1 microseconds, with a capacity of 8192 18-bit words (16 data bits plus 2 parity bits). If the optional 8K semiconductor memory is installed, cycle time is 200 nanoseconds. However, it is software supported only in the disk and paging stations.

REGISTERS

The BC contains the following registers.

Program address	(P)	16 bits
Accumulator	(A)	16 bits
Two index registers	(B_1, B_2)	16 bits each

It also contains a 16-bit adder with a 17th adder generate bit which indicates an end-off carry. The BC also contains a condition bit which can be altered and tested under program control. All arithmetic in the BC is done in twos complement mode.

DATA CHANNELS

The BC provides a single, parallel-block transfer channel with hardware ready/resume for high speed data transfer. Its maximum rate is one 16-bit word plus 2 parity bits per memory cycle.

The BC also provides up to 512 individually programmed normal channel bits for lower speed data transfer and device and station control. These bits are organized into 16 input and 16 output channels with 16 bits in each channel. Their use is specified by the individual station design.

INSTRUCTION SET

The instruction set of the BC is designed to effectively manipulate the normal channel bits and perform logical operations. It uses several modes of addressing: no address, direct, indirect, indexed, relative forward/backward, and indirect indexed. The types of instructions are:†

- Selective stop - conditioned by control panel
- Selective set, clear, and complement bit n of (A)
- Count leading zeros, Ai to Af
- Shift (A) right n places
- Transfer (A) to B_1 or B_2
- Set/clear condition bit - unconditional, condition by internal conditions [(A) contains odd number of ones, adder bit is one, and shift network bit 15 is one] , or the condition of a specified normal channel bit
- Set/clear specified normal channel bit
- Transfer (A) to/from normal channel (16 bits)
- Enter A_1, B_1, or B_2 with program address
- Compare specified index with operand to alter condition bit
- Load (A) - normal/complement, left/right byte, destructive load
- Subtract
- Exclusive OR

†Parentheses indicate contents of

- Logical product
- Replace add, replace add one
- Replace left/right byte
- Store - normal, zeros
- Jump - unconditional, (A) zero/nonzero, (A) positive/negative, condition bit true/false
- Input/output block of n words

The timing of the BC is based on the memory cycle time (MCT).

- No address instructions - 1 MCT
- Direct, indexed, relative instructions - 2 MCTs
- Replace type instructions - add 1 MCT
- Block transfer instructions - add a minimum of 1 MCT for each word transferred
- Indexing of an address does not require an additional MCT
- Parallel shift network allows all shifts to be executed in 1 MCT

MICRODRUM

The microdrum is a fixed-head magnetic drum storage device used for station autoload, program overlays, display refresh, and temporary storage. It is under direct control of the BC and is accessed by the BC on normal channels.

The microdrum has 36 data tracks plus four control tracks. Each track has one fixed head. Twenty-four data tracks contain write lockout protection controlled from the maintenance panel.

Rotation rate of the microdrum is 3600 rpm (17 milliseconds per revolution); the bit recording frequency is 1 MHz. Data is recorded serially and in phase modulation. Inherent capacity of the drum is 16,000 bits per track, with a total of 576,000 bits for the unit.

Two storage formats are used with the microdrum, display format and data format. In data format, the data is recorded in 18 sectors per track and is sector-alterable. Each sector contains 64 16-bit words.

In display format, data is stored to match the display refresh timing and storage requirements. A single track holds a display image; it is recorded in 576 sectors, each sector recording two 8-bit characters. The data is sector-alterable.

KEYBOARD/DISPLAY

The keyboard/display is used for station maintenance, and where applicable, operator communications. This same keyboard/display unit is also used as the terminal display for the display station.

The display head is a cathode ray tube display with a diddle yoke. Characters and the cursor are formed within a 5 by 9 dot matrix created by the diddle pattern and digital control of the electron beam (Figure 1-21).

The display format contains 18 lines with 64 characters per line. The bottom two lines are normally for control entry. The character generator within the SCU generates the 64-character ASCII set. The keyboard uses a standard typewriter keyboard arrangement, but includes the full 64-character ASCII symbol set. It also includes four cursor control keys, 10 function keys (momentary), and four mode keys (lock-up). The use of these keys is controlled by the station software.

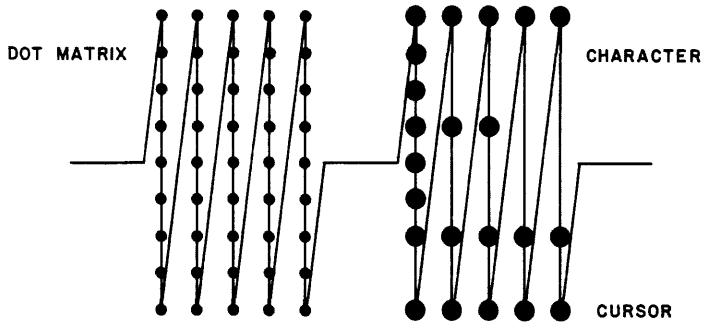


Figure 1-21. Diddle Pattern

PORTABLE MAINTENANCE CONSOLES

Both portable maintenance consoles are suitcase-sized portable units (Figures 5-2 and 5-3) designed for providing maintenance personnel access to and control of the BC in an SCU.

The control panel of the TF201 console contains switches, controls, indicators, a paper tape reader, input and output signal cables, and a power cable. Error checking and output control logic are housed in a card rack which mounts under the front panel.

The console's paper tape reader is an electromechanical assembly which reads eight-level paper tape at a nominal speed of 40 tape bytes per second. At 10 bytes per inch, reader speed may also be expressed as 4 inches per second. This speed is fixed by the console's timing and cannot be varied.

The control panel of the TF204 console contains switches, controls, a keyboard, display readouts, input and output signal cables, and a power cable. Also, a cassette magnetic tape unit is mounted on the control panel which is used for transmitting prerecorded data to the BC or for recording data from the BC memory.

STATION DISPLAY UNIT

The SDU is a cabinet containing logic and memory power supplies, four core memory modules, core control logic, and any additional logic as required by the station configuration. Each SDU is connected to an SCU for control purposes. The core memory modules and memory power supply are the same component modules as those used in the SBU and SCU. Figure 1-22 shows the physical layout of an SDU.

405 CARD READER

The 405 card reader is a high speed, electromechanical device having the capability to read up to 1200 80-column punched cards per minute. The card reader's input hopper can accommodate 4000 cards and the output stacking bin can likewise handle 4000 cards. A limited size secondary stacking bin holds up to 240 cards which may be the resulting output of a limited sorting or card rejecting operation.

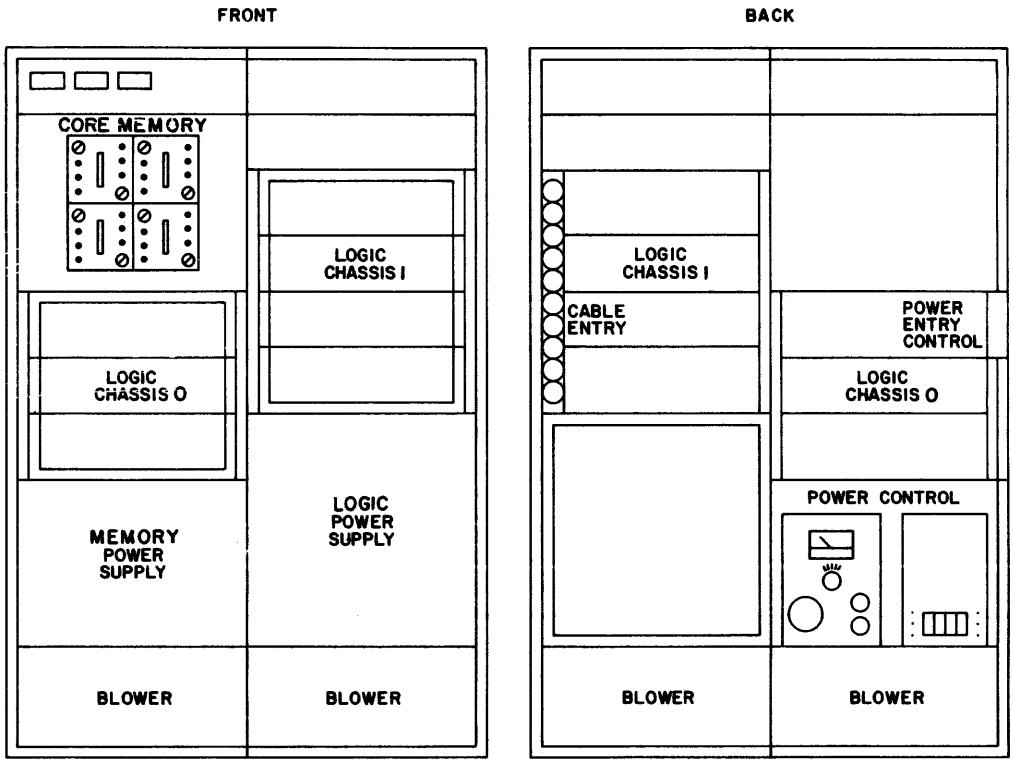


Figure 1-22. Station Display Unit Component Locations

415 CARD PUNCH

The 415 card punch is an electromechanical device having the capability to punch up to 250 80-column cards per minute under program control. The input hopper holds 1200 blank 80-column cards, and the output stacker can receive 1500 punched cards. To reduce the possibility of an incorrectly punched card being used in a subsequent card reading operation, a read-after-punch feature checks each card for punching errors.

512-1 LINE PRINTER

The 512-1 line printer is a high-speed line printer using a 96-character train cartridge. Line width is 136 columns, and line spacing is 6 or 8 lines per inch as selected by either an operator or by the software. Paper skip speed is 70 inches per second at six lines per inch and 60 inches per second at eight lines per inch. The 595 series train cartridge is not included with the printer but is available on separate order.

657 AND 659 MAGNETIC TAPE TRANSPORTS

These two series of magnetic tape transports are capable of recording data on seven- and nine-track 1/2-inch magnetic tapes at various rates. Both transport families are available in four models each and all use the NRZI (nonreturn to zero, change on ones) recording format. The 659 series (9-track) units also operate in the phase encoded mode.

Tape motion is accomplished by dual vacuum capstans. Vacuum columns act as buffers between the synchronous tape speed, as produced by the capstans, and the asynchronous motion produced by the reel drive servo systems. Reel drive servos are controlled by pressure transducers. Maximum tape life is obtained by allowing the capstan to contact only the nonrecording surface on the tape.

A brief summary of the two transport families is listed in Table 1-1.

841 MULTIPLE DISK DRIVE (MODIFIED)

The modified 841 multiple disk drives used on the storage media and STAR 1B service stations each have a storage capacity of 235.9×10^6 data bits. Each drive has 20 recording surfaces, 180 tracks per surface, and 3600 data tracks for each unit. Each of the two sectors per track contains 32,768 data bits. The recording surfaces rotate at 2184 rpm, and the pickup heads suspended over the recording surfaces average between 25 and 135 milliseconds for positioning over the proper sector and track upon computer command. The data recording rate is 2,520,000 bits per second, maximum. The disk drives use the 851 disk packs as the recording medium.

TABLE 1-1. PHYSICAL AND FUNCTIONAL PROPERTIES OF MAGNETIC TAPE TRANSPORTS

	657 Series (7-track)				659 (9-track)			
Recording format	-1 NRZI	-2 NRZI	-3 NRZI	-4 NRZI	-1 NRZI/ Ph Enc	-2 NRZI/ Ph Enc	-3 NRZI/ Ph Enc	-4 NRZI/ Ph Enc
Recording density/ transfer rate	200 cpi/ 7.5K cps	200 cpi/ 15K cps	200 cpi/ 22.5K cps	200 cpi/ 30K cps	800 cpi (NRZI)/ 30K cps	800 cpi (NRZI)/ 60K cps	800 cpi (NRZI)/ 90 K cps	800 cpi (NRZI)/ 120K cps
	556 cpi/ 20.8K cps	556 cpi/ 41.7K cps	556 cpi/ 62.4K cps	556 cpi/ 83.4K cps	1600 cpi (Ph Enc)/ 60K cps	1600 cpi (Ph Enc)/ 120K cps	1600 cpi (Ph Enc)/ 180K cps	1600 cpi (Ph Enc)/ 240K cps
	800 cpi/ 30K cps	800 cpi/ 60K cps	800 cpi/ 90K cps	800 cpi/ 120K cps				
Tape speed	37.5 ips	75 ips	112.5 ips	150 ips	37.5 ips	75 ips	112.5 ips	150 ips
Rewind time (2400 ft/reel)	210 sec, max	130 sec, max	85 sec, max	85 sec, max	210 sec, max	130 sec, max	85 sec, max	85 sec, max
Start time (in msec)	3.0, max	3.0, max	3.0, max	3.0, max	3.0, max	3.0, max	3.0, max	3.0, max
Stop time (in msec)	3.0, max	3.0, max	2.5, max	2.3, max	3.0, max	3.0, max	2.5, max	2.3, max

NOTES:

NRZI means nonreturn to zero (change on ones).
Ph Enc means phase encoded.
Tape speed is given in inches per second (ips).
Max means maximum.
cpi means characters per inch.
cps means characters per second.

844 DISK STORAGE UNIT

The 844 disk storage unit, used in several of the peripheral stations, has a storage capacity of 755 million bits grouped in three sectors per track, with each disk surface having a total of 404 usable tracks. The disk rotates at 3600 rpm, and head positioning time over the recording surfaces averages between 10 and 55 milliseconds. The 844 utilizes 19 read/write heads and one track-servo head for recording on its 19 disk surfaces (top of the top disk and bottom of the bottom disk are protective nonrecording surfaces). Data recording rate is 6.8 million bits per second. The 844 uses the 881 disk pack as its recording medium.

854 DISK STORAGE DRIVE

The 854 disk storage drive, used with the STAR-100 or STAR-65 maintenance control unit, has a storage capacity of 53.4×10^6 data bits. The disk pack recording surfaces rotate at 2400 rpm. The data recording rate is 1.25×10^6 data bits per second. Each of the ten disk recording surfaces is subdivided into 200 tracks, thereby providing a total of 2000 tracks for the unit. Head positioning time of the single access mechanism is in the range of 24 to 135 milliseconds.

865 DRUM STORAGE UNIT

The drum storage unit is a model 865 modified to be compatible with a 7000 I/O channel and the 7000 interfaces in the SBUs of the paging (drum) and STAR-100 service stations. It is an electromechanical recording drum having a total storage capacity of 46.1×10^6 bits. The drum is addressable in sectors of quarter or full pages. Refer to Table 1-2 for sector data bit capacity. It rotates at 1800 rpm and uses 12 parallel heads in any one of 64 head groups to cover 768 data tracks.

TABLE 1-2. PERIPHERAL STORAGE UNIT CHARACTERISTICS

	841 Disk Storage Drive (Modified)	844 Disk Storage Unit	854 Disk Storage Drive	865 Drum Storage Unit (Modified)	7638-1 Disk File Subsystem (Modified)	High Capacity Disk File (819) Subsystem
Total bit capacity per drive (data)	235.9×10^6	755×10^6	53.4×10^6	46.1×10^6	5×10^9	2.1×10^9 †††
No. of tracks per surface	180	404	200	768	512	404
Sectors per track	2	3	16	†	38	16
Rotational speed	2184 rpm	3600 rpm	2400 rpm	1800 rpm	1710 rpm	3600 rpm
No. of recording surfaces	20/drive	19/drive	10/drive	1	32	40/drive
Latency	27.6 ms	16.7 ms	25 ms	33.3 ms	35 ms	16.7 ms
Access time	24.5-135 ms	10-55 ms	24-135 ms	17 ms	24-140 ms	15-80 ms
Bits/sector(data)	32,768	32,768	1664	††	32,768	32,768
Clock rate (bits per second)	2.52×10^6	6.8×10^6	1.25×10^6	2×10^6	2.5×10^6	9.67×10^6
Data transfer rate (bits per second)	2.52×10^6	6.8×10^6	1.25×10^6	24×10^6	40×10^6	38.7×10^6
Disk pack used	871	881	851	NA	NA	NA

†Quarter page has 88 sectors per track; full page has 22 sectors per track.
††Quarter page has 8192 bits per sector; full page has 32,768 bits per sector.
†††Add 2×10^9 bits for each optional HCD added; maximum of 4 per controller.

7638-1 DISK FILE SUBSYSTEM

The 7638-1 disk file subsystem used with the disk station consists of two controllers and one 817 Disk File. The physical file is referenced as two logical files. The controllers contain the logic necessary for controlling operations between the disk file and the 7000 I/O channel. The 817 Disk File contains a memory unit and a hydraulic unit. The memory unit consists of two double-ended disk drive (stacks) which rotate asynchronously at 1710 rpm. Each of the four half-stacks contains 18 disks (36 surfaces, 32 of which are available for data storage). There are two hydraulic positioning assemblies, and each services a pair of half-stacks. Each of the two access assemblies is defined as one file unit (0 and 1). Each access assembly contains two separate horizontally opposed groups of 16 head arms, one group for each half-stack. Two head pads, each of which contains one read/write head, are mounted on the end of each head arm. Therefore, each of the 32 surfaces on a half-stack has its own read/write head. Each unit is divided into four groups of 16 heads each (two groups on each half-stack). Each access assembly can be moved to any of the 512 data positions on each stack by receiving the proper function. Refer to Table 1-2 for the disk file characteristics.

HIGH CAPACITY DISK (HCD) FILE SUBSYSTEM

The high capacity disk (819 HCD) file subsystem used with the HCD station consists of a controller and a high capacity disk. The controller contains the logic for controlling operations between the HCD file and a 7000 I/O channel in an SBU. The HCD file has an operational capacity of over 2.1 billion bits when used in a sectored format of 16 sectors per track, 10 head addresses per cylinder, and 404 cylinders per drive. Head positioning time is an average of 50 milliseconds and rotates at 3600 rpm. The data transfer rate is 38.7 million bits per second. Table 1-2 lists pertinent characteristics of the HCD file subsystem.

STATION BUFFER UNIT INTERFACES 2

GENERAL

This section contains a general overview of the station buffer unit (SBU) followed by a brief operational description of the SBU and detailed descriptions of each SBU interface.

NOTE

Throughout this section, word formats illustrating unused bits by the cross-hatch pattern (for example, Figure 2-4) must have these bits all set to zeros. If they are not, undefined results may occur.

SBU OVERVIEW

The SBU consists of a number of functional subsections (Figure 2-1) but functions mainly as a buffer memory for I/O data being transferred between peripheral equipment and a higher level processor (HLP). Each SBU is connected to a companion station control unit (SCU) which controls the input/output interfaces within the SBU. The SBU has the following basic characteristics and capabilities.

- Core memory storage for 32,768 data words organized around eight memory modules of 4096 16-bit words. Each module has independent read/write capability and a 1.1-microsecond cycle time.
- Twelve independent memory access channels (MAC) for connection of input/output interfaces. Each MAC provides for incrementing memory addresses, indicating page boundaries, and controlling data transfer functions.
- A channel priority system which resolves conflicts in any memory module according to a priority number where channel number 0 is the highest priority and access number 11 is the lowest.
- A phased memory system which permits an effective memory cycle time of up to 136 nanoseconds for references to sequential memory locations.
- The SBU is controlled by the buffer controller (BC) in the SCU via the normal I/O channel interface.

SBU memory connects to I/O interfaces present in an SBU cabinet through the 12 MACs and to the SCU normal channel interface logic. The particular method in which these connections are made is determined by the type of station in which the SBU is to be used. Generally, I/O interfaces allow connection of the SBU to an HLP memory, to other peripheral stations, to peripheral devices, and to the SCU normal channels. The I/O interfaces connect to the SBU core control logic for memory access and to the SCU normal channels for control.

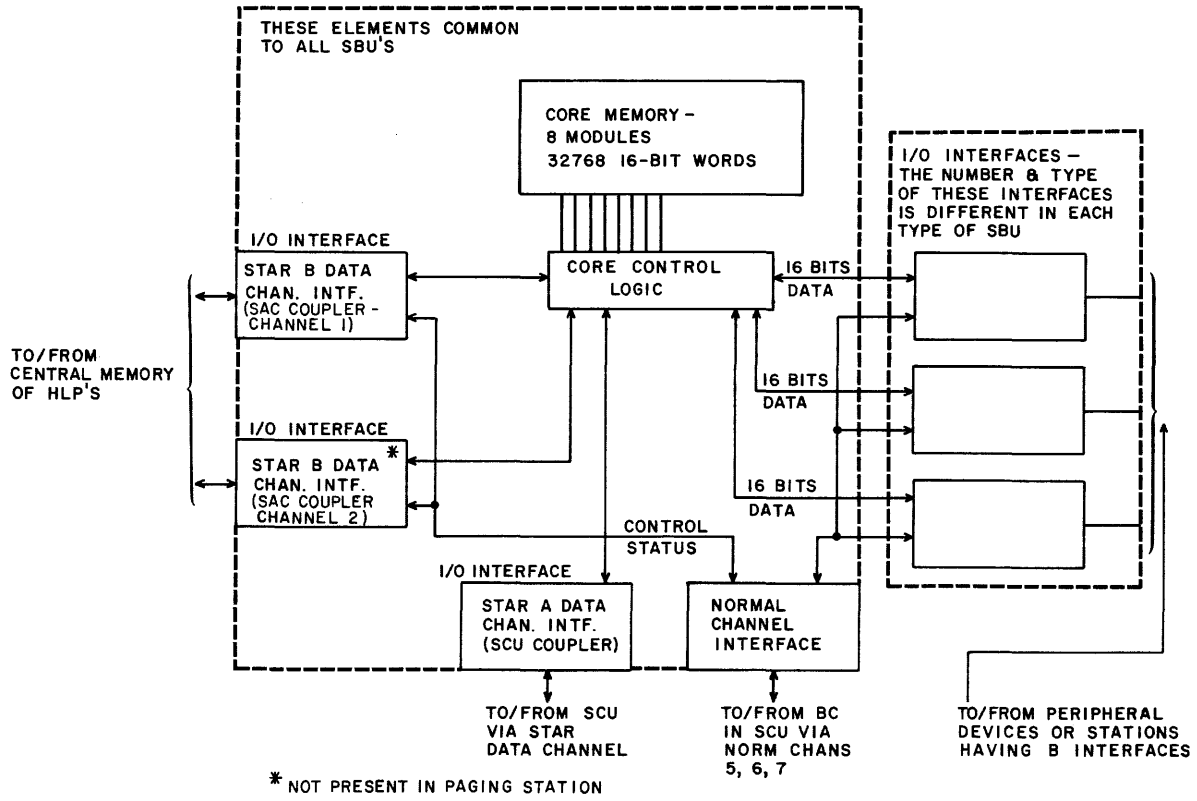


Figure 2-1. SBU Block Diagram

ORGANIZATION OF SBU MEMORY

The SBU memory consists of eight 4096-word modules combined to form 32,768 words of contiguous storage. Memory word length is 18 bits. There are 16 data bits and two parity bits.

The eight memory modules operate independently, permitting memory references to take place concurrently in all eight of the modules. This feature allows overlapping of memory references (bank phasing) to gain an effective memory cycle time much faster than the 1.1-microsecond cycle time of the individual memory modules when the total memory is used in a sequential mode. Overlapping is achieved by assignment of sequential memory addresses to different modules. Immediately after a memory reference has been initiated, a second reference can be initiated for the next sequential location which lies in another module. The second memory reference can proceed before the first is completed. Shortly after the second reference has begun, a reference can be started for the next sequential address which is in a third module, and so on. Eight sequential addresses can be referenced without waiting for completion of previous memory references. The module scanning sequence is fixed. Figure 2-2 shows the timing of the overlapped memory references.

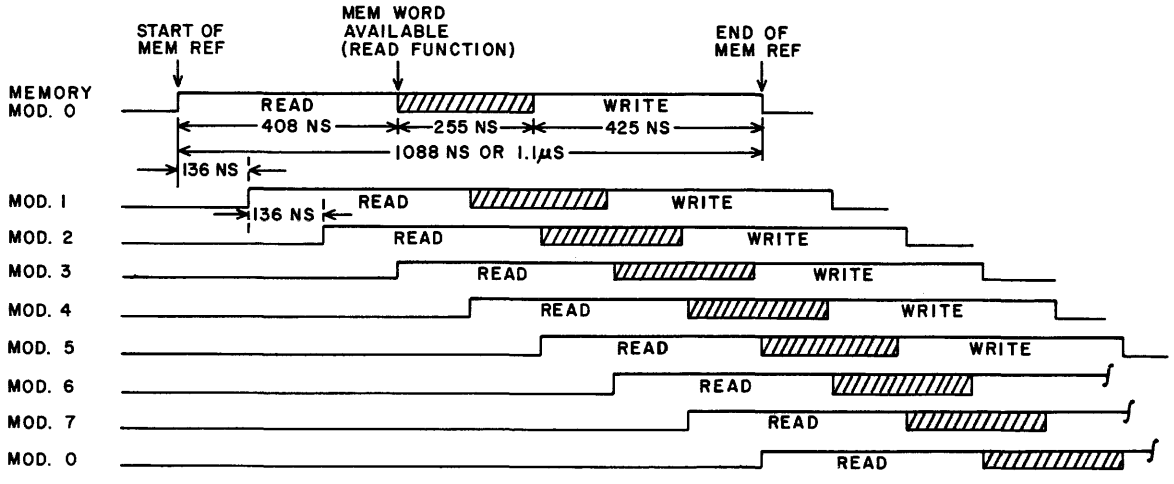


Figure 2-2. Overlapped Memory References

This system yields an effective memory cycle time of 136 nanoseconds during sequential addressing of the SBU. However, during random memory addressing, the effective cycle time is longer. In the worst case, where all memory addresses lie within one module, memory cycle time becomes 1.1 microseconds.

SBU memory is segmented into 16 2048-word pages. Page 0 contains the lowest 2048 addresses; page 1, the next 2048 words, and so on. Some SBU I/O interfaces are page oriented; they are designed to transfer one or more pages of data during each input or output operation.

SBU I/O INTERFACES

The input/output interfaces, present in an SBU, transfer data between peripheral devices and SBU memory.

There are a number of SBU interfaces, each intended for use with a different type of peripheral equipment. The following interfaces are typical of those used in the SBUs.

844 disk interface	Allows 844 disk units to exchange data with SBU memory.
3000 I/O interface	Simulates a standard CDC 3000 I/O channel. Allows 3000-series magnetic tape subsystems to transfer data into or out of SBU memory.
7000 I/O interface	Simulates a standard CDC 7000 I/O channel. Allows 7638-1, 865, and high capacity disk subsystems to transfer data into or out of SBU memory.
STAR B data channel interface	Allows an SBU to transfer data to or from an HLP across a STAR data channel. Acts as the B or active end of a STAR data channel.
STAR A data channel interface	Acts as the A or passive end of a STAR data channel. Allows an SBU to transfer data across a STAR data channel in response to requests from devices equipped with STAR B interfaces.

Several types of SBUs are used in STAR stations. All SBUs contain a 32,768-word memory and interfaces for communication with a companion SCU and HLPs. The major difference between the various SBUs is in the number and type of I/O interfaces. Each type of SBU is intended for use with a certain set of peripheral devices and contains the interfaces necessary to drive these devices.

Two types of interfaces, the STAR B data channel interface and the STAR A data channel interface, are present in all SBUs. The STAR B interface provides a means of transferring data between SBU memory and an HLP over a STAR data channel. To allow communication with two HLPs, most SBUs contain two B interfaces. The STAR A interface allows the SCU that controls each SBU to transfer data into or out of SBU memory.

In addition to I/O interfaces, each SBU contains an interface for three 16-bit normal output channels and three 16-bit normal input channels from the SCU. These normal channel bits connect to the I/O interfaces in an SBU and allow the SCU to control the interfaces.

FUNCTIONS PERFORMED BY SBU INTERFACES

In addition to providing a path between peripheral devices and SBU memory, the SBU interfaces perform a number of other functions related to the input/output process. For example, the 3000 I/O interface performs the following functions in addition to data transfer.

Assembly/disassembly	Converts between 12-bit and 16-bit word size. This process matches the 12-bit word size of the 3000 I/O channel to the 16-bit word size of SBU memory.
Code conversion	Converts 6-bit BCD coded characters from a 3000 peripheral device to 8-bit codes (such as ASCII) before storage in SBU memory. Also performs the reverse conversion during an output operation.
Parity checking	Detects parity errors in data sent to the SBU from a 3000 I/O device.

In some interfaces these additional functions are program-selectable by the SCU.

SCU CONTROL OF SBU INTERFACES

Almost all of the SBU interfaces are controlled by the SCU. In each SBU three BC normal output channels (output channels 5, 6, and 7) are cabled from the companion SCU to the SBU. These three 16-bit output channels allow the SCU to issue control bits and codes to the interfaces in the SBU. Also, three normal input channels (input channels 5, 6, and 7) are cabled to the SBU so that the SCU can read in status information from the I/O interfaces in the SBU. The normal channel control codes, control bits, and status bits are somewhat different for each SBU. Appendixes A and B in this manual give the normal channel bit assignments for each of the standard stations.

In general, the control responsibilities of the SCU are limited to initiating data transfers. To start a data transfer operation, the SCU typically:

Issues a code to select one of the SBU interfaces.

Issues codes and control bits to prepare the interface for a data transfer operation.

Issues a starting address that specifies the beginning of the region in SBU memory where the interface writes or extracts data.

Issues a word count that specifies the amount of data to be transferred.

Issues a code or control bit that causes the interface to begin transferring data.

Once a data transfer operation has begun, the SCU provides no further control. The interface and related peripheral equipment transfers the specified number of 16-bit words without further help from the SCU. During the data transfer operation, the interface automatically increments the memory address to read or write sequential memory addresses. After a block of data has been transferred, the SCU can read in status information from the interface to determine if the operation progressed normally.

After the SCU has initiated a data transfer operation, it is free to select another interface and start data transfer through this second interface. The SCU can then select other interfaces and initiate data transfer operations. Thus, several interfaces within an SBU can transfer data concurrently into or out of SBU memory. To avoid conflicts within SBU memory, it is the responsibility of the SCU program to assign different buffer areas to each interface.

SELECTION OF INTERFACES BY SCU

Before the SCU can issue control bits to an SBU interface or read in status information from an interface, the SCU must issue a four-bit connect code to select the interface. Within an SBU each I/O interface is assigned a unique connect code. When the SCU issues the connect code for an interface, the selected interface connects to the six SCU normal input and output channels, and all other interfaces within the SBU disconnect from the normal channels. The SCU can then issue control information to the selected interface or read in status information from this interface. The selected interface remains connected to the normal channels only as long as the SCU continues to issue the connect code.

In all stations, the SCU issues connect codes on normal output channel 6, bits C through F. Table 2-1 lists the connect codes for all standard stations.

SBU INTERFACE STATUS INFORMATION

Two types of status information is available from each SBU, interface status and scanner status.

INTERFACE STATUS

Each interface within an SBU provides a number of status bits which indicate various operating conditions within the interfaces. Busy, operation finished, parity error, and abnormal termination are typical of the conditions reported by these status bits. Status from an interface is available, on normal input channels 5, 6, and 7, only when the interface is selected by the SCU. Status bit assignments for all standard stations are listed in appendixes A and B.

SCANNER STATUS

Scanner status is a single 16-bit status word that gives an overall status indication for an SBU. It provides one or more status bits for each of the interfaces within an SBU. A major purpose of the scanner status word is to show which interfaces are busy and which ones have finished data transfer operations.

The SCU requests scanner status in one of two ways, by issuing a special scanner connect code or by issuing a scanner select bit. Some SBUs require a connect code while others employ a scanner select bit. In response, the SBU places the scanner status word on one of the normal input channels. Scanner status remains available as long as the SCU maintains the scanner connect code or scanner select bit on the normal channels. As with interface connect codes, the SCU issues the scanner connect code on normal output channel 6, bits C through F. Table 2-1 lists the scanner connect code or select bit for each standard station.

The normal input channel used for scanner status is not the same for all SBUs. Appendix A shows the input channel used for scanner status in each station and indicates the bit assignments within the scanner status word.

MEMORY PRIORITIES

The SBU contains a priority system that resolves conflicts resulting from simultaneous memory requests received from two or more I/O interfaces. Each memory access channel is assigned a priority ranging from 0 to 11 where 0 carries the highest priority. A priority network examines each of the 12 MAC requests and enables the one with the highest priority. Interface priorities have been predetermined for each station and are assigned a priority number within the station (Table 2-2). Priorities are assigned so that interfaces which serve high-speed, fixed-data-rate devices (such as disk and drum units) are given high priority. This arrangement ensures that SBU memory will be able to supply or accept data at the rate dictated by the fixed-rate peripheral devices.

TABLE 2-1. SBU CONNECT CODES

STATION / INTERFACE	Paging (Drum)	Disk	844 Service	844 Storage	Magnetic Tape	Storage (Media)	STAR-100 Service	STAR-1B Service
7000-0	2_{16}	2_{16}	NA	NA	NA	NA	2_{16}	NA
7000-1	1_{16}	3_{16}	NA	NA	NA	NA	NA	NA
SAC-1	0_{16}	0_{16}	0_{16}	1_{16}	1_{16}	1_{16}	0_{16}	1_{16}
SAC-2	NA	1_{16}	1_{16}	3_{16}	3_{16}	3_{16}	1_{16}	3_{16}
Index compare	3_{16}	NA	NA	NA	NA	NA	NA	5_{16}
SCU coupler	†	†	3_{16}+ NOC-6, bit 8	†	†	†	3_{16}+ NOC-6, bit 8	7_{16}+ NOC-6, bit 1
844-A	NA	NA	2_{16}	0_{16}	0_{16}	NA	NA	NA
844-B	NA	NA	††	2_{16}	2_{16}	NA	NA	NA
Network trunk	NA	NA	†††	NA	NA	NA	NA	NA
Peripheral A interface group	NA	NA	3_{16}	NA	NA	NA	3_{16}	7_{16}
3000-0	NA	NA	NA	C_{16}	C_{16}	C_{16}	NA	NA
3000-1	NA	NA	NA	D_{16}	D_{16}	D_{16}	NA	NA
Scanner	NOC-6, bit B	NOC-6, bit B	NOC-6, bit B	6_{16}	6_{16}	6_{16}	NOC-6, bit B	NOC-6, bit 8
MDD-A (841)	NA	NA	NA	NA	NA	0_{16}	NA	0_{16}
MDD-B (841)	NA	NA	NA	NA	NA	2_{16}	NA	2_{16}

NA = not applicable

† = Connected to dedicated channel (either NC -2 or -4) between SBU coupler in SCU and SCU coupler in SBU

†† = 2 more 844's optional

††† = Controlled by remote terminal

TABLE 2-2. SBU MEMORY PRIORITIES

Priority Level	Interface
Paging (Drum) Station	
0	7000 channel 1
1	7000 channel 0
2	SAC channel 1
3	SCU coupler
4	Index compare
Disk Station	
0	7000 channel 0
1	7000 channel 1
2	SCU coupler
3	SAC channel 1
4	SAC channel 2
844 Service Station	
0	844-A
1	Network trunk 1
2	Network trunk 2
3	Network trunk 3
4	Network trunk 4
5	SCU coupler
6	SAC channel 1
7	SAC channel 2
8	Peripheral A No. 1
9	Peripheral A No. 2
10	Peripheral A No. 3
11	Peripheral A No. 4

Priority Level	Interface
STAR-100 Service Station	
0	7000 channel 0
1	SCU coupler
2	SAC channel 1
3	SAC channel 2
4	Peripheral A No. 1
5	Peripheral A No. 2
6	Peripheral A No. 3
7	Peripheral A No. 4
8	Peripheral A No. 5
9	Peripheral A No. 6
10	Peripheral A No. 7
11	Peripheral A No. 8
STAR-1B Service Station	
0	MDD-A (841)
1	MDD-B
2	Index compare
3	SCU coupler
4	SAC channel 1
5	SAC channel 2
6	Peripheral A No. 1
7	Peripheral A No. 2
8	Peripheral A No. 3
9	Peripheral A No. 4
10	Peripheral A No. 5
11	Peripheral A No. 6
844 Storage Station	
0	844-A
1	844-B
2	3000 channel 1
3	3000 channel 2
4	SCU coupler
5	SAC channel 1
6	SAC channel 2

TABLE 2-2. SBU MEMORY PRIORITIES (Cont'd)

Priority Level	Interface
Storage (Media) Station	
0	MDD-A (841)
1	MDD-B
2	3000 channel 0
3	3000 channel 1
4	SCU coupler
5	SAC channel 1
6	SAC channel 2

Priority Level	Interface
Magnetic Tape Station	
0	844-A
1	844-B
2	3000 channel 1
3	3000 channel 2
4	SCU coupler
5	SAC channel 1
6	SAC channel 2

SBU PERIPHERAL A INTERFACE

NOTE

In the illustrations and in text, the term peripheral A interface is used to differentiate the logic connected to the B interface(s) of other stations from the A interface which connects to the B interface in the SCU.

The SBU peripheral A interface provides a data path between a STAR data channel and SBU memory. It acts as the A end (passive end) of a STAR data channel. Peripheral A interfaces are used in the STAR service stations to communicate with second level peripheral stations that contain STAR B interfaces. Figure 2-3 shows how the peripheral A interface fits into a system.

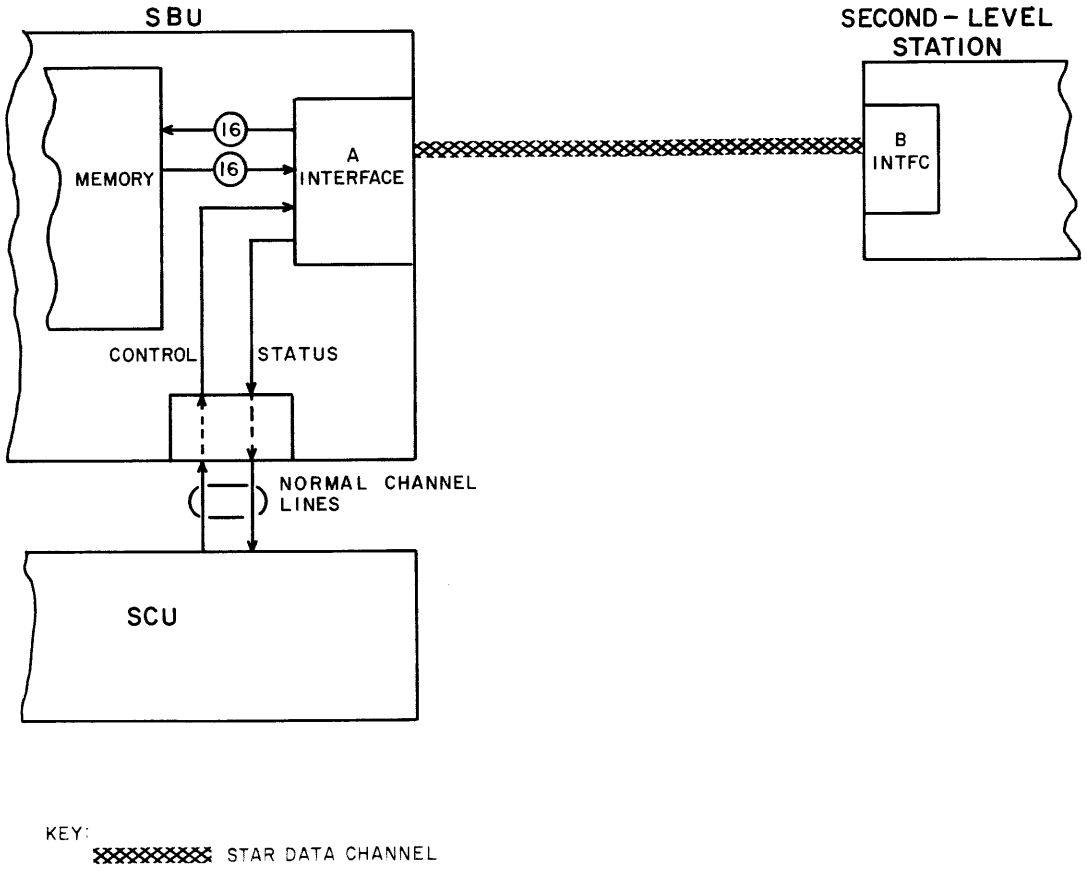


Figure 2-3. Peripheral A Interface System Relationship

Several peripheral A interfaces can be present in the same SBU. The peripheral A interfaces are almost entirely independent of each other and all can transfer data concurrently.

DATA TRANSFER OPERATIONS

The A interface can perform two data transfer operations.

Block read	Transfers a block of 16-bit data words from SBU memory to the B end of the channel.
Write	Transfers a block of 16-bit data words, sent from the B end of the channel, into SBU memory.

In both cases, the data transfer operation is initiated by the B end of the STAR data channel.

All data blocks transferred on the STAR data channel must be a multiple of 32 bits. Two 16-bit words are thus the smallest unit of data that can be transferred.

BLOCK READ OPERATION

The B end of the channel starts a block read operation by sending two items of information, a function code which specifies a block read operation, and a SBU starting address that specifies the lower boundary of the data source area in SBU memory.

In response, the A interface extracts data words from sequential locations in SBU memory and transmits them to B one after the other. The operation continues until B stops accepting data.

WRITE OPERATION

The B end of the channel initiates a write operation by sending a write function code and a SBU starting address that specifies the lower boundary of the data receiving area in SBU memory. B then begins to send a series of 16-bit data words. In response, the A interface writes the block of data into sequential locations in SBU memory. The operation continues until B stops sending data.

NORMAL CHANNEL CONTROL OF THE A INTERFACES

Unlike most other SBU interfaces, SCU normal channel bits play only a limited role in controlling the peripheral A interface. The data transfer operations, write and block read, are initiated and controlled by the B end of the STAR data channel. The A interface carries out these operations without any control from normal channel bits.

Normal channel bits associated with the A interface are used for the following purposes.

Program clear	A program clear bit stops the operation in progress and clears the operating mode established by commands from the B interface.
Parity status	A normal input channel bit associated with each peripheral A interface allows the SCU to sense SBU parity errors. Three bank count bits indicate the memory bank in which the parity error occurred.
Control function to B	Two normal channel bits allow the SCU to send 2-bit control function codes to the B end of the channel.
Interrupt from B	A normal input channel bit associated with each peripheral A interface allows the SCU to sense interrupts sent from second level stations (at B end of the STAR channels).

The normal channel bits are defined in detail in Table 2-5, near the end of this interface description.

FUNCTION CODES

Two types of function codes are used to control operations on the STAR data channel.

Operation function codes	Sent to the A interface from the B end of the channel. These codes are used to initiate and control data transfer operations.
System control functions	Codes sent from the SCU at the A end of the channel, through the A interface, to the B end of the channel. These codes provide a means for the SCU to coordinate the activities of the lower level stations with system software activities.

OPERATION FUNCTION CODES

Table 2-3 lists the codes sent from B.

TABLE 2-3. OPERATION FUNCTION CODES

Name	CODE
Write	010_2
Block read	101
Read	001
Special function	011 †
Data	100
Null	000
End of operation	111
Note: Code 110 is illegal. † Applies to STAR-65 only	

Write (010_2)

This code prepares the A interface for a write operation. It also causes the A interface to read in the first word of a two-word starting address that B sends along with this function code.

Block read (101)
Read (001)

The effect of these two codes is identical. They prepare the A interface for a block read operation. They also cause the A interface to read in the first word of a two-word starting address that B sends along with these codes.

Special function (011_2)

This code is used by the STAR-65 maintenance station to initiate special functions other than normal input/output operations. The exact meaning of this code must be determined by the users at each end of the channel prior to its use.

Data (100)

This code does not specify an operation. It identifies the accompanying 16-bit word sent from B as the first (most significant) part of a 32-bit data word. Used only during write operations.

Null (000) — This code does not specify an operation. It identifies the accompanying 16-bit word sent from B as the second part of a 32-bit starting address or data word.

End of operation (111) — This code notifies the A interface that the current block read or write operation is complete.

Illegal code (110) — This code should not be used. If issued, the A interface returns an illegal response signal to the B end of the channel.

SYSTEM CONTROL FUNCTIONS

The burden of initiating and controlling I/O operations on the STAR channel lies with the peripheral station at the B end of the channel. This relieves the processor (SCU) at the A end of the channel of much I/O overhead. However, there are situations that require A to send I/O control information to B. System control function codes (Table 2-4) provide a method for A to signal B.

TABLE 2-4. SYSTEM CONTROL FUNCTION CODES

Name	Code
Channel flag	10_2
External flag	01
Suspend	11
Invalid	00

The SCU in the service station uses two normal channel bits to send these codes to B through the A interface.

Channel flag (10_2) — This code informs the B end of the channel that the system software has placed a message for B in a predetermined area in SBU memory at the A end of the channel. B normally responds to this code by initiating a block read operation to read in the message.

External flag (01_2)	This code directs B to master clear and then enter an autoload sequence.
Suspend (11_2)	This code causes B to terminate a write or block read operation.
Invalid (00)	The service station SCU should not transmit this code. It appears on the channel only if a transmission fault occurs in one of the valid codes. The B end of the channel should assume that a suspend code was intended. It should terminate the data transfer operation in progress and then send an interrupt to A to signal the malfunction.

STARTING ADDRESS FORMAT

The STAR data channel procedure requires that two 16-bit address words (Figure 2-4) be sent from B to A at the beginning of a write or block read operation. The A interface uses only the second address word since only 14 address bits are required to address the entire SBU memory (16K of 32-bit words). A two-word address must be sent to satisfy channel requirements but only one is used by the A interface.

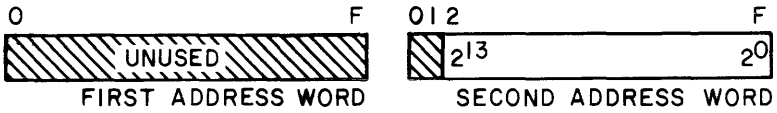


Figure 2-4. STAR Channel Starting Address Format

Memory addressing on the STAR channel is always based on a memory word size of 32 bits. Thus, the SBU memory is regarded as 16K of 32-bit words even though actual SBU memory word size is 16 bits. The A interface always transfers data in multiples of two 16-bit memory words.

STAR CHANNEL TRANSMISSION SEQUENCE

Function codes and data transferred on the STAR channel must follow a fixed sequence. The required sequences for write and block read operations are described in the following paragraphs. Note that each 16-bit unit of information transferred on the data path must be accompanied by a function code.

WRITE SEQUENCE

	Function Code (B to A)	Data Path
Step 1	Write code (010_2)	1st address word (B to A)
Step 2	Null code (000)	2nd address word (B to A)
Step 3	Data code (100)	16-bit data word (B to A)
Step 4	Null code (000)	16-bit data word (B to A)
	Repeat steps 3 and 4 for each additional 32-bit unit of data transferred from B to A.	
Step 5	End of operation code (111)†	Not used
Step 6	Null code (000)	Not used

BLOCK READ SEQUENCE

	Function Code (B to A)	Data Path
Step 1	Block read code (101_2)	1st address word (B to A)
Step 2	Null code (000)	2nd address word (B to A)
Step 3	None	16-bit data word (A to B)
Step 4	None	16-bit data word (A to B)
	Repeat steps 3 and 4 for each additional 32-bit unit of data transferred from A to B.	
Step 5	End of operation code (111)†	Not used
Step 6	Null code (000)	Not used

†If another data transfer operation follows immediately, the end of operation code is not necessary. A new write or block read terminates the previous operation.

PARITY CHECKING

Two types of parity checking take place during A interface operations.

FUNCTION CODE PARITY CHECK

The A interface checks each function code, sent from B, for odd parity. If A detects a function parity error, it notifies B by sending a parity error signal and the illegal signal.

ADDRESS AND DATA PARITY CHECK

The A interface checks each address or data word, sent from B, for odd parity. If A detects a parity error, it notifies B by sending a parity error signal.

SBU MEMORY PARITY CHECK

During a block read operation, the A interface checks parity for each 16-bit data word read from SBU memory. If a parity error occurs, A notifies B by sending a parity error signal. The peripheral A interface also reports an SBU parity error to the SCU on a normal channel bit. A 3-bit bank count, which indicates the SBU memory bank in which a parity error occurred, is also available to the SCU on normal channel bits.

When a parity error occurs, the block read operation continues unless stopped by the B end of the channel. However, the A interface does not accept a function code for a new operation unless B first sends an end of operation code or the SCU issues a program clear to the A interface on the normal channels.

INTERRUPTING THE A INTERFACE

The STAR data channel includes an interrupt line on which B can send an interrupt signal to A. The interrupt signal is used to facilitate communication between B and the SCU system software at the A end of the channel. Communication from B to A consists of two steps.

1. B writes a message for A into a predefined area in SBU memory at the A end of the channel.
2. B sends an interrupt signal to notify A that a message has been placed in SBU memory.

The A interface reports the interrupt signal to the SCU on a normal channel bit.

One such normal channel bit is available for each peripheral A interface in the SBU. It is the responsibility of the system software to periodically scan these interrupt bits and to interpret and act on messages placed in memory by B.

NORMAL CHANNEL CONTROL AND STATUS BITS

Figure 2-5 shows the SCU normal channel bits associated with the peripheral A interfaces. Each bit is defined in Table 2-5. The channel and bit assignments given are for the STAR 100 service station which contains eight peripheral A interfaces. Channel and bit assignments for peripheral A interfaces used in other stations may be different. Refer to appendix A near the end of this manual for a complete set of normal channel bit assignments for all STAR stations.

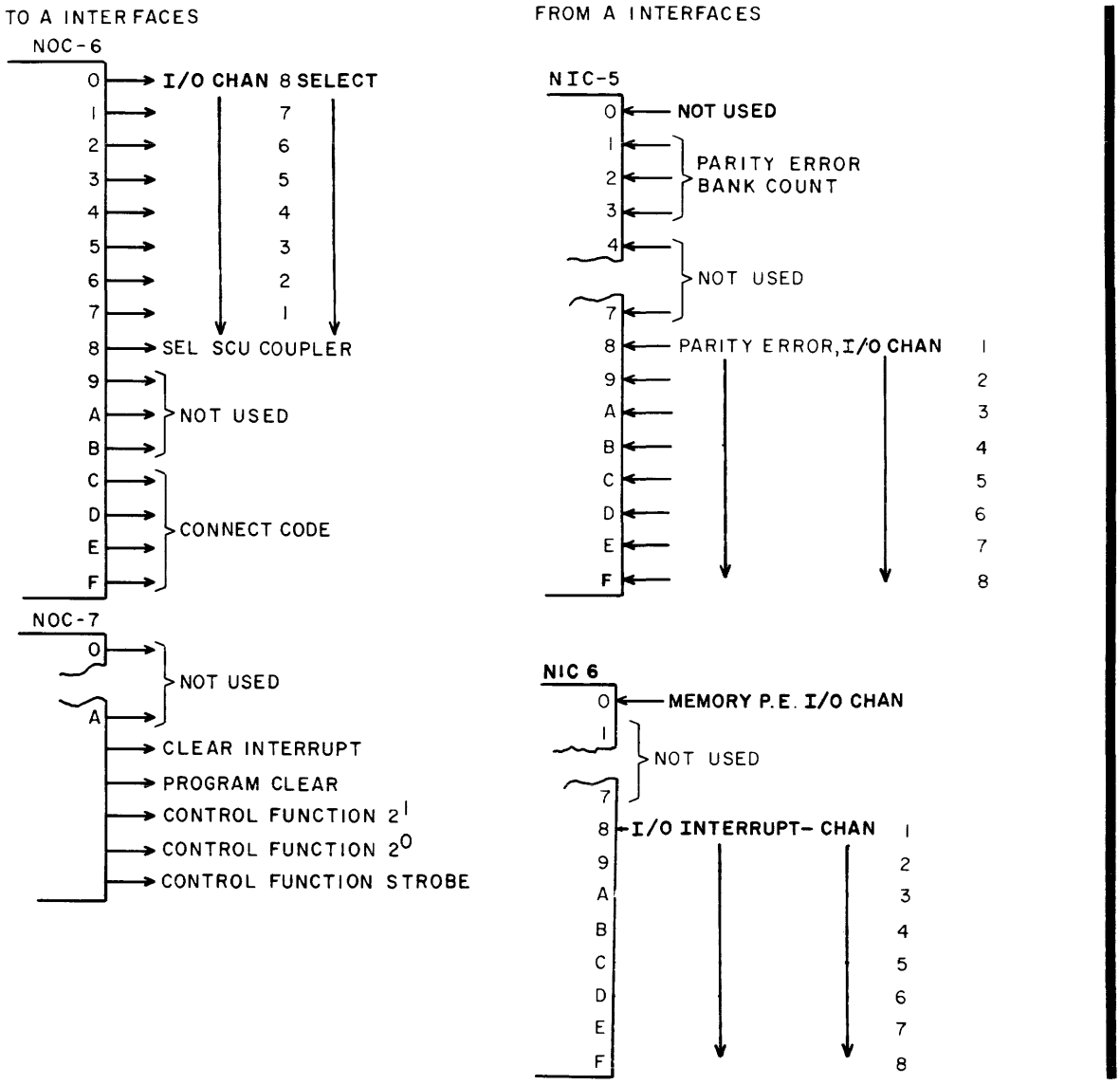


Figure 2-5. Normal Channel Bits for Peripheral A Interface

TABLE 2-5. NORMAL CHANNEL BIT DEFINITIONS FOR PERIPHERAL A INTERFACE

Channel	Bit	Name	Function
NOC-6	0	Sel. periph. intf. 8	These bits select one of the A interfaces for SCU control. When an interface is selected, the SCU can: 1. Send control function codes to B through the selected A interface. 2. Issue program clear and clear interrupt commands to the selected A interface. 3. Read in the parity error bank count from the selected interface.
NOC-6	1	7	
NOC-6	2	6	
NOC-6	3	5	
NOC-6	4	4	
NOC-6	5	3	
NOC-6	6	2	
NOC-6	7	1	
NOC-6	$C(2^3)$ ↓ $F(2^0)$	Connect code	These bits determine which interface or group of interfaces in the SBU is connected to the SCU normal channels. (Refer to Table 2-1 for peripheral A interface connect codes.)
NOC-7	B	Clear interrupt	Setting and then clearing this bit clears the interrupt from B status bit for the selected interface.
NOC-7	C	Program clear	Setting and then clearing this bit clears the following items in the selected A interface. 1. The operation selection (write or block read). 2. The SBU parity error status bit. 3. The parity error bank count.
NOC-7	D	Control function bit 2^1	These two bits are used for transmitting a control function code to B through the selected A interface. Codes:
NOC-7	E	Control function bit 2^0	

Bit D (2^1)	Bit E (2^0)	Function
0	0	Invalid
0	1	External flag
1	0	Channel flag
1	1	Suspend

TABLE 2-5. NORMAL CHANNEL BIT DEFINITIONS FOR A INTERFACE

Channel	Bit	Name	Function
NOC-7	F	Control function strobe	Setting this bit causes the selected A interface to transmit a control function code (NOC-7, bits E and F) to B.
NIC-5	0	Memory parity error, SCU coupler	When 1, this bit indicates that an SBU memory parity error has occurred during a block read operation conducted by the SCU coupler.
NIC-5	1	Parity error bank counter 2^2	For the selected interface, these bits indicate the SBU memory bank in which a parity error has occurred. They apply to the SCU coupler as well as the peripheral A interfaces. These bits are meaningful only if one of the parity error status bits is 1.
NIC-5	2	Parity error bank counter 2^1	
NIC-5	3	Parity error bank counter 2^0	
NIC-5	8	Mem. parity error, interface 8	When 1, this bit indicates that a memory parity error has occurred during a block read operation conducted by A interface 8.
NIC-5	9	Mem. parity error, interface 7	
NIC-5	A	Mem. parity error, interface 6	
NIC-5	B	Mem. parity error, interface 5	
NIC-5	C	Mem. parity error, interface 4	
NIC-5	D	Mem. parity error, interface 3	
NIC-5	E	Mem. parity error, interface 2	
NIC-5	F	Mem. parity error, interface 1	

PROGRAMMING SEQUENCES

The following paragraphs indicate the required programming sequence for those A interface activities that can be controlled by the SCU. Normal channel bit assignments referenced are for the STAR 100 service station.

1. Connect Peripheral Interfaces

   Place the appropriate connect code in NOC-6, bits C through F (code 3_{16} for STAR 100 service station).

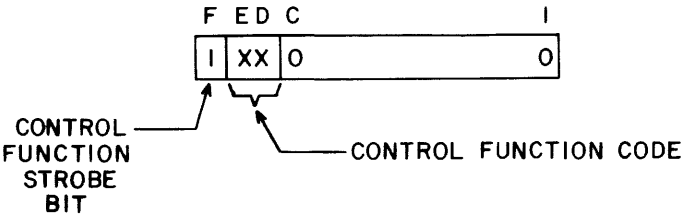
   The SCU remains connected to the group of peripheral A interfaces as long as the connect code is held in NOC-6.

2. Transmit Control Function Code to B

   a. Connect according to step 1.

   b. Select desired A interface (NOC-6, bits 0 through 7).

   c. Set following pattern in A register.



   d. Output from A to NOC-7.
   This causes the selected A interface to transmit the code.

   e. NOC-7 may be cleared immediately.
   Note: This operation does not apply to the SAC coupler.

3. Clear Interrupt

   a. Connect according to step 1.

   b. Select desired peripheral A interface (NOC-6, bits 1 through 8).
   Note: This operation does not apply to the SAC coupler.

   c. Set and then clear interrupt bit (NOC-7, bit B).

4. Program Clear

   a. Connect according to step 1.

   b. Select desired peripheral A interface (NOC-6, bits 1 through 8).

   c. Set and then clear program clear bit (NOC-7, bit C)
   Note: SAC coupler can be program cleared as well as the peripheral A interfaces.

5. Sense Parity Error Status

   a. Connect according to step 1.

   b. Input to A from NIC-5.

      Note: Bit 0 of A indicates SCU coupler parity error.
      Bits 8 through F indicate peripheral A interface parity errors.

6. Read Parity Error Bank Count Status

   a. Connect according to step 1.

   b. Select desired interface (NOC-6, bits 0 through 8).

   c. Input to A from NIC-5.
      Bank count appears in bits 1 through 3.

      Note: Parity error bank count can be read for SAC coupler as well as the peripheral A interfaces.

SCU COUPLER INTERFACE (STAR A DATA CHANNEL INTERFACE)

The SCU coupler provides a data path between SBU memory and an SCU. It is present in all SBUs. Figure 2-6 shows how the SCU coupler fits into a STAR peripheral station.

The data path between an SBU and SCU is a STAR data channel with the SCU coupler acting as the A end of the channel. In the SCU, the SBU coupler is the B end of the STAR channel.

The SCU coupler is nearly identical to the peripheral A interfaces used in STAR service stations, but does not include the capability for transmitting control function codes back to the B end of the channel.

DATA TRANSFER OPERATIONS

The SCU coupler can perform two data transfer operations.

Block read	Transfers a block of 16-bit data words from SBU memory to the B end of the channel.
Write	Transfers a block of 16-bit data words, sent from the B end of the channel, into SBU memory.

In both cases, the data transfer operation is initiated by the B end of the STAR data channel (the SBU coupler in the SCU).

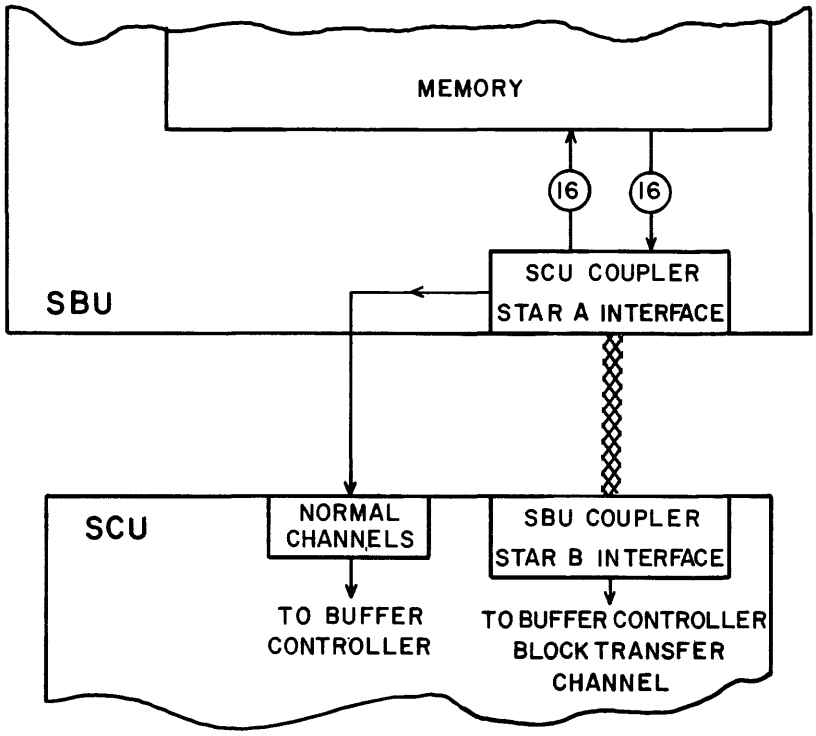


Figure 2-6. SCU Coupler System Relationship

All data blocks transferred on the STAR data channel must be a multiple of 32 bits. Two 16-bit words are thus the smallest unit of data that can be transferred.

BLOCK READ OPERATION

The B end of the channel starts a block read operation by sending the following two items of information to the SCU coupler.

A function code which specifies a block read operation

An SBU starting address that specifies the lower boundary of the data source area in SBU memory.

The B end then begins to request data. In response, the SCU coupler extracts data words from sequential locations in SBU memory and transmits them to B one after the other. The operation continues until B stops requesting data.

WRITE OPERATION

The B end of the channel initiates a write operation by sending a write function code and an SBU starting address that specifies the lower boundary of the data receiving area in SBU memory. B then begins to send a series of 16-bit data words. In response, the SCU coupler writes the block of data into sequential locations in SBU memory. The operation continues until B stops sending data.

NORMAL CHANNEL CONTROL OF THE SCU COUPLER

Unlike most other SBU interfaces, SCU normal channel bits are not used to control the SCU coupler. Both data transfer operations, write, and block read are initiated and controlled by the B end of the STAR data channel. The SCU coupler carries out these operations without any control from normal channel bits. In most applications, the only use of normal channel bits for the SCU coupler is to provide parity error status information. These normal input channel bits are defined later on.

OPERATION FUNCTION CODES

Operation function codes are sent from the B end of the channel (SCU) to the SCU coupler in the SBU. They are used to initiate and control data transfer operations.

These codes provide a means for the SCU to coordinate the activities of the lower level stations with system software activities.

Table 2-6 lists the codes sent from B.

TABLE 2-6. SCU COUPLER OPERATION FUNCTION CODES

Name	Code
Write	010_2
Block read	101
Read	001
Special function	011
Data	100
Null	000
End of operation	111
Note: Code 110 is illegal	

Write (010_2) — This code prepares the SCU coupler for a write operation. It also causes the SCU coupler to read in the first word of a two-word starting address that B sends along with this function code.

Block read (101)
Read (001) — The effect of these two codes is identical. They prepare the SCU coupler for a block read operation. They also cause the SCU coupler to read in the first word of a two-word starting address that B sends along with these codes.

Special function — This function is used by the B end of the channel for initiating a special control or diagnostic function provided both B and A have the facilities for its implementation. Specifically, this function is used for performing a special function other than normal input/output. The meaning of the function is established before its use.

Data (100) — This code does not specify an operation. It identifies the accompanying 16-bit word sent from B as the first (most significant) part of a 32-bit data word. Used only during write operations.

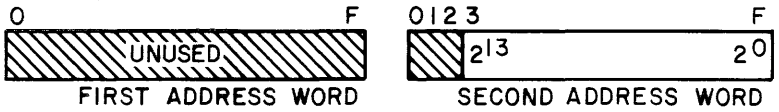
Null (000) — This code does not specify an operation. It identifies the accompanying 16-bit word sent from B as the second part of a 32-bit starting address or data word.

End of operation (111) — This code notifies the SCU coupler that the current read or write operation is complete.

Illegal code (110) — This code should not be used. If issued, the SCU coupler returns an illegal response signal to the B end of the channel.

STARTING ADDRESS FORMAT

The STAR data channel procedure requires that two 16-bit address words be sent from B to A at the beginning of a write or block read operation. The SCU coupler uses only the second address word since only 14 address bits are required to address the entire SBU memory (16K of 32-bit words). Two words must be sent to satisfy channel requirements but only one is used by the SCU coupler.



Memory addressing on the STAR channel is always based on a memory word size of 32 bits. Thus, the SBU memory is regarded as 16K of 32-bit words even though actual SBU memory word size is 16 bits. The A interface always transfers data in multiples of two 16-bit memory words.

STAR CHANNEL TRANSMISSION SEQUENCE

Function codes and data transferred on the STAR data channel must follow a fixed sequence. The required sequence for write and block read operations is described in the following paragraphs. Note that most 16-bit units of information transferred on the data path must be accompanied by a function code.

WRITE SEQUENCE

	Function Code (B To A)	Data Path
Step 1	Write code (010_2)	First address word (B to A)
Step 2	Null code (000)	Second address word (B to A)
Step 3	Data code (100)	16-bit data word (B to A)
Step 4	Null code (000)	16-bit data word (B to A)
	Repeat steps 3 and 4 for each additional 32-bit unit of data transferred from B to A.	
Step 5	End of operation code (111)†	Not used
Step 6	Null code (000)	Not used

†If another data transfer operation follows immediately, the end of operation code is not necessary. A new write or block read code terminates the previous operation.

BLOCK READ SEQUENCE

	Function Code (B To A)	Data Path
Step 1	Block read code (101_2)	First address word (B to A)
Step 2	Null code (000)	Second address word (B to A)
Step 3	None	16-bit data word (A to B)
Step 4	None	16-bit data word (A to B)
	Repeat steps 3 and 4 for each additional 32-bit unit of data transferred from A to B.	
Step 5	End of operation code (111)†	Not used

PARITY CHECKING

Two types of parity checking take place during SCU coupler operations.

FUNCTION CODE PARITY CHECK

The SCU coupler checks each function code sent from B for odd parity. If the SCU coupler detects a function parity error, it notifies B by sending a parity error signal and the illegal signal.

ADDRESS AND DATA PARITY CHECK

The A interface checks each address or data word sent from B for odd parity. If A detects a parity error, it notifies B by sending a parity error signal.

†If another data transfer operation follows immediately, the end of operation code is not necessary. A new write or block read code terminates the previous operation.

SBU MEMORY PARITY CHECK

During a block read operation, the SCU coupler checks parity for each 16-bit data word read from SBU memory. If a parity error occurs, the SCU coupler notifies B by sending a parity error signal on the STAR channel. Also, the SCU coupler sends a 3-bit bank count that indicates the SBU memory bank in which a parity error occurred to the SCU via normal channel bits.

When a parity error occurs, the block read operation continues unless stopped by the B end of the channel. However, the A interface does not accept a function code for a new operation unless B first sends an end of operation code or the SCU issues a program clear to the A interface.

NORMAL CHANNEL STATUS BITS

In most applications, the only normal channel bits associated with the SCU coupler are three normal input channel bits that carry a parity error bank address for the SCU coupler. The parity error bank address bits indicate the SBU memory bank in which a parity error occurred during a block read operation. They are meaningful only after the SCU coupler has signaled the SCU, via the STAR data channel, that a parity error has occurred. The three address status bits remain valid until the SCU issues an end of operation function code to the SCU coupler.

The normal input channel bit assignments for these status bits are not the same for all stations. In most cases, they occupy three bits of the scanner status word or three bits of one of the normal channel feedback words. Normal channel assignments for each STAR station are listed in appendix A of this manual.

In the three STAR service stations, the normal channel bits for the SCU coupler are a little different than in other stations. Several peripheral A interfaces are present in the service stations. Since the SCU coupler and peripheral A interfaces are quite similar, they share some of the same normal channel bits. The normal channel bits that apply to the SCU coupler in a service station are listed in Table 2-7.

TABLE 2-7. NORMAL CHANNEL BIT DESCRIPTIONS FOR SCU COUPLER INTERFACE

Channel	Bit	Name	Function
NOC-6	D (2^2) E (2^1) F (2^0)	Connect code	Three normal output channel bits connect the A interface group within the SBU to the normal channel lines. The A interface group includes the SCU coupler and the peripheral A interfaces. Additional select bits are used to select one interface within the group.
NOC-6	8 (2^7)	Select SCU coupler	This normal output channel bit selects the SCU coupler as the A interface connected to the normal channel lines. It must be used along with the connect code.
NOC-7	C (2^3)	Program clear	This normal output channel bit clears the selected A interface. Setting and then clearing this bit clears the following items in the SCU coupler. 1. The operation selection (write or block read) 2. The parity error status bit 3. The parity error bank address
NIC-5	0 (2^{15})	Memory parity error SCU coupler	This normal input channel bit is a status bit. When 1, it indicates that an SBU memory parity error has occurred during a block read operation conducted by the SCU coupler.
NIC-5	1 (2^{14}) 2 (2^{13}) 3 (2^{12})	Parity error bank address (three bits)	These three normal input channel bits are status bits shared by the SCU coupler and peripheral A interfaces. For the selected interface, they indicate the SBU memory bank in which a parity error has occurred.

STAR B DATA CHANNEL INTERFACE (SAC COUPLER)

The SAC coupler is an SBU interface that controls data transfer operations between a STAR peripheral station and an HLP. Within a peripheral station the coupler allows data to be transferred to or from either the SBU memory or the SCU (via normal channels). Thus, the coupler permits two basic classes of data transfer operations.

Transfers between SBU memory and HLP memory.

Transfers between SCU normal channels and HLP memory.

Figure 2-7 shows the relationship between the SAC coupler and the system elements with which it communicates.

The SAC coupler communicates with the HLP over a STAR data channel and is designated the B or active end of the channel. The HLP is the A or passive end of the channel. Within the HLP, the source or destination of data transferred over the STAR data channel is the HLP memory. All transfers over the STAR data channel are in multiples of 32-bit words. A single 32-bit word is the smallest unit of transfer.

As the active end of the channel, the peripheral station must initiate all data transfer operations. This involves sending the HLP a function code that specifies the type of operation along with an address designating the starting point within HLP memory.

Operation of the SAC coupler is controlled by the SCU via normal channels 5, 6, and 7. The principal control responsibilities of the SCU are:

Issue HLP memory address.

Issue starting and terminating addresses that define the limits of the data transfer within SBU memory.

Issue function code to specify type of operation (read or write).

Specify the data source/destination (SBU memory or SCU) within the peripheral station.

Transmit or read-in each data word (only when the SCU is the data source/destination).

Monitor status conditions.

Issue end of operation function code.

Issue program clear to clear certain error or operating conditions.

The term HLP refers to either a STAR central computer or a higher level STAR peripheral station (such as the three STAR service stations) which contain a STAR A interface.

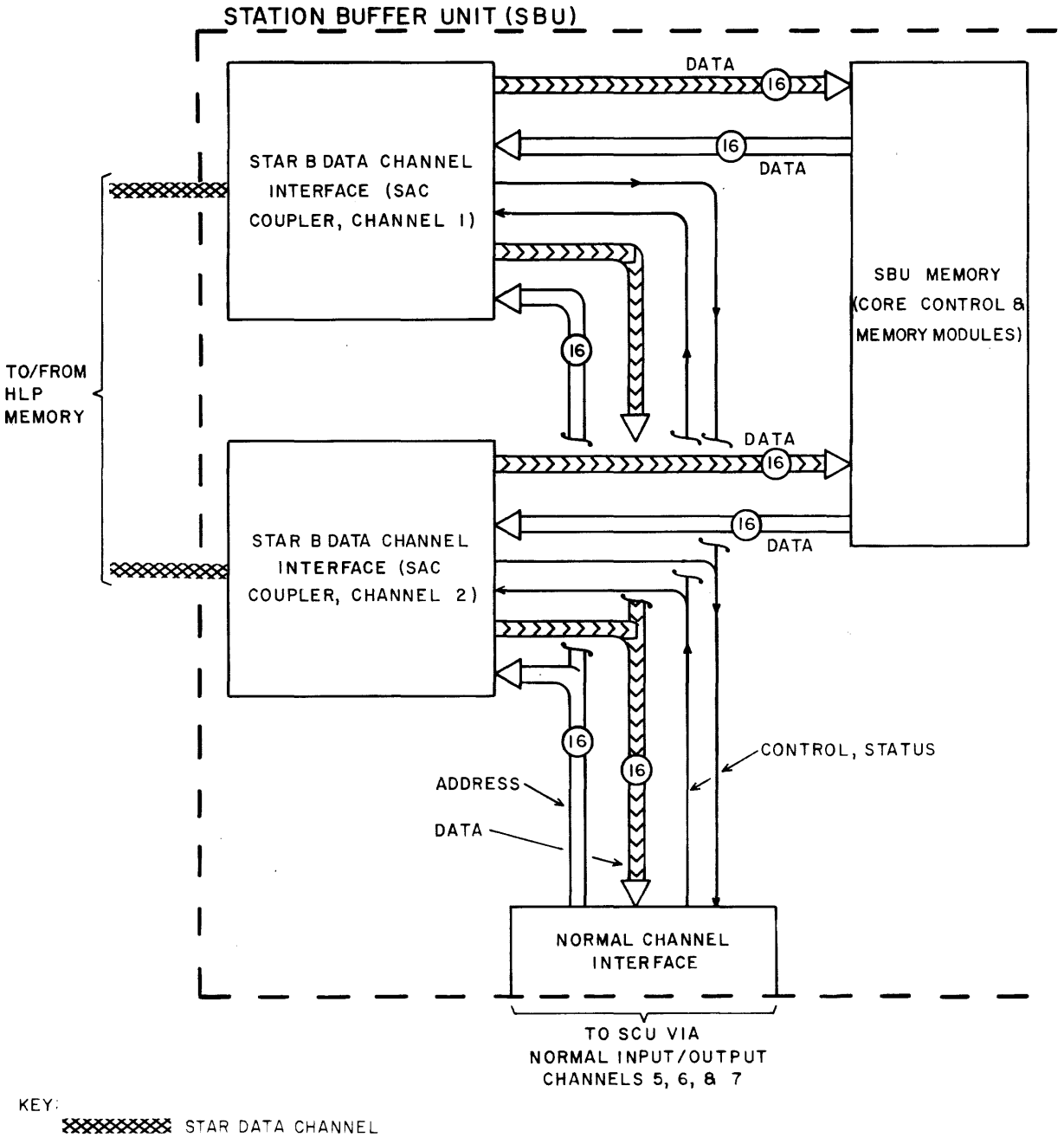


Figure 2-7. SAC Coupler System Relationship

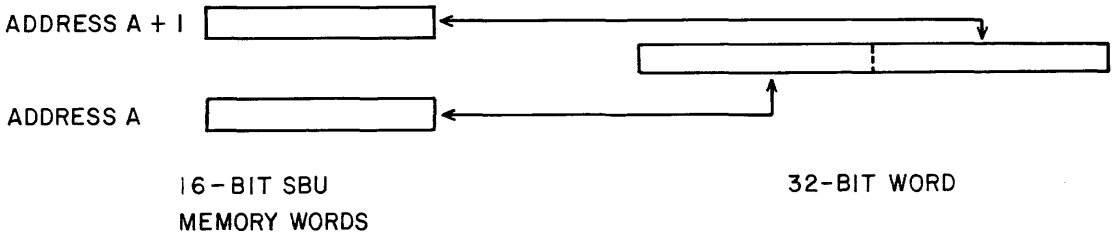
Most STAR peripheral stations contain two SAC couplers. They are assigned different connect codes† for the purpose of selection by the SCU. The two couplers are completely independent and can perform simultaneous data transfer operations, but not at the maximum data transfer rate.

SBU/HLP DATA TRANSFER OPERATIONS

Two types of transfers are possible between SBU memory and the HLP memory.

Block read	Transfers a block of 32-bit data words from HLP memory to SBU memory.
Write	Transfers a block of 32-bit data words from SBU memory to HLP memory.

Each 32-bit word consists of two consecutive 16-bit SBU memory words as shown:



Before the operation begins, the SCU must issue three memory addresses to the SAC coupler.

SBU starting address - 15 bits

SBU terminating address - 15 bits

HLP starting address - 21 bits

The two SBU addresses define the limits of the read or write operation; when the terminating address is reached, the SAC coupler automatically terminates the operation. The HLP starting address defines the beginning of the buffer area in HLP memory.

The 21-bit HLP memory address is sent from the SCU in two parts, the upper 5 bits first followed by the remaining 16 bits. Along with the first part of the address, the SCU sends a 3-bit function code (101_2, block read or 010_2, write) to specify the type of operation and an additional function bit to select HLP-to-SBU memory mode.

† Refer to Table 2-1 for SAC coupler connect codes.

The SAC coupler forwards the address and 3-bit function code to the HLP and then immediately starts the data transfer operation. Normally the operation continues until the SBU terminating address is reached, although there are several conditions that terminate a data transfer early.

After a data transfer operation has been initiated, the SCU need not remain connected to the SAC coupler except to check status or terminate an operation early. The SAC coupler then transfers the entire block of data defined by the starting and terminating addresses without further control from the SCU.

SCU/HLP DATA TRANSFER OPERATIONS

The SAC coupler allows two types of data transfers between the SCU and HLP memory.

Block read	Transfers a block of 32-bit data words from HLP memory to normal input channel 7.
Write	Transfers a block of 32-bit data words from normal output channel 5 to HLP memory.

Each 32-bit word consists of two successive 16-bit normal channel words, the first of the two words being the most significant. During these operations, the BC (in the SCU) must individually transmit or read in each 16-bit word with an output-from-A or input-to-A instruction. The SCU must thus remain connected to the SAC coupler throughout an entire data transfer operation.

Before actual data transfer begins, the SCU must issue the following information to the SAC coupler.

1. 21-bit HLP memory starting address (two successive output words on normal output channel 5).
    a. Upper 5 bits
    b. Lower 16 bits
2. Issue four function bits
    a. 3-bit function code selects type of operation (write, 010_2; block read, 101_2)
    b. 1-bit selects HLP-to-SCU mode

The coupler automatically forwards the HLP address and 3-bit function code to the HLP. At this point, both coupler and HLP are conditioned for a data transfer operation. The SCU must now execute a series of output-from-A or input-to-A instructions to transfer a block of data words. Normally, the SCU issues an end-of-operation function code to terminate the operation.

FUNCTION CODES

Two types of function codes control activities on the STAR data channel, operation function codes (sent from the peripheral station to HLP) and system control functions (sent from the HLP to peripheral station).

OPERATION FUNCTION CODES

The function codes listed in Table 2-8 are sent from the SCU to the HLP through the coupler.

TABLE 2-8. SAC COUPLER OPERATION FUNCTION CODES

Function	Code
Write	010_2
Block read	101
Read	001
Null (no operation)	000
End of operation	111
Data†	100
Illegal codes	110
	011

Write (010_2) — This code conditions the coupler and HLP for a peripheral station-to-HLP data transfer operation.

Block read (101_2) — This code conditions the coupler and HLP for an HLP memory-to-peripheral station data transfer operation.

† The STAR data channel requires that the data function code (100_2) be sent along with the first part of each 32-bit data word. The SAC coupler automatically issues this code; no program step is required to transmit it.

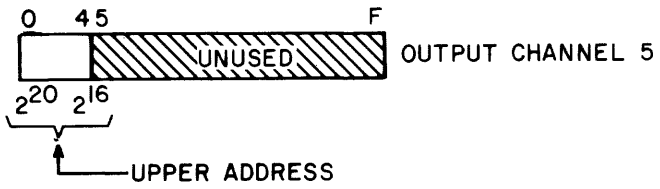
Read (001_2)	Identical to block read (101_2).
Null (000_2)	This code does not specify an operation. It is transmitted along with the second part of the 21-bit HLP address to inform the HLP that the lower part of an address is on the channel.
End of operation (111_2)	This code causes the coupler and HLP to terminate a data transfer operation.
Illegal codes (110_2, 011_2)	These codes should not be used. If issued, no operation is initiated, and the HLP returns an illegal response signal that sets a coupler status bit.

The SCU issues the operation function codes on normal output channel 7, bits D, E, and F.

The SCU must issue two supporting items of information along with the write (010_2), block read (101_2), and read (001_2) function codes.

SCU/SBU select bit (NOC-7, bit C). This bit selects either the SBU memory or SCU as the data source/destination. Logical 1 selects the SBU and logical 0 selects SCU.

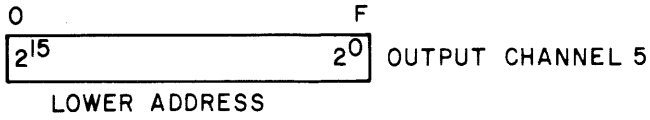
Upper five bits of the HLP starting address (normal output channel 5).



No supporting information is required with the end-of-operation function code.

After a function code and supporting information have been placed in the appropriate normal channels, the SCU must pulse a function strobe bit (normal output channel 6, bit 2). Setting and then immediately clearing this bit transfers the information from the normal channels into the coupler. In turn, the coupler forwards the function code, and if present, the address to the HLP. The function code and supporting information need not be held in the normal channels after transfer to the coupler.

Following a write (010_2), block read (101_2), or read (001_2) function code, the SCU must issue a null function code (000) and lower 16 bits of HLP address (on normal output channel 5).



The function strobe bit transfers the null function code and address to the coupler. Again, the coupler automatically forwards the code and address to the HLP.

If an HLP/SBU memory operation has been selected (SCU/SBU select bit equals 1, the second function code (null) causes the coupler and HLP to begin transferring data. If an HLP/SCU operation has been selected (SCU/SBU select bit equals 0), the SCU must individually read-in or transmit each data word.

SYSTEM CONTROL CODES

The system control codes listed in Table 2-9 provide a means for the HLP to send control information to the SAC coupler.

TABLE 2-9. SAC COUPLER SYSTEM CONTROL CODES

Function	Code
Channel flag	01
External flag	10
Suspend	11
Invalid	00

The two control function bits appear as status bits on normal input channel 5. Two of the control codes directly affect operation of the peripheral station hardware.

Channel flag (01) — This code sets the channel flag status bit in the SBU scanner status word. It is used to indicate that a message for the peripheral station has been placed in a prearranged area in HLP memory.

External flag (10) — This code forces the SCU to master clear and autoload from the microdrum.

Suspend (11) — This code causes the SAC coupler to terminate a data transfer operation.

Invalid (00) — The HLP does not transmit this code. It appears on the channel only if a transmission fault occurs. If this code appears, the peripheral station software should assume that a suspend code was intended; it should terminate the data transfer operation in progress and then send an interrupt to inform the HLP of the malfunction.

The control function code is captured by two status bits that are available on normal input channel 5 (bits B and C). Associated with the two bits is a control function strobe status bit (normal input channel 5, bit D) that indicates that the HLP has sent a control function code. The two control function status bits are significant only when the function strobe status bit is set.

PARITY CHECKING

Three types of parity checking take place during SAC coupler operations.

ADDRESS AND DATA PARITY CHECK

The SAC coupler generates an odd parity bit for each 8 bits of data transmitted to the HLP.

The SAC coupler checks the parity of each 8 bits of data coming from the HLP and notifies the HLP of any errors.

FUNCTION CODE PARITY

The coupler appends an odd parity bit to the 3-bit function code sent to the HLP. If the HLP detects a function parity error, it ignores the function code and returns a signal to the coupler that sets the transmission parity error status bit. This status bit is available to the SCU on normal input channel 5.

HLP READ PARITY †

During block read operations, data read from HLP memory is checked for proper parity. If a parity error occurs, the HLP signals the coupler. In response, the coupler terminates the read operation and sets the SAC parity error status bit which is available to the SCU on normal input channel 5.

SBU MEMORY PARITY CHECK

During SBU memory-to-HLP write operations, data read from SBU memory is parity checked. If a parity error occurs, the coupler terminates the write operation and sets the SBU parity error status bit. This status bit appears on normal input channel 5.

† Does not apply where the STAR 100 Computer is the HLP.

STATUS

Three types of status information from the SAC coupler are available to the SCU.

NORMAL INPUT CHANNEL 5 STATUS

Several miscellaneous status bits, such as coupler busy, parity error, etc., appear on normal input channel 5. Table 2-11 defines each of these status bits.

RETURN ADDRESS STATUS

During HLP/SBU memory block read or write operations, the current SBU memory address is available to the SCU on normal input channel 7.

SCANNER STATUS

Two SAC coupler status bits not busy and SAC flag, are available as part of the SBU scanner status word that appears on normal input channel 6. In SBUs that contain two SAC couplers, the scanner status word contains a pair of these status bits for each coupler. The bits are labeled as follows:

SAC-1	Not busy	coupler 1
SAC-1	Flag	
SAC-2	Not busy	coupler 2
SAC-2	Flag	

Not busy — Logical 1 indicates that the coupler is not engaged in a data transfer operation.

SAC flag — Logical 1 indicates that the HLP has sent a channel flag control code.

TERMINATING CONDITIONS

Several conditions cause the SAC coupler to terminate a data transfer operation.

SBU terminating address	During an SBU memory/HLP data transfer operation, the operation terminates when the terminating address is reached.
End of operation function code†	The SCU can terminate a data transfer operation by issuing this function code.
New function code†	If the SCU issues a block read (101_2) or write (010_2) function code while a data transfer operation is in progress, the current operation terminates and the coupler begins the new operation.
Parity error	Two types of parity errors terminate coupler operations. SBU memory parity error (terminates an SBU memory-to-HLP write operation) HLP memory parity error (terminates an HLP-to-SBU memory write operation or HLP-to-SCU write operation)
Suspend control function code	This code, sent from the HLP, terminates either a write or block read operation.
External flag control function code	This code, sent from the HLP, causes the SCU to master clear and autoload. The SCU, in turn, master clears the SBU and thus terminates the operation of all SBU interfaces.

INTERRUPTING THE HLP

The SAC coupler provides a means for the SCU to interrupt the HLP. A normal channel bit (NOC-6, bit 7) controls the interrupt; setting and then clearing this bit causes the coupler to send an interrupt signal to the HLP. The interrupt signal sets an external interrupt flag in the HLP.

† These two methods should not be used during HLP/SBU memory transfers. The operation causes an abnormal termination. To restore the channel to normal operation, initiate a manual master clear from the SCU and the HLP.

SCU CONTROL OF THE SAC COUPLER

Tables 2-10 and 2-11 define the SCU normal channel bits used for control of the SAC coupler. Bit assignments given in the table are for the disk station. SAC coupler normal channel bit assignments are different for some signals in other stations. Appendix A of this manual contains a full set of normal channel assignments for all STAR stations.

The SAC coupler must remain connected to the SCU in order to receive control information or data from the SCU. The coupler remains connected as long as the SCU holds the unique connect code, assigned to the coupler, on output channel 6, bits C through F.

TABLE 2-10. BIT DESCRIPTIONS FOR SAC COUPLER/NOC-5, -6, AND -7 INTERFACES (DISK STATION)

Channel	Bit	Name	Function
NOC-5	0 ↓ F	TA/SCU Addr/data bit 2^{15} ↓ TA/SCU Addr/data bit 2^{0}	These bits serve two purposes. 1. Transmit SBU terminating address to selected SAC coupler. 2. Transmit SCU address and data to HLP through selected SAC coupler.
NOC-6	0(2^{15})	Starting address SBU	This set/clear† bit is used to strobe the starting address, transmitted by NOC-7 into the SAC coupler.
NOC-6	1(2^{14})	Term. address SBU	This set/clear bit is used to strobe the SBU terminating address, transmitted by NOC-5, into the SAC coupler.
NOC-6	2(2^{13})	Function	This set/clear bit transfers three items of control information from the SCU into the SAC coupler. 1. Function code on NOC-7, bits D, E, and F 2. SBU memory/SCU select bit on NOC-7, bit C 3. HLP starting address word on NOC-5 The coupler then forwards the function code and address word to the HLP.

†A set/clear bit must be set and immediately cleared to perform its intended function.

TABLE 2-10. BIT DESCRIPTIONS FOR SAC COUPLER/NOC-5, -6, AND -7 INTERFACES (Cont'd)
(DISK STATION)

Channel	Bit	Name	Function
NOC-6	$3(2^{12})$	Program clear	This set/clear bit transfers the following control bits from the SCU into the SAC coupler. 1. Clear parity error, NOC-7, bit 7 2. Clear illegal, NOC-7, bit 8 3. Clear control function, NOC-7, bit 9 4. Master clear, NOC-7, bit A
NOC-6	$4(2^{11})$	Request	During SCU to HLP write operations, this set/clear bit causes the coupler to transfer a data word from the SCU (via NOC-5) to the HLP.
NOC-6	$5(2^{10})$	Accept	This bit acknowledges that the SCU has received one word of data on NIC-7. Prior to this the SCU must have received the data word when the first rank full staus bit was true.
NOC-6	$6(2^{9})$	Select status	This bit is used to determine whether data or a return address is made available on NIC-7. When bit is set, return address is available; when bit is clear, data is available.
NOC-6	$7(2^{8})$	Set interrupt	This set/clear bit provides a means of interrupting the central processor (via the SAC channel) and sets the external interrupt bit in the HLP.
NOC-6	$C\text{-}2^{3}$ ↓ ↓ $F\text{-}2^{0}$	Connect code $(2^{3} - 2^{0})$	These bits determine which interface in the SBU is connected to the SCU normal channels. (Refer to Table 2-1 for SAC coupler connect codes.)

TABLE 2-10. BIT DESCRIPTIONS FOR SAC COUPLER/NOC-5, -6, AND -7 INTERFACES (Cont'd)
(DISK STATION)

Channel	Bit	Name	Function
NOC-7	0 ↓ 6	SA bit 2^{15} ↓ SA bit 2^{9}	These bits as well as bits 7 through F are used to transmit the SBU starting address information to the SAC coupler. Bits 7 through F serve a dual purpose.
NOC-7	7	SA bit 2^{8}/clear parity error	Used for clearing the three parity error bits (NIC-5, bits 0, 4, and 5) or for bit 2^{8} of the starting address of a memory location in SBU core memory.
NOC-7	8	SA bit 2^{7}/clear illegal	Used for clearing the illegal status bit (NIC-5, bit 6) or for bit 2^{7} of the starting address.
NOC-7	9	SA bit 2^{6}/clear cont. function	Used for clearing the two control function status bits (NIC-5, bits B and C) or for bit 2^{6} of the starting address.
NOC-7	A	SA bit 2^{5}/MC	Used for transmitting the master clear bit to the SBU or for starting address bit 2^{5}.
NOC-7	B		Used only for starting address bit 2^{4}.
NOC-7	C	SA bit 2^{3}/select SAC + SCU	Other than being used as one bit of the starting address, this bit is used to select a data path between the HLP and the SCU or between the HLP and the SBU. When this bit is a 0, the HLP to SCU path is selected and when it is a 1, the HLP to SBU path is selected.
NOC-7	D ↓ F	SA/func. code bit 2^{2} ↓ SA/func. code bit 2^{0}	These bits are used for transmitting bits 2^{0} through 2^{2} of the SBU starting address information to the SAC coupler or for transmitting the desired function code to the HLP via the SAC coupler.

TABLE 2-10. BIT DESCRIPTIONS FOR SAC COUPLER/NOC-5, -6, AND -7 INTERFACES (Cont'd)
(DISK STATION)

Channel	Bit	Name	Function
			Codes: Function / 2^2 / 2^1 / 2^0 Null 0 0 0 Read 0 0 1 Write 0 1 0 Block read 1 0 1 EOP 1 1 1 Data 1 0 0

TABLE 2-11. BIT DESCRIPTIONS FOR SAC COUPLER/NIC-5, -6, AND -7 INTERFACES

Channel	Bit	Name	Function
NIC-5	$0(2^{15})$	SBU parity error	This status bit is used for indicating a parity error in SBU memory.
NIC-5	$1(2^{14})$ $2(2^{13})$ $3(2^{12})$	$\overline{\text{Bank counter } 2^2}$ $\overline{\text{Bank counter } 2^1}$ $\overline{\text{Bank counter } 2^0}$	If a parity error occurs, these three bits indicate the not SBU memory bank count. To find the memory bank with the parity error, complement the bank counter and count back four banks.
NIC-5	$4(2^{11})$	Transmission parity error	This status bit indicates that the HLP has detected a parity error in a function code transmitted from the SCU via the SAC coupler.
NIC-5	$5(2^{10})$	SAC parity error	This bit is used to indicate that a parity error occurred in the HLP memory during a read operation.
NIC-5	$6(2^{9})$	Illegal	This status bit indicates that the HLP has: 1. Detected an illegal function (011_2 or 110_2) sent from the SCU, or 2. Detected a parity error in a function code sent from the SCU.

TABLE 2-11. BIT DESCRIPTIONS FOR SAC COUPLER/NIC-5, -6, AND -7 INTERFACES (Cont'd)

Channel	Bit	Name	Function
NIC-5	$7(2^8)$	Busy	This bit sets when the SCU transfers an address to the HLP at the beginning of a read or write operation. It remains set until the operation terminates.
NIC-5	$8(2^7)$	Term. address compare	This bit is used to indicate that the latest HLP to SCU memory read or write operation is complete.
NIC-5	$9(2^6)$	First rank full	This status bit indicates that half of a 32-bit data word from the HLP (read operations only) is available on NIC-7 for read-in by the SCU.
NIC-5	A(10) (2^5)	Last rank full	This bit indicates a word is in the last rank of buffering in the SAC interface.
NIC-5	B(11) (2^4) C(12) (2^3)	Control function 2^1 Control function 2^0	These two bits are control signals which the HLP uses to communicate with the SAC coupler and are coded as follows:
NIC-5	D(13)2^2	Control function strobe	This status bit indicates that the HLP has sent a control function code to the SAC coupler. The SCU can read in the control function on NIC-5, bits B and C.

Bit B(Fctn 2^1)	Bit C(Fctn 2^0)	Function
0	0	Invalid
0	1	Channel flag
1	0	External flag
1	1	Suspend

TABLE 2-11. BIT DESCRIPTIONS FOR SAC COUPLER/NIC-5, -6, AND -7 INTERFACES (Cont'd)

Channel	Bit	Name	Function
NIC-7	0 ↓ F	Data/RA bit 2^{15} ↓ Data/RA bit 2^{0}	These bits serve two purposes. 1. Receive data from HLP during HLP-to-SCU block read operations. 2. Receive return address information. (The return address represents the last SBU address referenced by the SAC coupler.)
		Scanner Status	
NIC-6	2	SAC-1 flag	HLP has sent a channel flag control function code to SAC coupler number 1.
NIC-6	3	SAC-2 flag	HLP has sent a channel flag control function to SAC coupler number 2.
NIC-6	4	SAC-1 not busy	SAC coupler number 1 is not engaged in data transfer.
NIC-6	5	SAC-2 not busy	SAC coupler number 2 is not engaged in data transfer.

PROGRAMMING SEQUENCES

The following procedures are to be used to aid the programmer during the preparation of the program for conducting data transfers between the SCU and the HLP via the SAC coupler.

CONNECTING PROCEDURE

1. Place the appropriate connect code in bits C through F of NOC-6 to select the SAC coupler. (Refer to Table 2-1 for proper code.)
2. Check status (NIC-5) and check for error conditions.
3. If errors are detected, proceed to Error Recovery Sequence procedure.

ADDRESSING THE SBU

1. If an SBU memory-to-HLP data transfer operation is desired, proceed with step 2. If the SCU desires to transfer data to or from the HLP via either of the SAC couplers, proceed to Addressing the HLP procedure.
2. Place the SBU starting address in bits 0 through F of NOC-7.
3. Set and then clear the starting address line (bit 0 of NOC-6). This bit strobes the starting address word into the SAC coupler.
4. Place the SBU data last word address in bits 0 through F of NOC-5.
5. Set and then clear terminating address line (bit 1 of NOC-6). This bit strobes the terminating address word into the SAC coupler.

ADDRESSING THE HLP

1. Place the upper 5-bit portion of the SAC address in bits 0 through 4 of NOC-5.
2. Place the appropriate logical state for the SBU/SCU select bit in bit C of NOC-7 and function code such as read, write, etc. in bits D through F of NOC-7. Select bit C and function codes are as follows:

Bit C = 0; HLP-to-SCU communications path selected

Bit C = 1; HLP-to-SBU memory path selected

	Function Codes		
Function	Bit 2^2(D)	Bit 2^1(E)	Bit 2^0(F)
Null	0	0	0
Read	0	0	1
Write	0	1	0
Block Read	1	0	1
EOP	1	1	1
Data	1	0	0

3. Set and then clear SBU function select bit (bit 2 of NOC-6). This bit notifies the HLP that an address word and function code are on the channel.
4. Wait for the last buffer rank full status to equal zero (NIC-5, bit A). This bit indicates that the HLP has accepted the first half of the address word placed in NOC-5 in step 1.
5. Place the lower 16-bit portion of the SAC address in bits 0 through F of NOC-5.
6. Place a null function code in bits D through F of NOC-7.
7. Set and then clear SBU function select bit (NOC-6, bit 2). This bit notifies the HLP that an address word and function code are on the channel and also starts the data transfer operation between the HLP and SAC coupler subject to the following conditions.

If the HLP/SBU memory operation has been selected in step 2, the SAC coupler automatically transfers the number of words specified by the starting and terminating addresses.

If the HLP/SCU operation has been selected in step 2, the SCU must transmit or receive each word separately.

8. Wait for last buffer rank full status to equal zero (NIC-5, bit A) and test status lines for errors. This bit indicates that the HLP has accepted the second address word.
9. If an error is detected, proceed to Error Recovery Sequence procedure.
10. If the SCU/HLP transfer operation was selected in step 2, proceed to SCU/SAC Channel Data Transfer. If not, continue with step 11.
11. Check the state of the SAC-1 or SAC-2 not busy bit (NIC-6, bit 8 or 9, in the scanner status word).

NOTE

One method of checking the busy bit is to adjust the toggle mask of the scanner subroutine to monitor the busy bit and wait for the hardware to complete the data transfer. (Also, refer to Testing Status For Completion of Data Transfer.)

SCU/SAC CHANNEL DATA TRANSFER

1. If the data transfer operation is an input to the SCU (block read), proceed to step 8. For an SCU to HLP output (write) operation, continue with step 2.
2. Place the output data in NOC-5.
3. Set and then clear the SCU request bit (NOC-6, bit 4). This bit causes the SAC coupler to transfer the data word from the SCU to the HLP.
4. Wait for last buffer rank full status to equal zero (NIC-5, bit A) and test the status bits of NIC-5 for errors.
5. If an error is detected, proceed to Error Recovery Sequence procedure.
6. If data transfer process is not complete, go to step 2 and repeat procedure.
7. If data transfer is complete, proceed to End of Operation procedure.
8. (Output operation continued from step 1.) Set and then clear the SCU accept bit (NOC-6, bit 5).
9. Wait for first buffer rank full status to equal one (NIC-5, bit 9) and test NIC-5 status bits for errors.

10. Read in data word on normal input channel 7.
11. If an error is detected, proceed to Error Recovery Sequence procedure.
12. If data transfer is not complete, return to step 8 and continue procedure. If data transfer is complete, proceed to End of Operation sequence.

END OF OPERATION

1. Transmit the end of operation function (code 7 using bits D through F, NOC-7).
2. Set and then clear function bit (NOC-6, bit 2).
3. Return to step 11 of Addressing the HLP procedure to start another data transfer operation.

TESTING STATUS FOR COMPLETION OF DATA TRANSFER

1. Check that the SAC not busy signal is a 1, via bit 8 or 9 of NIC-6 (depending upon the SAC channel selected). If signal is not a 1, proceed to Error Recovery Sequence procedure.
2. Check status bits of NIC-5 for proper completion of data transfer. If an error is detected, proceed to Error Recovery Sequence procedure.
3. Exit to calling subroutine.

ERROR RECOVERY SEQUENCE

1. If this is the third retry, set abort (bit 0 of software control package) status and exit to calling subroutine.
2. Check status bits B and C of NIC-5 for presence of suspend, invalid, or external flag (autoload) control codes from the HLP. If any of these are present, set abort status and exit to calling subroutine.
3. Set master clear via bit A of NOC-7 (SCU to SBU).
4. Set and then clear program clear bit (NOC-6, bit 3).
5. Return to Connecting Procedure and try again for successful data transfer.

3000 I/O CHANNEL INTERFACE

The 3000 SBU interface simulates a standard CDC 3000 series 12-bit I/O channel. It controls data transfers between 3518/3528 magnetic tape controllers and SBU memory. Figure 2-8 shows the relationship between the 3000 interface and other system elements.

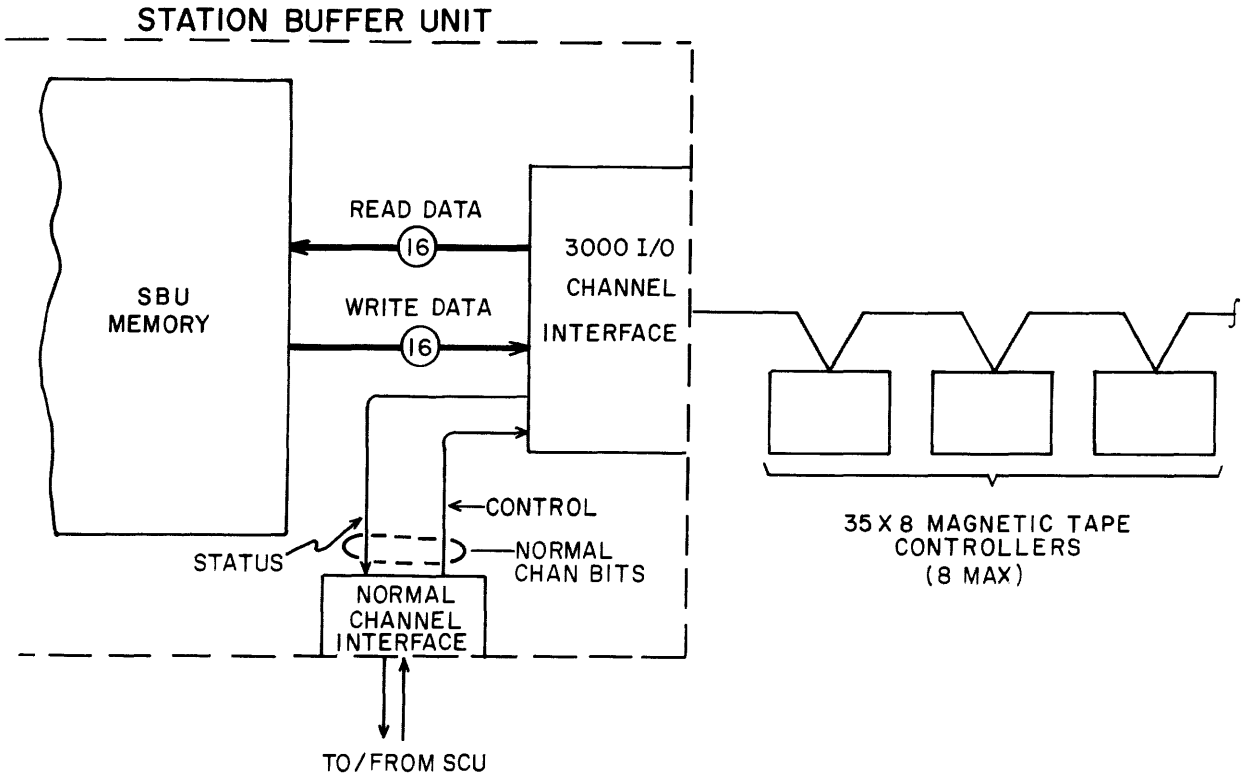


Figure 2-8. 3000 Interface System Relationship

Two 3000 interfaces can coexist in the same SBU. They are assigned different connect codes for the purpose of selection by the SCU. The two 3000 interfaces are completely independent and can perform simultaneous data transfer operations.

A major difference between the 3000 interface and standard 3000 series I/O channels is the number of interrupt lines available. Standard I/O channels can handle eight interrupt lines, one from each of eight peripheral controllers. The 3000 interface is limited to four interrupt lines. Thus, no more than four magnetic tape controllers can be connected to the 3000 interface if the interrupt feature is used.

This section describes the programming characteristics of the 3000 interface only; it does not cover any of the peripheral controllers that can be connected to the 3000 interface. The reader should be familiar with 3000 I/O philosophy and programming conventions in order to fully understand this material.

MAIN FUNCTIONS

The principal functions of the 3000 interface are:

Issues connect code to 3000 controllers.

Issues function codes to 3000 controllers.

Data transfer control; governs exchange of data between peripheral controllers and SBU memory.

Assembly/disassembly; during input operations, breaks 12-bit input bytes into 4-bit segments and packs four of these segments into each 16-bit SBU memory word. During output operations, the interface reverses the process.

Code conversion; converts 6-bit characters into equivalent 8-bit characters and vice versa. This program-selectable feature can be used to convert between BCD and ASCII codes.

Status input; provides a path from the 3000 controller to the SCU for a 12-bit controller status word.

Interrupt input; provides a means for the SCU to inspect interrupt lines from up to four 3000 controllers.

The SCU controls all of these functions by issuing control information on normal output channels 5 and 6. Full-time attention of the SCU is not required, however. Once an input or output operation has been initialized, the 3000 interface transfers the specified number of data words without further support from the SCU.

DATA TRANSFER OPERATIONS

All data transfer operations involve the exchange of a block of data words between SBU memory and the 3000 controller. Before the SCU initiates a data transfer operation, it must send an SBU starting address to specify the beginning of the buffer area in memory. For write operations, a terminating address must also be specified.

The six types of data transfer operations are:

Read binary

Write binary

Read coded

Write coded

Read backward (binary)

Read backward (coded)

READ BINARY

During this operation, 12-bit data bytes from a 3000 controller are broken into 4-bit segments; these segments are then packed into 16-bit SBU words. Figure 2-9 illustrates the process.

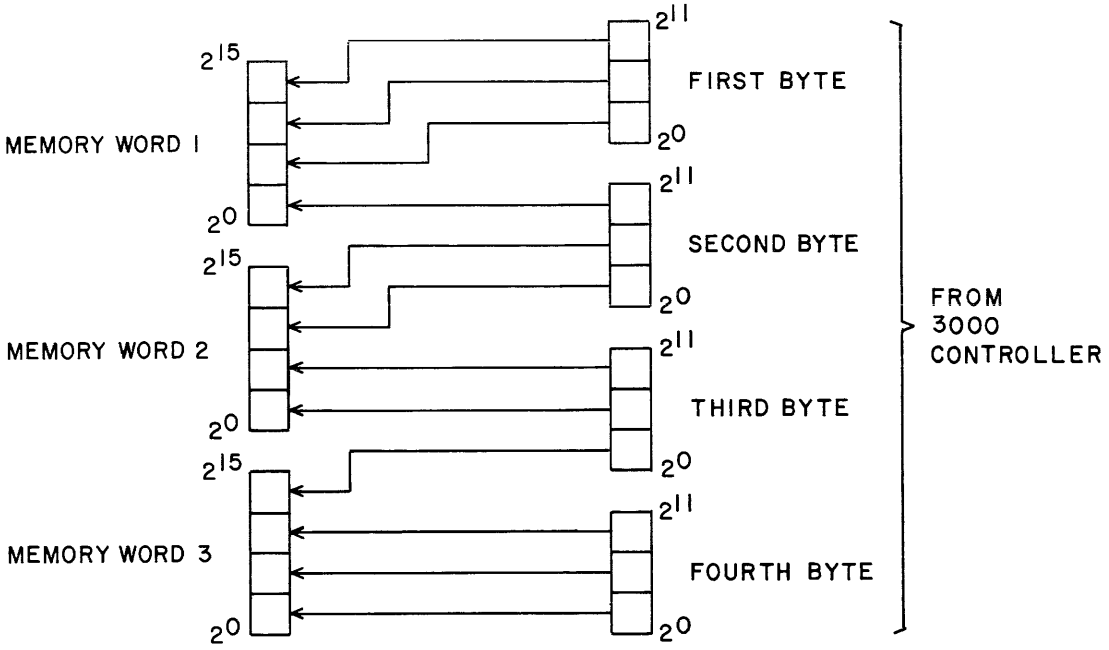


Figure 2-9. Memory Word Assembly, Read Binary

WRITE BINARY

During this operation, 16-bit SBU words are disassembled into 4-bit segments. The segments are then packed into 12-bit bytes that are transmitted to the 3000 controller. Figure 2-10 shows the disassembly process.

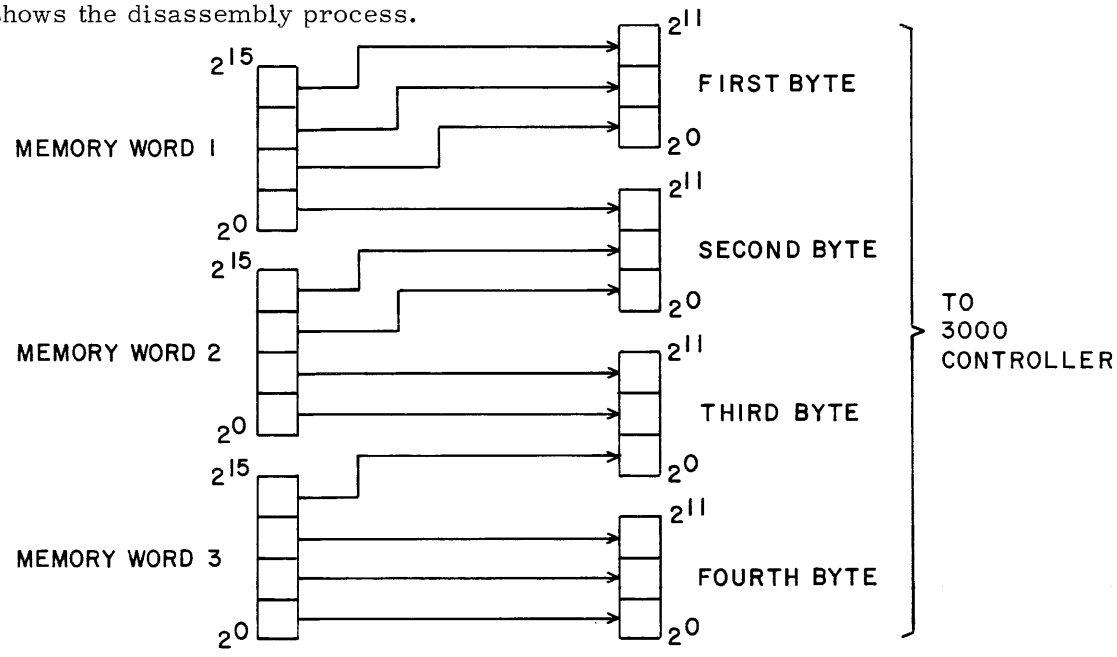


Figure 2-10. Memory Word Disassembly, Write Binary

READ CODED

In this operation, each 12-bit input byte from the 3000 controller is treated as two coded 6-bit characters. The 3000 interface converts each 6-bit character code to a corresponding 8-bit character code and then packs two successive 8-bit characters into an SBU memory word. A 64-word table in SBU memory contains 8-bit codes corresponding to each 64 possible 6-bit code combinations. Figure 2-11 illustrates the conversion process.

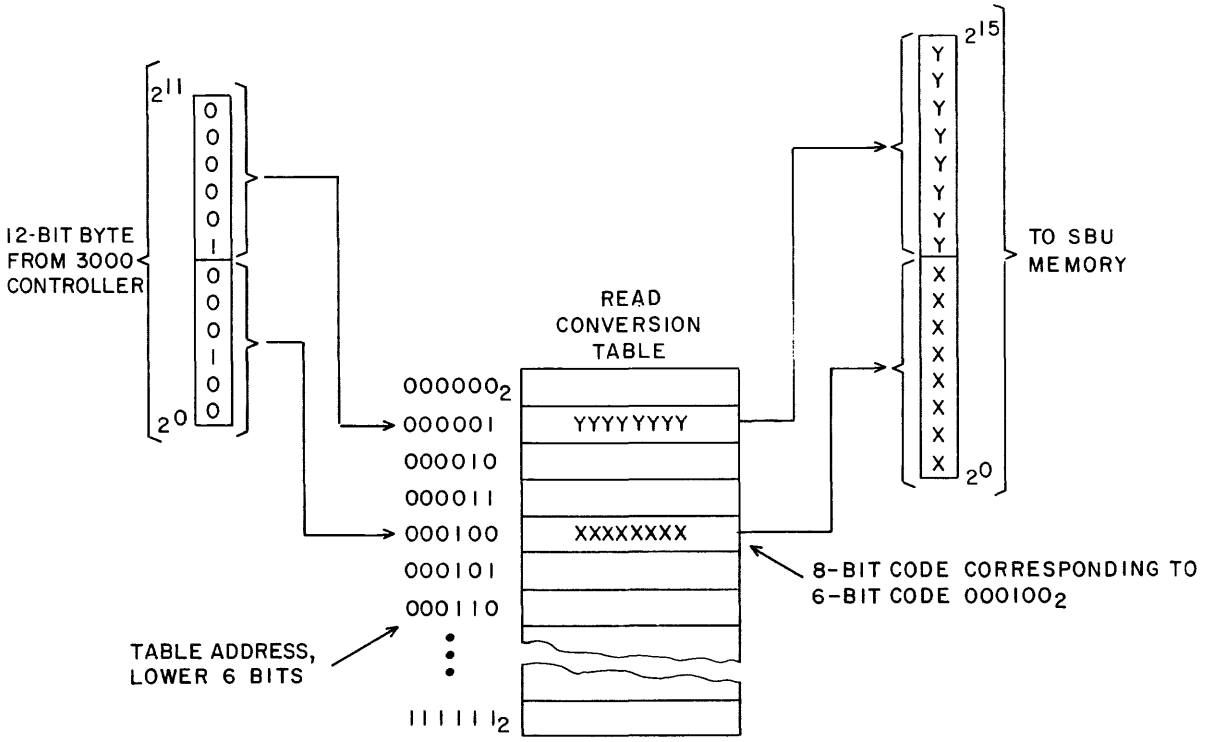


Figure 2-11. Read Code Conversion

The 6-bit codes correspond to the lower six bits of the table address. Thus, each of the 64 6-bit code combinations selects a unique table address. An 8-bit code is stored in each cell of the table.

The upper seven bits of the table address designate a 256-word region in memory that the table occupies. The read conversion table must be loaded into the lower 64 words of this area. Before the start of a coded read operation, the SCU must transmit this 7-bit table address to the 3000 interface.

WRITE CODED

In this operation, 16-bit memory words are treated as two 8-bit coded characters. The 3000 interface reads a block of words from memory and converts each 8-bit character code to a corresponding 6-bit code. Each pair of 6-bit characters forms a 12-bit byte that the interface transmits to the connected peripheral controller. The 3000 interface refers to a 256-word code conversion table in SBU memory to obtain the 6-bit code corresponding to each 8-bit code. Figure 2-12 shows the conversion process.

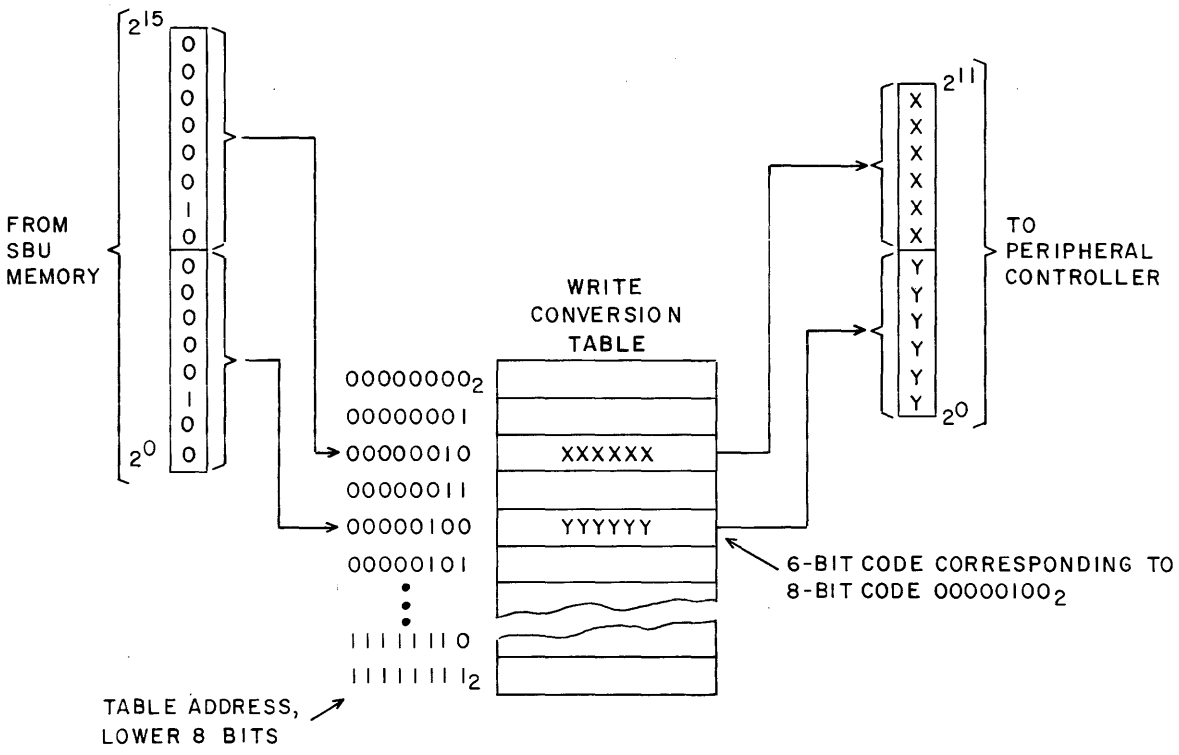


Figure 2-12. Write Code Conversion

The 8-bit character codes form the lower eight bits of the table address. Each of the 256 8-bit code combinations selects a unique table address that contains a corresponding 6-bit code.

The upper seven bits of the table address designate the area in memory occupied by the 256-word table. Before the start of a write coded operation, the SCU must load the conversion table with the appropriate 6-bit code set. The SCU also must send the upper seven bits of the table address to the 3000 interface.

READ BACKWARD OPERATIONS

The two backward read operations, read backward (binary) and read backward (coded), are used when reading in reverse from magnetic tape. The operations are similar to the forward read operations with the following differences.

The data transfer operation progresses backward through SBU memory, from the highest address to the lowest address in the buffer area.

The order of memory word assembly is reversed.

Thus, data read in reverse from tape under a backward read operation appears in memory in the same order as data read from tape in the forward direction under a forward read operation.

In the backward read operations, the starting SBU address is the upper boundary, and the terminating address is the lower boundary.

CODE CONVERSION TABLES

Before a read coded operation or write coded operation begins, a code conversion table must be loaded into SBU memory. Two tables are required, one for read and another for write. The tables can be loaded from the SCU through the SBU coupler.

READ CONVERSION TABLE

The read table is a 64-word table used for 6-bit to 8-bit conversion during read coded operations. The format is shown in Figure 2-13.

Each cell must be loaded with an 8-bit code corresponding to one of the 64 possible 6-bit code combinations.

Table requirements are:

The table must be loaded into the lower 64 words of any 256-word quarter page. This means the lowest address must be $XX00_{16}$.

The appropriate 8-bit code must be duplicated in both halves of each memory word.

8-bit codes must be loaded in the following sequence.

Address $XX3F_{16}$	Enter 8-bit code corresponding to 6-bit code 111111_2
Address XX3E	Enter 8-bit code corresponding to 6-bit code 111110
.	.
.	.
.	.
.	.
.	.
Address XX01	Enter 8-bit code corresponding to 6-bit code 000001
Address $XX00_{16}$	Enter 8-bit code corresponding to 6-bit code 000000_2

The table boundary must not fall on a 2K page boundary; that is, the starting table address plus the highest 8-bit code (FF_{16}) must not equal $7FF_{16}$.

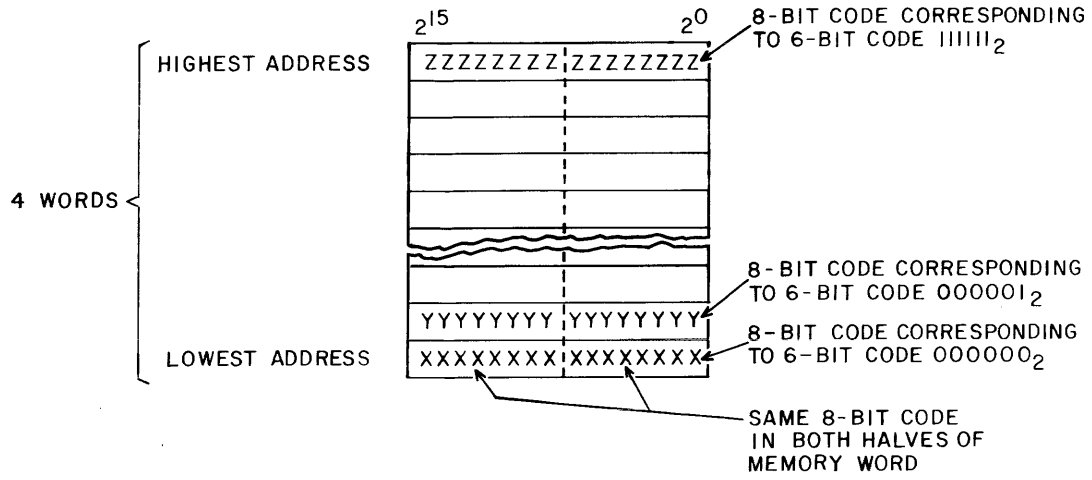


Figure 2-13. Read Conversion Table

WRITE CONVERSION TABLE

The write table is a 256-word table used for 8-bit to 6-bit code conversion during write coded operations.

The format is shown in Figure 2-14.

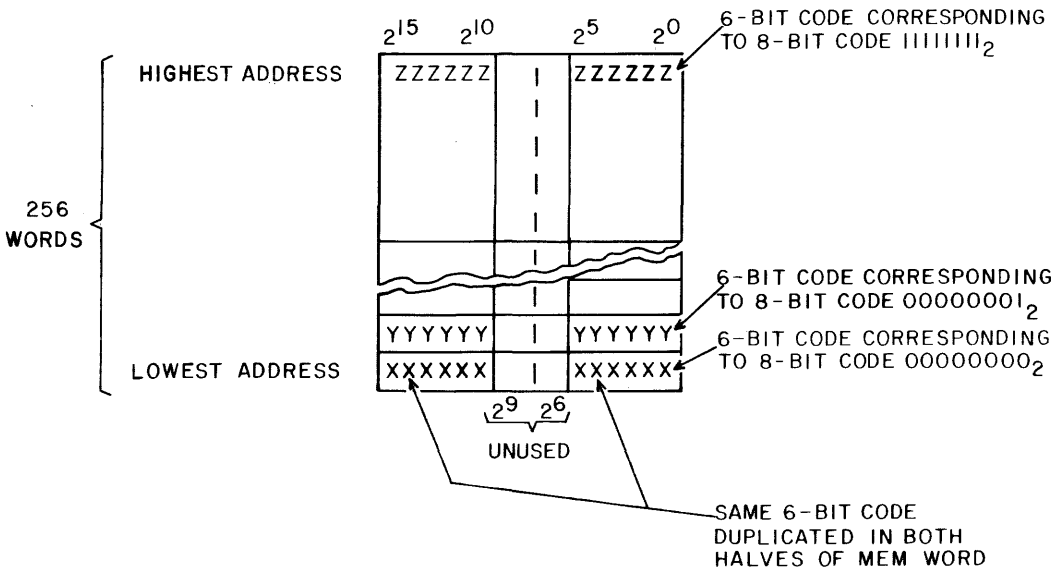


Figure 2-14. Write Conversion Table

Each memory word must be loaded with a 6-bit code corresponding to one of the 256 possible 8-bit code combinations.

Table requirements are:

The table may be loaded into any 256-word quarter page. Thus, the lowest address must be $XX00_{16}$.

The appropriate 6-bit code must be duplicated in both halves of each memory word and left- or right-justified as shown in Figure 2-14.

6-bit codes must be loaded in the following sequence.

Address $XXFF_{16}$	Enter 6-bit code corresponding to 8-bit code 11111111_2
Address XXFE	Enter 6-bit code corresponding to 8-bit code 11111110

Address XX01	Enter 6-bit code corresponding to 8-bit code 00000001
Address XX00	Enter 6-bit code corresponding to 8-bit code 00000000_2

ADDRESS FORMATS

Three types of SBU memory addresses are used in conjunction with various 3000 interface operations.

SBU starting address

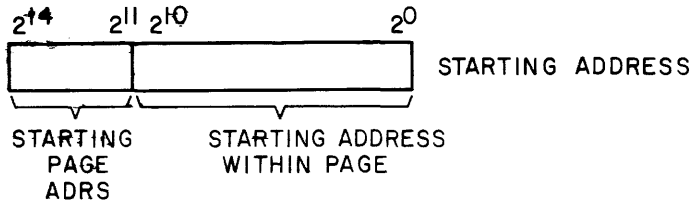
SBU terminating address

Table address

Not all addresses are required for every data transfer operation. The SCU must issue the appropriate addresses to the 3000 interface via the normal channels before initiating a data transfer operation.

STARTING ADDRESS

This 15-bit address specifies the starting point in SBU memory for a data transfer operation. It is required for all data transfer operations. The starting address is held in the 3000 interface, and the lower 11 bits are incremented each time a data word is stored or extracted from memory.

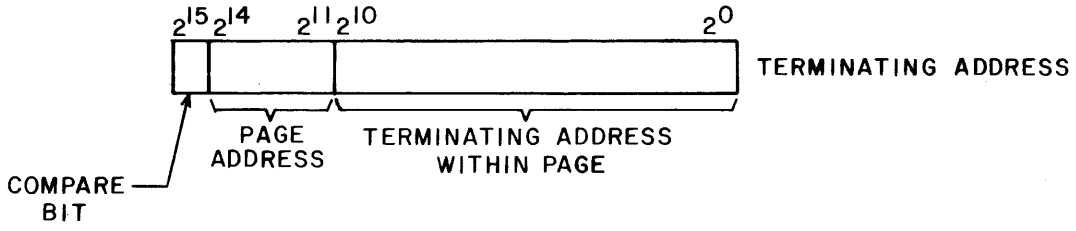


TERMINATING ADDRESS

This address serves two purposes.

It can be used to specify the next page address for multipage read or write operations.

It can be used to specify the upper limit of a write operation. The terminating address cannot be used to end a read operation.

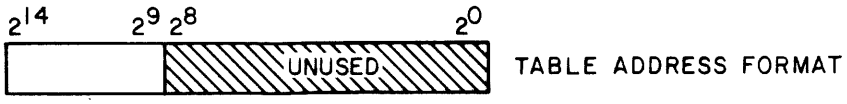


If the page address in the terminating address word is different from the starting page address, it specifies the next page. When the end of the first page is reached, the operation automatically continues at the beginning of the next page.

The compare bit must be 1 for a write operation to terminate. If this bit is set, a write operation terminates when the terminating page address matches the current page address and the terminating address within the page is reached.

TABLE ADDRESS

This 6-bit address specifies the location in SBU memory of a 256-word quarter page that contains a code conversion table. A table address is required only for read coded, write coded, and read backward (coded) operations.



MULTIPAGE DATA TRANSFER OPERATIONS

If a data transfer operation spans more than one page of memory (2048 16-bit words), the upper four bits of the terminating address are used to specify the next page. Before a multipage operation is initiated, the address of the second page must be specified in the upper four bits of the terminating address. When the data transfer operation reaches the end of the first page, it automatically jumps to the beginning of the next page as specified in the previously

loaded terminating address. If still another page of memory is required, the SCU must issue another page address after the operation has progressed into the second page. The SCU has the transfer time of an entire page to issue the next page address. This process allows a single data transfer operation to progress through several noncontiguous pages.

TERMINATING CONDITIONS

Several conditions cause the 3000 interface to terminate a data transfer operation.

SBU terminating address/ current address compare	This compare can be used only to terminate a write operation.
End of record	The 3000 interface terminates a read operation when the peripheral device signals that it has reached the end of a record. This is the normal method of terminating a read operation.
Parity error	Three types of parity error terminate a data transfer operation early.

1. An SBU memory parity error can occur only when data is being extracted from memory during a write operation.
2. A read parity error is a transmission parity error in a data byte sent from a 3000 peripheral controller to the 3000 interface parity error in a data byte sent from the 3000 interface to a 3000 peripheral controller.
3. Write parity error.

The terminate-on-parity error feature must be selected by a normal channel bit. If not selected, a parity error does not terminate a data transfer operation.

SCU CONTROL OF THE 3000 INTERFACE

The SCU exercises control over the 3000 interface and subordinate peripheral controllers through control information issued on normal output channels 5 and 6. Status information, available on normal input channels 5 and 6, allows the SCU to monitor operation of the 3000 interface and peripheral controllers. Figure 2-15 shows these normal channel bits.

Several types of control information must be sent to the 3000 interface to prepare for and initiate a read or write operation.

Starting address	Specifies the beginning of the source of destination area in SBU memory.
Table address	Specifies location of a code conversion table in SBU memory.
3000 interface function bits	Initiates read or write operations and specifies operating conditions within the 3000 interface.
Peripheral controller connect code	Selects and activates one of the eight possible controllers subordinate to the 3000 interface.
Peripheral controller function codes	Prepares the controller for an I/O operation.
Master clear 3000 interface	Stops current I/O operation and clears all function selections within the 3000 interface.
3000 interface connect code	Connects 3000 interface to SCU normal input and output channels 5 and 6. A unique 4-bit code distinguishes the 3000 interface from all other I/O interfaces present in the SBU.
Terminating address	Specifies the upper limit of the destination area in SBU memory used for write operations only.

Three 16-bit status words are available on input channels 5 and 6. Two of these status words contain information about operating conditions within the 3000 interface. The third word contains status information from the connected peripheral controller.

The 3000 interface must be connected to the SCU in order to receive control information or return status information. The interface remains connected as long as the SCU holds the unique connect code assigned to the 3000 interface on output channel 6. The 3000 interface need not remain connected to the SCU normal channels to maintain a connected condition between the 3000 interface and a 3000 equipment and/or unit.

NORMAL CHANNEL ASSIGNMENTS

All of the normal channel signals used to control the 3000 interface are defined in the following section. The normal channel bit assignments given apply to the storage (media) station. 3000 interface bit assignments for other stations may be different. Appendix A of this manual contains a full set of normal channel assignments for all STAR stations.

NORMAL OUTPUT CHANNEL 5

Normal output channel 5 is used to transmit the following five classes of control information to the 3000 interface.

Terminating address

Starting address

Table address

3000 controller connect/function code

3000 internal function bits

Control information on output channel 5 must be accompanied by one of five strobe signals on output channel 6. The purpose of the strobe signals is to identify the class of information being issued and to transfer this information from channel 5 into the interface. Figure 2-15 shows the relationship between the strobe signals and the five classes of control information.

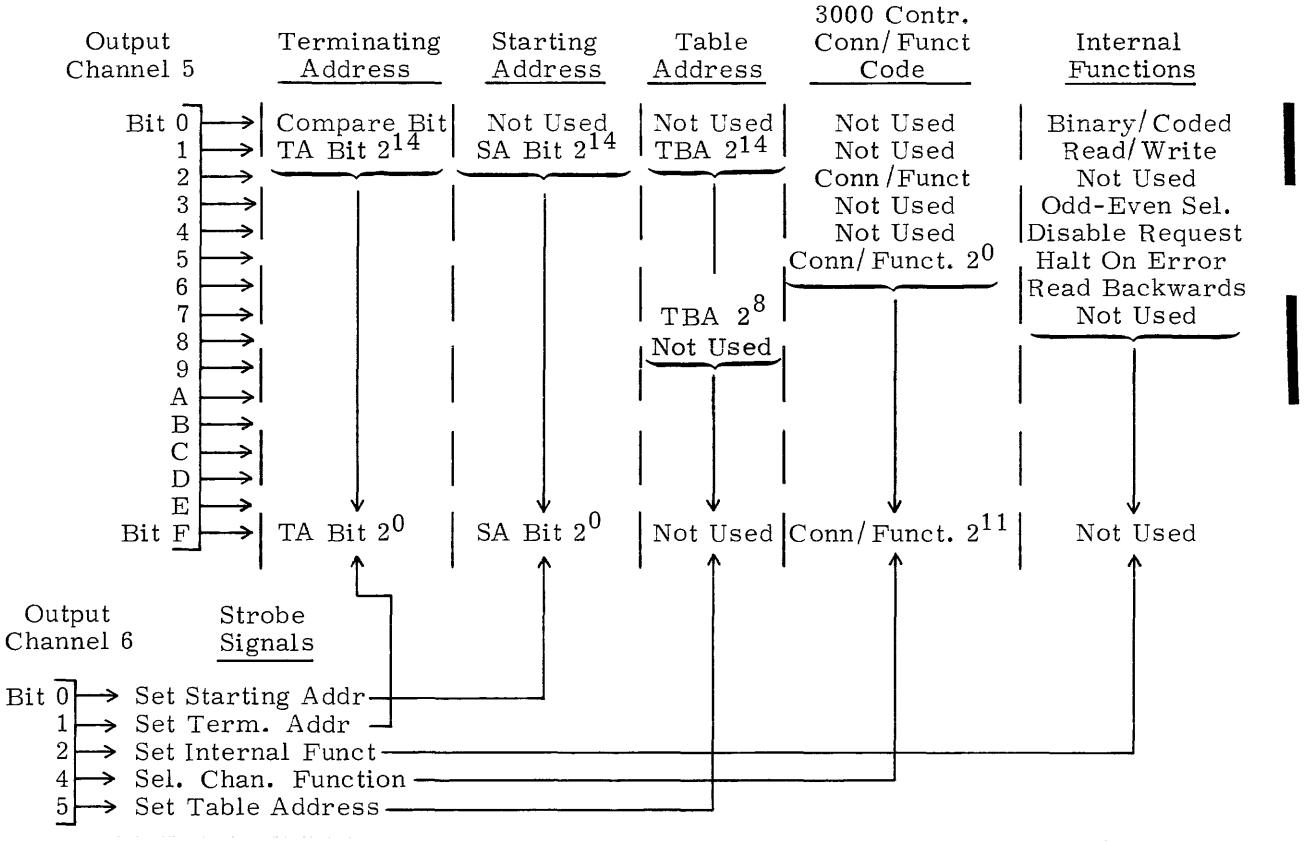


Figure 2-15. Output Channel 5 Control Information, 3000 Interface

TERMINATING ADDRESS

The compare bit (bit 0) is a control bit that is transmitted along with the 15-bit terminating address. If set, this bit causes the 3000 interface to terminate a write operation when the terminating address is reached. It has no effect on a read operation.

The terminating address (bits 1 through F) has two parts.

Bits F through 5 (2^0 through 2^{10}); the lower 11 bits specify the terminating address within a 2K page.

Bits 4 through 1 (2^{11} through 2^{14}); next page address. For read or write operations longer than one page (2048 16-bit words), these bits specify the address of the next memory page.

The SCU must issue this address (2^0 through 2^{14}) in ones complement form.

STARTING ADDRESS

This 15-bit address specifies the starting SBU memory address for a read or write operation. The address must be issued in ones complement form.

TABLE ADDRESS

These seven address bits (channel bits 7 through 1) form the upper (most significant) part of a 15-bit table address. They specify the location of a 256-word code conversion table in SBU memory. When the code conversion mode is in effect, this table is used to convert from 6-bit to 8-bit format and vice-versa. This address must be issued in ones complement form.

3000 CONTROLLER CONNECT/FUNCTION CODE

The connect function bit (bit 2), which accompanies the 12-bit connect/function code, determines whether the 3000 interface will transmit the code as a connect code or a function code (1 equals connect, 0 equals function). The channel function bit (NOC-6, bit 4) gates the code to the external equipment.

The 3000 channel connect/function code (bits 4 through F) is a 12-bit code forwarded to the 3000 peripheral controllers by the 3000 interface.

INTERNAL FUNCTION BITS

These six bits affect the 3000 interface. They specify the type of operation (read or write) that the interface is to perform and establish various operating conditions that affect the way the operation is performed. These bits are activated when the internal function select bit (NOC-6, bit 2) is set.

Bit 0, binary/coded

1. When 0, this bit establishes binary mode. In binary mode, no data conversion takes place during read or write operations.
2. When 1, this bit establishes code conversion mode. When this mode is in effect, data is converted between 8-bit and 6-bit format. During a read (input) operation, each 6-bit input character is converted to a corresponding 8-bit character that is then stored in SBU memory. During a write (output) operation, the process is reversed; 8-bit characters from memory are converted to 6-bit output characters. A code conversion table, in SBU memory, specifies the equivalent 6-bit and 8-bit codes.

Bit 1, read/write

1. Clearing this bit initiates a read (input) operation.
2. Setting this bit initiates a write (output) operation.

Bit 2, unused

Bit 3, odd/even select

Setting this bit causes the 3000 interface to reference every other bank in SBU memory. Clearing this bit causes the 3000 interface to reference all banks in memory consecutively.

Bit 4, disable request

When set, this bit conditions the 3000 interface to inhibit references to SBU memory. A read or write operation increments the memory address normally but the interface neither reads nor writes SBU memory.

Bit 5, halt on error

When set, this bit conditions the 3000 interface to enable the stop-on-parity error feature. The type of parity error affecting the 3000 interface may exist in any one of three categories; SBU memory, a transmission error, or in a magnetic tape controller. If this bit is not set, a parity error does not terminate a data transfer operation and the operation continues until halted by address compare or end of record.

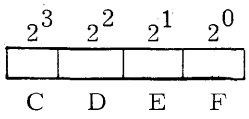
Bit 6, read backwards

When set, this bit reverses storage in memory during a read binary or read coded operation; for read binary, it also reverses the order of assembly. This feature is intended for use when data is being read backwards from tape.

TABLE 2-12. NOC-6 BIT DESCRIPTIONS FOR 3000 INTERFACE

Channel	Bit	Name	Function
NOC-6	0	SA SBU	This bit identifies the information on output channel 5 as an SBU starting address. Setting and then immediately clearing this bit transfers the starting address into the 3000 interface.
NOC-6	1	TA SBU	This bit identifies the information on output channel 5 as the SBU terminating address. Setting and then immediately clearing this bit transfers the terminating address into the 3000 interface.
NOC-6	2	Hardware function	This bit identifies the information on output channel 5 as internal function bits. Setting and then immediately clearing this bit transfers the function bits to the 3000 interface. The conditions specified by the function bits remain in effect until new function bits are issued or the interface is cleared.
NOC-6	3	Program clear	This bit clears the 3000 interface. Setting this bit causes the clearing action to take place.
NOC-6	4	Select channel function	This bit identifies the information on output channel 5 as a 3000 channel connect or function code. Setting and then immediately clearing this bit causes the 3000 interface to transmit the code to the subordinate 3000 peripheral controllers.
NOC-6	5	Table address-SBU	This bit identifies the information on output channel 5 as the table address. Setting and then immediately clearing this bit transfers the table address (output channel 5, bits 1 through 8) to the 3000 interface.
NOC-6	6	Current address-SBU	Setting this bit selects the current SBU memory address for input as a status word on input channel 5. Once selected, current address status remains available until this bit is

TABLE 2-12. NOC-6 BIT DESCRIPTIONS FOR 3000 INTERFACE (Cont'd)

Channel	Bit	Name	Function
			cleared. When current address status is not selected, the default status word is available on input channel 5.
NOC-6	7-B	Unused	
NOC-6	C-F	Interface connect	Interface connect codes for the 3000 interfaces are: 3000 Channel 0: code C_{16} (1100_2) 3000 Channel 1: code D_{16} (1101_2) This code connects the 3000 interface to SCU normal input and normal output channels 5 and 6 for control purposes. The paths between the normal channels and 3000 interface remain connected only as long as the proper connect code is present on output channel 6. 

NORMAL INPUT CHANNEL 5

Two status words (refer to Tables 2-13 and 2-14) from the 3000 interface are available to the SCU on input channel 5.

When the SCU issues the 3000 interface connect code, default status is automatically selected. Current address status is selected by bit 6, normal channel 6.

TABLE 2-13. WORD FORMATS FOR NIC-5 STATUS WORDS, 3000 INTERFACE

Bit of NIC-5	Current Address Status	Default Status
Bit 0	Not used	Not used
1	Not used	Not used
2	Interrupt active†	Interrupt active†
3	SBU parity error	SBU parity error
4	Read parity error	Read parity error
5	Current address 2^{10}	Interrupt line 3
6	Current address 2^{9}	Interrupt line 2
7	Current address 2^{8}	Interrupt line 1
8	Current address 2^{7}	Interrupt line 0
9	Current address 2^{6}	Assembly count 2^{3}
A	Current address 2^{5}	Assembly count 2^{2}
B	Current address 2^{4}	Assembly count 2^{1}
C	Current address 2^{3}	Assembly count 2^{0}
D	Current address 2^{2}	$\overline{\text{Phase counter}}$ 2^{2}
E	Current address 2^{1}	$\overline{\text{Phase counter}}$ 2^{1}
F	Current address 2^{0}	$\overline{\text{Phase counter}}$ 2^{0}

† Note that bits 2, 3, and 4 of the two status words are identical.

TABLE 2-14. NIC-5 STATUS WORD BIT DESCRIPTIONS FOR 3000 INTERFACE

Channel/Bit	Name	Function
	Current Address Status	
NIC-5/bit 0		Unused
NIC-5/bit 1		Unused
NIC-5/bit 2	Interrupt active	Indicates that one or more of the four interrupt lines from the peripheral controllers is active.
NIC-5/bit 3	SBU parity error	Indicates that a parity error has occurred in a word read from SBU memory.
NIC-5/bit 4	Read parity error	Indicates that the 3000 interface has sensed a parity error in a data byte from a peripheral controller.
NIC-5/bits 5-F_{16}	Current address	These 11 status bits contain the current SBU memory address for the read or write operation in progress. Only the lower 11 bits of the 15-bit address (the address within a 2K page) are available.
	Default Status	
NIC-5/bit 0		Unused
NIC-5/bit 1		Unused
NIC-5/bit 2	Interrupt active	Same as current address status
NIC-5/bit 3	SBU parity error	
NIC-5/bit 4	Read parity error	
NIC-5/bit 5	Interrupt line 3	Interrupt line from controller number 3. Indicates state of the interrupt line from the peripheral controller assigned connect code 3. Logical 1 equals interrupt present.
NIC-5/bit 6	Interrupt line 2	Interrupt line from controller number 2.
NIC-5/bit 7	Interrupt line 1	Interrupt line from controller number 1.
NIC-5/bit 8	Interrupt line 0	Interrupt line from controller number 0.

TABLE 2-14. NIC-5 STATUS WORD BIT DESCRIPTIONS FOR 3000 INTERFACE (Cont'd)

Channel/Bit	Name	Function
NIC-5/bits 9-C_{16}	Assembly count	These four bits indicate the zero fill in the assembly or disassembly of a 16-bit memory word. 1. During write and forward read operations, the bits have the following meanings. • Bit C=1; upper four bits (2^{15} through 2^{12}) assembled or disassembled data (remainder equals 0). • Bit B=1; upper two 4-bit segments assembled or disassembled data (remainder equals 0). • Bit A =1; upper three 4-bit segments assembled or disassembled data (remainder equals 0). • Bit 9=1; assembly or disassembly of last memory word completely filled. 2. During read backward operations, the order of assembly is reversed. • Bit C=1; indicates the lowest 4-bit segment has been assembled or disassembled. • Bits B, A, and 9 indicate completion of the remaining three assembly or disassembly steps.
NIC-5/bits D, E, F_{16}	$\overline{\text{Phase counter}}$	This counter indicates the current not bank address of SBU memory. When an SBU memory parity error is detected, the memory bank with the parity error is determined by complementing the phase counter status and counting back four memory banks.

NORMAL INPUT CHANNEL 6

A 12-bit status word and four control signals from the connected 3000 controller are available on input channel 6 (refer to Table 2-15).

TABLE 2-15. NIC-6 STATUS WORD BIT DESCRIPTIONS FOR 3000 INTERFACE

Channel/Bit	Name	Function
NIC-6/bit 0	Write parity error	This control signal indicates that the 3000 controller has detected a parity error in a data byte or function code sent from the 3000 interface.
NIC-6/bit 1	End of record	This control signal indicates that the 3000 controller has completed sending a record of data.
NIC-6/bit 2	Reject	This control signal indicates the 3000 controller has rejected a function code sent from the 3000 interface, or the 3000 controller failed to respond to a function code within 100 microseconds.
NIC-6/bit 3	Reply	This control signal indicates that the 3000 controller has accepted one of the following units of information sent from the 3000 interface. Connect code Function code Output data byte
NIC-6/bit 4 ↓ ↓ NIC-6/bit F	Status bit 11 ↓ Status bit 0	12-bit status word from 3000 controller. Each bit indicates the presence of some operating condition such as ready or busy. The meaning of the bits is different for each type of controller. Refer to appendix A and B for magnetic tape unit status bits.

SCANNER STATUS BITS

Several status bits for the 3000 interface appear in the SBU scanner status word. In the media station, these bits are available on normal input channel 6 (refer to Table 2-16). To access the scanner status word, the SCU must issue the scanner connect code (6_{16} for the media station).

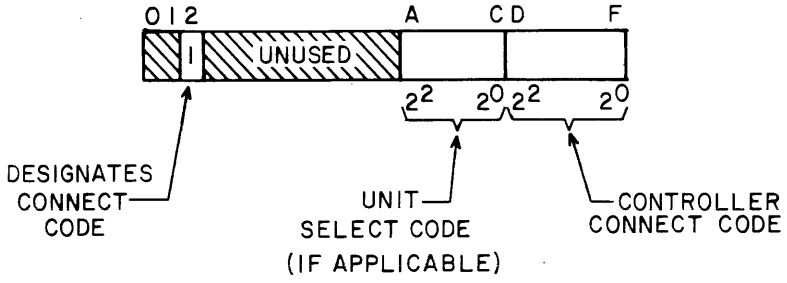
TABLE 2-16. 3000 INTERFACE SCANNER STATUS BITS

Channel/Bit	Name	Function
NIC-6/Bit 0 NIC-6/Bit 1	Response, controller 1	Indicates that a 3000 controller has returned one of the following response signals to the 3000 interface. End of record Interrupt The specific condition causing response status can be determined by checking 3000 interface status on NIC-5 and NIC-6.
NIC-6/Bit 0 NIC-6/Bit 1	Terminating address empty, controller 1 Terminating address empty, controller 2	Indicates that a page boundary has been reached. In multipage operations, it means that the operation has jumped to the next page. If yet another page is required, a new next-page address can now be loaded. The status bit goes to 0 when a new page address is loaded.
NIC-6/Bit 7 NIC-6/Bit 9	Address compare, controller 1 Address compare, controller 2	Indicates that a write operation has reached the terminating address and has terminated. Not significant for a read operation.

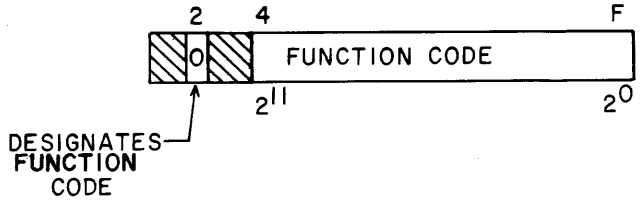
PROGRAMMING SEQUENCES

The following paragraphs define the basic programming steps required to initiate and control data transfer operations through the 3000 interface.

1. Connect 3000 interface
   Place connect code for 3000 interface in NOC-6, bits B through F. (Connect code must be held in NOC-6 as long as SCU is issuing commands to the 3000 interface.)
2. Connect 3000 tape controller and subordinate unit
   a. Place controller and unit connect code in NOC-5. Also, set NOC-5, bit 2 to specify a connect code.



   b. Set and then clear the channel function bit, NOC-6, bit 4. (This transmits the connect code through the 3000 interface to the controller.)
   c. Reply status (NIC-6, bit 3) may be checked to determine if connect was completed; logical 1 equals a successful connect. Refer to step 7 for status input procedure.
3. Issue function code(s) to 3000 controller
   a. Place controller function code in NOC-5. Also, clear NOC-5, bit 2 to specify a function code.



   b. Set and then clear channel function bit NOC-6, bit 4. (This transmits the function code through the interface to the 3000 controller.)

c. The following status bits can be checked to determine if the controller accepted the function code.

NIC-6, bit 0	1 indicates parity error in function code.
NIC-6, bit 2	Reject-1 indicates controller did not accept function code.
NIC-6, bit 3	Reply-1 indicates controller accepted function code.

(Refer to step 7 for status input procedure.)

d. Repeat steps a, b, and c to issue additional function codes if necessary.

e. Controller status can be read in on NIC-6 to determine if controller is ready. Refer to step 7.

4. Initiate binary write operation – block < 1 page (2048 words).

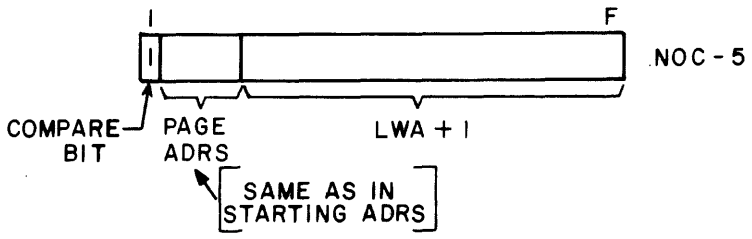
a. Issue connect and function codes according to steps 1, 2, and 3.

b. Load starting address

1) Place starting address in NOC-5, bits 1 through E_{16}.
2) Set and then clear starting address bit NOC-5, bit 0. (This loads the address into the 3000 interface.)

c. Load terminating address

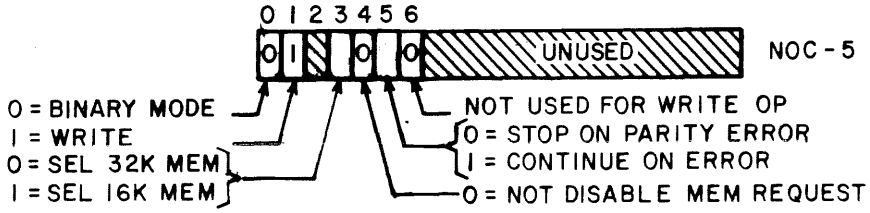
1) Place terminating address in NOC-5.



2) Set and then clear terminating address bit (NOC-6, bit 1). (This loads the address into the 3000 interface.)

d. Issue 3000 internal function bits.

1) Place function bits in NOC-5.



2) Set and then clear internal function bit, NOC-6, bit 2. (This causes data transfer to begin if 3000 peripheral controller is ready to accept data.)

3) The SCU may now disconnect from the 3000 interface.

e. Wait for compare status, scanner status bit, NIC-6, bit 7 or 9. (This bit indicates that the terminating address has been reached.)

f. After termination, status can be read in on NIC-5 and NIC-6 and checked to determine if the operation terminated normally. Refer to step 7.

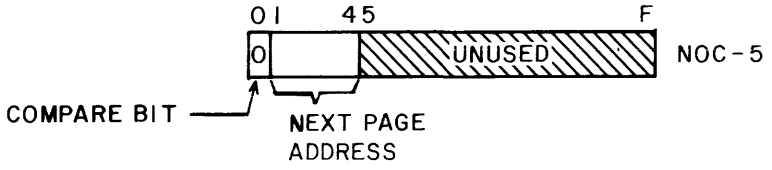
5. Initiate binary write operation - block > 1 page

a. Issue connect and function codes according to steps 1, 2, and 3.

b. Load starting address according to step 4b.

c. Load next page address.

1) Place next page address in NOC-5. Also clear the compare bit NOC-5, bit 0.



2) Set and then immediately clear terminating address bit (NOC-6, bit 1). (This transfers the address to the 3000 interface.)

d. Issue 3000 internal function bits.

1) Same as 4d. This starts data transfer. (The operation automatically continues in next page when first page is emptied.)

2) SCU may now disconnect from the 3000 interface.

e. Wait for terminating address empty status to occur and then load new next-page address.
(This is a scanner status bit, NIC-6, bit 6 or 8. It sets when the write operation reaches the end of a page and jumps to the previously loaded next-page address.)

1) If the new page is not the last page:

a) Connect 3000 interface according to step 1.

b) Load new page address according to step 5c.

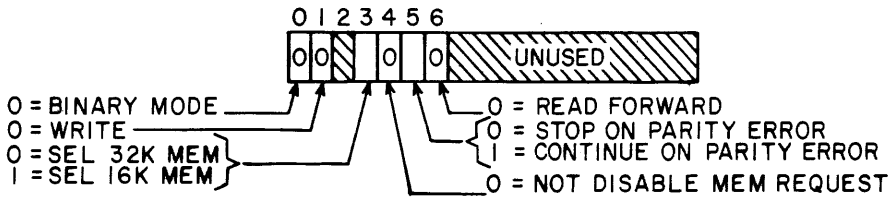
c) SCU may disconnect from the 3000 interface.

d) Go to back to step 5e.

2) If the new page is to be last page:

a) Connect 3000 interface according to step 1.

b) Place terminating address in NOC-5.



c) Set and then clear the terminating address bit (NOC-6, bit 1).

d) SCU may disconnect from 3000 interface.

3) Wait for compare status, scanner status bit, NIC-6, bit 7 or 9. (This bit indicates that the terminating address has been reached.)

4) After terminating, check status available on NIC-5 and NIC-6 for normal or abnormal termination. Refer to step 7.

6. Initiate binary read operation

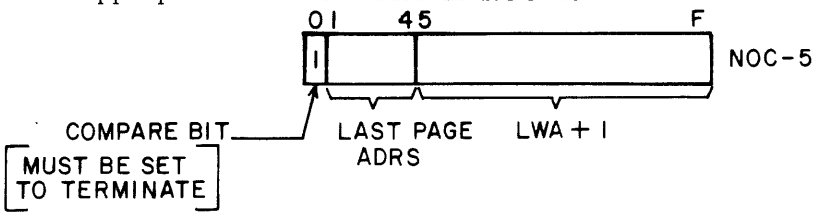
a. Issue connect and function codes according to steps 1, 2, and 3.

b. Load starting address according to steps 4b and 4c.

c. Load next page address according to step 5c (not required if block less than one page, 2048 words).

d. Issue 3000 interface function bits to select operating mode.

1) Place appropriate function bits in NOC-5.



2) Set and then clear internal function bit, NOC-6, bit 2. (This causes data transfer to begin if the 3000 peripheral device is ready to send data.)

3) The SCU may now disconnect from the 3000 interface.

e. Wait for one of the following scanner status bits.

TA empty status - NIC-6, bit 6 or 8
(Indicates the read operation has progressed to next page.)

Response status - NIC-6, bit 0 or 1
(Indicates that 3000 controller has terminated the read operation or has sent an interrupt.)

1) If TA empty status occurs:

   a) Connect 3000 interface according to step 1.

   b) Load new next-page address according to 5c.

   c) SCU may disconnect from the 3000 interface.

   d) Go back to step 6e.

2) If response status occurs:

   a) Check status available on NIC-5 and NIC-6 to determine if operation terminated normally. Refer to step 7. End of record status indicates normal termination.

3) To determine last word filled in SBU memory:

   a) Connect 3000 interface according to step 1.

   b) Set and then clear current address bit (NOC-6, bit 6). (This selects current address status on NIC-5.)

   c) Input to A register from NIC-5.
   (Bits 5 through 15 contain last word address in final page.)

4) To determine number of bytes in last word:

   a) Connect 3000 interface according to step 1.

   b) Set and then clear table address bit, NOC-6, bit 6.
   (In this case, this is a dummy command that serves only to clear the interface logic to select default status.)

   c) Input to A register from NIC-5.
   (Bits 9 through 12 contain the byte count.)

7. Status input procedure

   a. Controller status

      1) Connect interface and controller (refer to steps 1 and 2).

      2) Input to A from NIC-6.

   b. Default status

      1) Connect interface (refer to step 1).

      2) NOC-6, bit 6 must be clear.

      3) Input to A from NIC-5.

   c. Current address status

      1) Connect interface (refer to step 1).

      2) Set NOC-6, bit 6.

      3) Input to A from NIC-5.

7000 I/O INTERFACE

The 7000 I/O interface coordinates data word-size conversion and provides buffering for data transfer between SBU memory and a 7000 peripheral device such as a 7638 disk storage subsystem (Figure 2-16). The SCU initiates data transfer operations, monitors status, and terminates these processes through normal channel inputs to the 7000 I/O interface. A 7000 I/O interface contains a control channel and a data channel. The control channel supplies function codes to the system and receives status reports from the system. The data channel exclusively handles data both to and from the peripheral system.

Two 7000 I/O interfaces may operate simultaneously in the SBU completely independent of each other. The SCU differentiates one 7000 I/O interface from the other by using a different connect code for each interface.

The following paragraphs of this subsection describe only the programming characteristics of the 7000 I/O interface; they do not cover the peripheral subsystems that can be connected to the interface. The reader should be familiar with 7000 I/O channel operation and the appropriate programming conventions in order to fully understand this material.

MAIN FUNCTIONS

The principal functions of the 7000 I/O interface are:

Issues function codes to 7000 peripheral subsystems

Governs exchange of data between peripheral subsystems and SBU memory

Performs word assembly/disassembly operations. During input operations, the interface breaks 12-bit bytes from the subsystem into 4-bit segments and packs four of these segments into each 16-bit SBU memory word. During output operations, the interface reverses the process.

Provides a path from the 7000 peripheral subsystem to the SCU for a 12-bit controller status word.

Provides several status bits from the 7000 peripheral subsystem which form part of the SBU scanner status word.

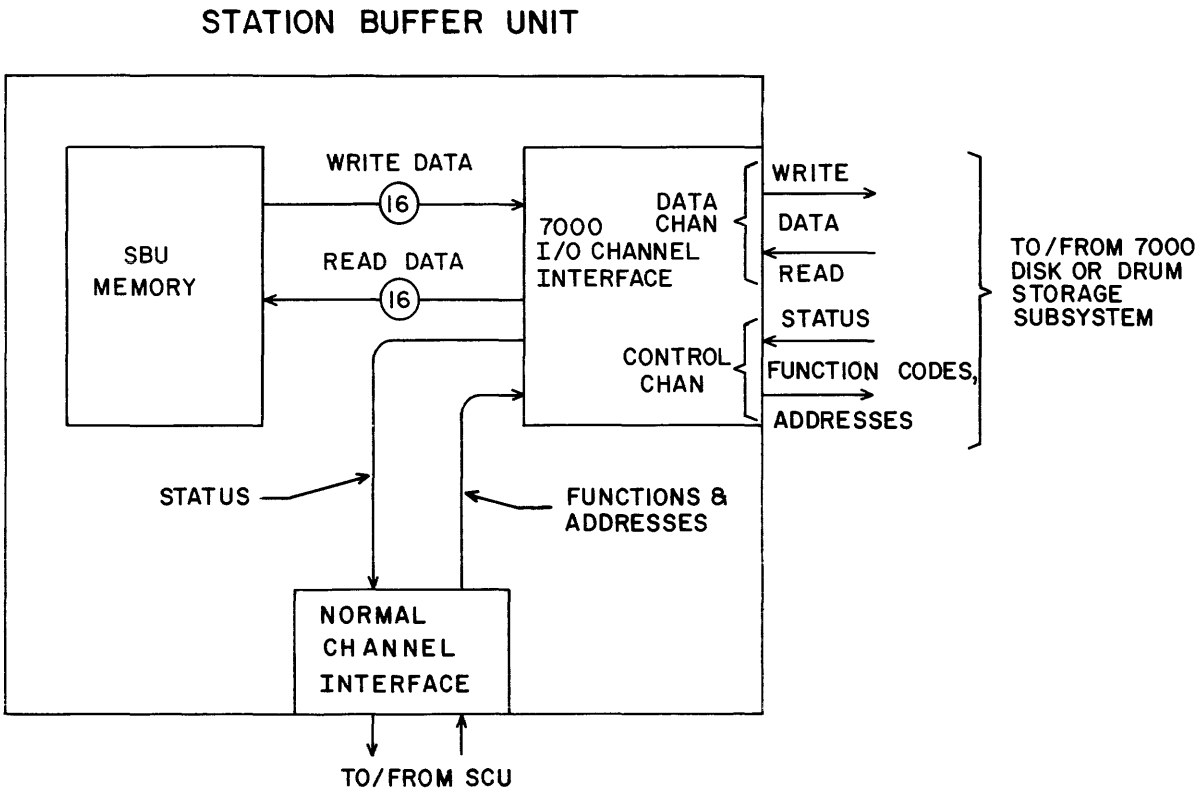


Figure 2-16. 7000 Interface System Relationship

The SCU controls all of these functions by issuing control information on normal output channels 5, 6, and 7. Fulltime attention of the SCU is not required once an operation has been initialized, since the 7000 interface transfers the specified number of data words without continued support from the SCU.

DATA TRANSFERS

All data transfer operations involve the exchange of blocks of data words between SBU memory and the 7000 peripheral subsystem. Before the SCU initiates a data transfer operation, it sends a starting address to the SBU to specify the beginning location in the buffer area of SBU memory where that data word is stored. Likewise, terminating addresses determine the stopping point in SBU memory for that data word. There are two types of data transfer operations, read and write.

The SCU sends function codes to the peripheral subsystem instructing it to perform specified operations. Bits 2^{11}, 2^{10}, and 2^{9} of the function word form a 3-bit function code which determines the operation that will be performed; for example, 001 for write, 000 for read, etc. Also, bit 2^{14} of the function word must be set to a 1 in order to initiate a data transfer. Otherwise, it must remain at a 0.

READ (INPUT) OPERATIONS

The SCU issues a starting address followed by the read function word to initiate the read operation. During this operation, 12-bit words or data bytes from the 7000 peripheral subsystem are divided into 4-bit segments. These segments are then packed into 16-bit words for storage in SBU memory. Figure 2-17 illustrates this procedure.

A terminating address also previously issued by the SCU terminates a read operation. If more than one area in memory is required for the number of data words coming from the 7000 peripheral subsystem, the SCU program must specify another starting and terminating address for the additional data words. A read operation may also be terminated by any abnormal condition, a not ready, a parity error at the end of the data transfer, etc.

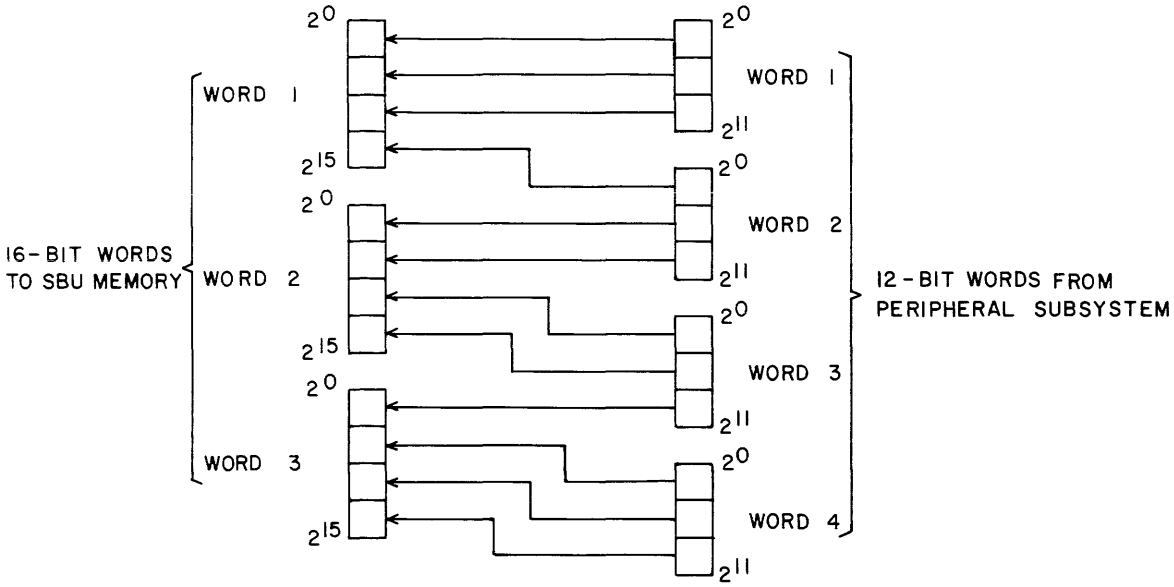


Figure 2-17. Input Operation, 12- to 16-Bit Assembly

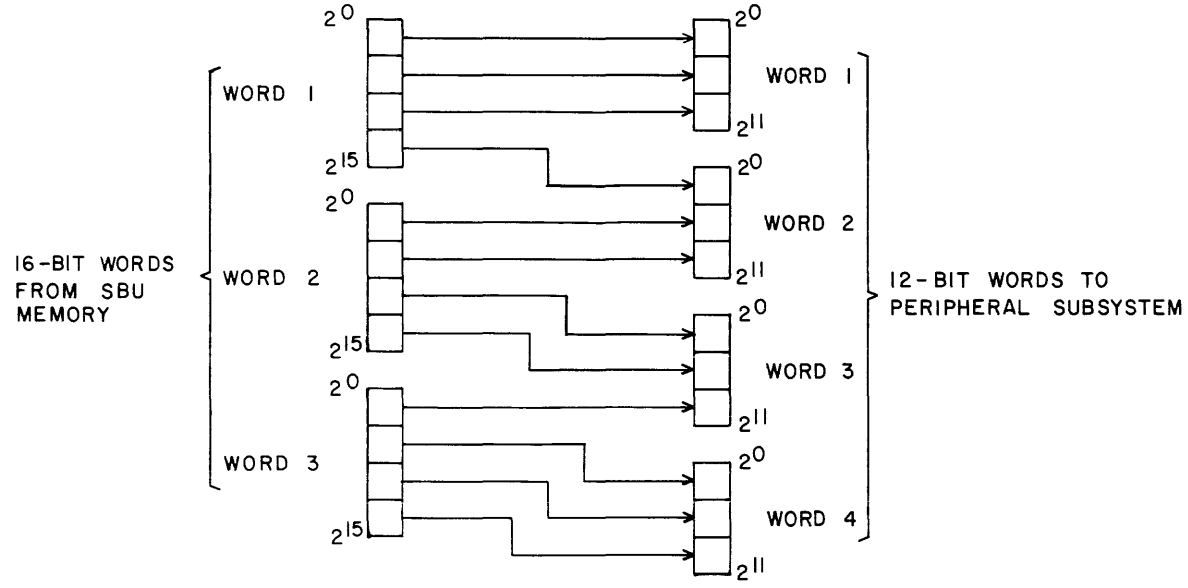


Figure 2-18. Output Operation, 16- to 12-Bit Disassembly

WRITE (OUTPUT) OPERATIONS

A write operation begins simlar to a read, except the SCU issues the write function word to initiate the operation. During this operation, 16-bit words are extracted from SBU memory, disassembled into 4-bit segments, and packed into 12-bit bytes (Figure 2-18). These bytes are then transmitted to the 7000 peripheral subsystem.

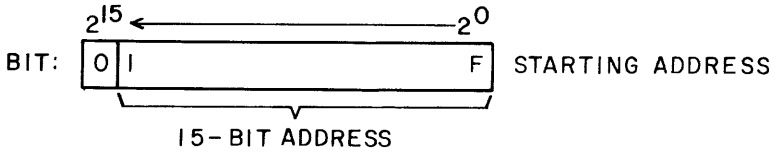
If more data words than those specified by the first starting/terminating addresses are to be transferred, registers are provided which permit the stacking of four sets of starting/terminating addresses. Normally, a write operation stops when the terminating address is reached which has bit 2^{15} set, indicating the end of a data sequence. However, as is the case for a read operation, a write may also be terminated by any abnormal condition, master clear, etc.

ADDRESS FORMATS

Two types of address word formats are sent by the SCU to the 7000 interface; they specify starting and terminating address locations in SBU memory. Before the SCU initiates a read or a write operation, it must send both addresses to the 7000 interface via a normal output channel.

STARTING ADDRESS

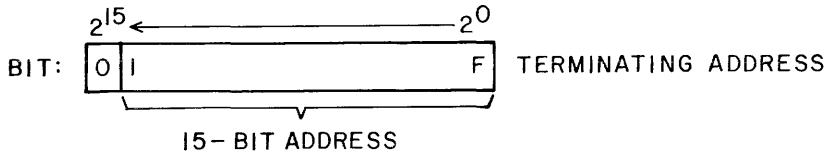
This 16-bit address specifies the starting point in SBU memory for both read and write operations. The starting address is held in core control and incremented each time a data word is stored or extracted from memory. The following is the format of the starting address.



Bit 0 (2^{15}) of the starting address word carries special meaning. In this case, bit 0 of the first starting address is set to indicate that the current address word is the first of a data sequence. For all other starting address words in the sequence, bit 0 remains a 0. A sequence may consist of a single address. Issuing several sets of starting and terminating addresses becomes necessary when the data transfer operation is longer than data stored in one page of SBU memory or when the data is scattered in SBU memory.

TERMINATING ADDRESS

This 16-bit address specifies the stopping point in SBU memory for read and write data transfer operations. The terminating address is compared with a return address from SBU memory; if they coincide, the SCU may stop data transfer or allow the 7000 interface to transfer a new address to SBU memory. The following is the format of the terminating address.



Bit 0 of the terminating address word is set only for the last of the terminating addresses in a sequence. For all other terminating addresses in the sequence, bit 0 remains a 0.

SCU CONTROL OF 7000 INTERFACE

The SCU controls the 7000 interface and subordinate peripheral subsystems using control information issued on normal output channels 5, 6, and 7. Status information, available on normal input channels 5, 6, and 7, allows the SCU to monitor operation of the interface and its associated peripheral equipment.

Several types of control information must be sent to the 7000 interface to prepare for and initiate a read or a write operation.

Starting and terminating addresses	Specifies the length of a read or write operation and the source or destination area in SBU memory.
7000 interface function bits	Used to initiate read or write operations and to specify operating conditions within the 7000 interface.

Peripheral subsystem function codes	These codes are issued by the SCU, sent through the 7000 interface, and interpreted by the peripheral subsystem.
Master clear of 7000 interface	Stops current I/O operation and clears all function selections within the 7000 interface.
7000 interface connect code	A unique 4-bit code distinguishing the 7000 interface from all other I/O interfaces present in the SBU.

Status bits for the drum station are available via normal input channels 5 and 6. The status obtained from NIC-5 deals with operating conditions within SBU core control and within the drum unit. Status obtained from NIC-6 provides additional information on the drum unit and 7000 interface for use by the SCU scanner program.

Status bits for the disk station are available via normal input channels 5, 6, and 7. The status obtained from NIC-5 and NIC-7 provides information on operating conditions within SBU memory and within the disk unit. Similar to the drum station, status obtained from NIC-6 provides information on the disk unit and 7000 interface.

The 7000 interface must be connected to the SCU in order to receive control information or to return status information. The interface remains connected as long as the SCU holds the unique connect code to the 7000 interface on normal output channel 6.

NORMAL CHANNEL BIT ASSIGNMENTS

All of the normal channel signals used to control the 7000 interface are defined either in the following paragraphs or in Tables 2-17, 2-18, and 2-19. The explanations of bit assignments given here apply to either the drum or disk peripheral stations. If the 7000 interface is used in other stations, bit assignments may be different. Appendix A of this manual contains a complete set of normal channel bit assignments for all STAR stations.

Normal output channel 5 is used to transmit the following classes of control information to the 7000 interface.

Terminating address
Starting address
7000 interface function bits

Control information placed on NOC-5 must be accompanied by the appropriate strobing signal on NOC-6. The strobe signal is used to identify the class of information being issued and to transfer this information from channel 5 into the interface. Figure 2-19 shows the relationship between the strobing signals and the three classes of control information.

FUNCTION CODES

Function codes, sent by the SCU to the 7000 interface, are used to prepare the interface and the peripheral subsystem for data transfer operations. The SCU places the appropriate function code or codes on the assigned normal output channel and then pulses a strobe bit (on another normal output channel) to gate the code into the 7000 interface. All function codes are formatted as shown in Figure 2-20.

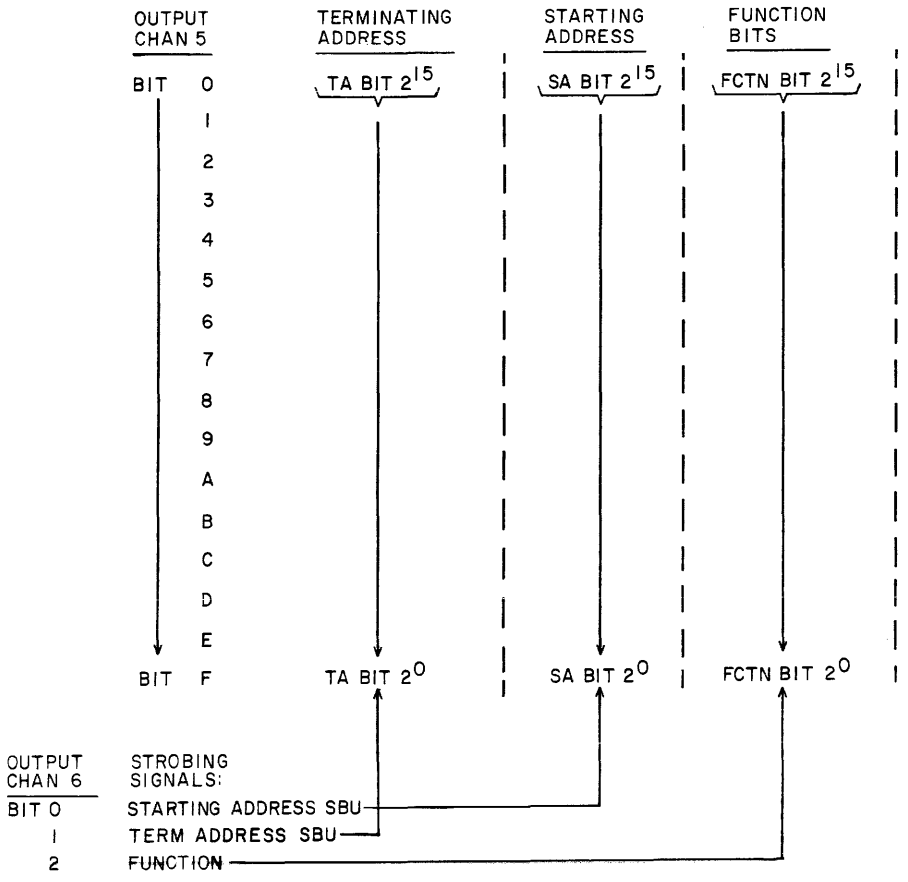


Figure 2-19. Output Channel 5 Control Information, 7000 Interface

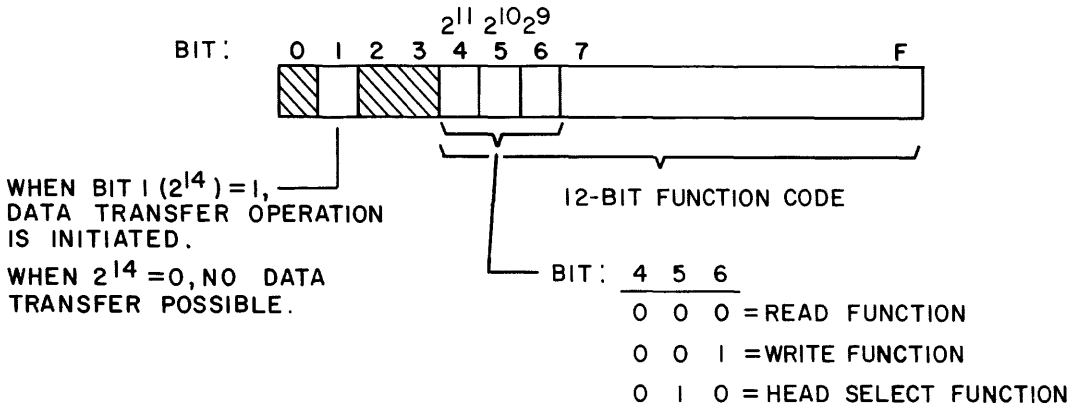


Figure 2-20. SCU/7000 Interface Function Code Format

Several function codes may be required to prepare the 7000 interface and the peripheral subsystem for a data transfer operation. Bit 2^{14} is set to indicate that a type of data transfer operation will be performed. Bits 2^{11} through 2^{9} of the 12-bit function code form a code which determines which function will be performed (Figure 2-20).

ERROR CONDITIONS

Data transfer operations between the 7000 interface and SBU memory are stopped when any one of three error conditions occurs. Factors causing each error condition to occur and the method of recovery are explained in the following paragraphs.

Drum or disk not ready	This condition indicates that the peripheral device has internal factors preventing normal operation. These factors could be such things as the device not being up to speed, air or oil temperature not within limits, device not on-line, etc.

Drum or disk parity error	This condition indicates that the 7000 storage subsystem detected a parity error at the end of a read operation or a skew error during a read operation.
Drum or disk abnormal	This condition occurs if the disk or drum becomes not ready or if the storage unit itself detects an unsafe condition. Unsafe conditions could be items such as read and write commands occurring simultaneously, commands received when disk not on-cylinder, improper number of heads selected, etc.

Once an error condition has been established, a detailed status check of the 7000 peripheral device should be made to further identify the source of the malfunction.

In all cases, the occurrence of an error condition stops the 7000 interface operation. If the error condition was an SBU parity error, the bank counter logic preserves the count relating to the address of the bank which failed. The count in the bank counter logic is four counts greater than the bank in which the error occurred. The difference is because of memory cycle timing in the SBU.

Each error condition is cleared by a program clear and must be followed by a new function code in order to reinitiate the data transfer.

STATUS

Three types of status information from the 7000 interface are available to the SCU when used in the disk station. They are normal input channels 5 and 7 and scanner status. In the drum station, the status available to the SCU is obtained via normal input channel 5 or the scanner.

NORMAL INPUT CHANNEL 5 STATUS

Several miscellaneous status bits, such as SBU parity error, disk or drum sector count, bank counter count, disk or drum abnormal, etc., appear on normal input channel 5 for both stations. Tables 2-17, 2-18, and 2-19 define each of these status bits.

SCANNER STATUS

Several 7000 interface status bits (actually from the peripheral device) are available to the SCU as part of the SBU scanner status word that appears on normal input channel 6. All 7000 scanner status bits are applicable to both the 7000-0 and 7000-1 interfaces and are listed as follows:

Disk Station	Drum Station
Disk on-cylinder	Drum sector count (2^0)
Disk error	Drum end of transfer (2^0)
Disk end of transfer (2^1)	Drum error
Disk end of transfer (2^0)	Drum end of transfer (2^1)

The meanings of these bits are explained in Tables 2-17 and 2-18.

RETURN ADDRESS STATUS

During 7000 interface/SBU memory read or write operations occurring in the disk station, the current SBU memory address is available to the SCU via normal input channel 7.

TABLE 2-17. BIT DESCRIPTIONS FOR 7000 INTERFACE/NIC-5 AND 6 (DRUM STATION)

Channel	Bit	Name	Function
NIC-5	0 (2^{15})	SBU parity error	This bit is used for indicating that a parity error occurred in SBU memory. Bank count FFs in 7000 interface stops at a $\overline{\text{count}}$ four counts greater than bank in which parity error occurred.
NIC-5	1 (2^{14}) 2 (2^{13}) 3 (2^{12})	$\overline{\text{Bank ctr}}$ 2^2 2^1 2^0	These three bits are the SBU not bank address count.
NIC-5	4 (2^{11}) A (2^5)	Drum SC 2^6 2^0	These seven bits are the drum sector count.
NIC-5	B (2^4)	Drum sector stable	This status bit indicates that drum head is not near a sector boundary and that the sector address can be sampled to obtain an accurate address location of the head.
NIC-5	C (2^3)	Drum EOT bit 2^1	This status bit represents the upper bit of a 2-bit end of transfer counter.
NIC-5	D (2^2)	Drum abnormal	This status bit indicates an abnormal condition on the 7000 interface connected to the drum.
NIC-5	E (2^1)	Drum parity error	This status bit indicates that a parity error occurred on the drum.
NIC-5	F (2^0)	Drum not ready	This status bit indicates that the drum is not physically ready to operate.
	Scanner Status		
NIC-6	0, 2	Drum SC 2^0	Used with both 7000 interfaces to denote sector count.
NIC-6	1, 3	Drum EOT bit 2^0	Used with both 7000 interfaces to denote the lower order bit of the end of transfer counter.
NIC-6	9, A	Drum error	Used with both 7000 interfaces to indicate an inclusive OR of any drum error detected.

TABLE 2-18. BIT DESCRIPTIONS FOR 7000 INTERFACING/NIC -5, -6, AND -7 (DISK STATION)

Channel	Bit	Name	Function
NIC-5	0 (2^{15})	SBU parity error	This bit is used for indicating that a parity error occurred in SBU memory.
NIC-5	1 (2^{14}) 2 (2^{13}) 3 (2^{12})	$\overline{\text{Bank ctr}}$ 2^2 2^1 2^0	These three bits are the SBU not bank address count.
NIC-5	4 (2^{11}) 9 (2^6)	Disk SC 2^5 2^0	These six bits are the disk sector count.
NIC-5	C (2^3)	Disk not on-cylinder	This bit indicates that the disk file is in the process of positioning.
NIC-5	D (2^2)	Disk abnormal	This status bit indicates an abnormal condition on the 7000 interface connected to the disk.
NIC-5	E (2^1)	Disk parity error	This status bit indicates that a parity error occurred in the disk unit.
NIC-5	F (2^0)	Disk not ready	This status bit indicates that the disk unit is not physically ready to operate.
	Scanner Status		
NIC-7	0 (2^{15}) 2 (2^{13}) 3 (2^{12})	Return function Return function	Three of these bits are function identifiers which, except for bit 2^{14}, are not used by the hardware but can be used by station software.
NIC-6	0, 4	Disk on-cylinder	Bit 0 (for 7000-0) and bit 4 (7000-1) used to indicate whether or not disk is positioned on requested cylinder.
NIC-6	1, 5	Disk error	Indicates an inclusive OR of any error detected by the 7000 interface (bit 1 for 7000-0, bit 5 for 7000-1)
NIC-6	2, 3, 6, 7	Disk EOT, 2^0 and 2^1	Two status bits used to indicate the status of the end of transfer counter. Bit 2: 7000-0 disk EOT 2^1 Bit 3: 7000-0 disk EOT 2^0 Bit 6: 7000-1 disk EOT 2^1 Bit 7: 7000-1 disk EOT 2^0

TABLE 2-19. BIT DESCRIPTIONS FOR 7000 INTERFACE/NOC-5 AND -6 (DRUM STATION)

Channel	Bit	Name	Function
NOC-5	0	EDS/SDS	EDS: The end data sequence bit accompanying the terminating address indicates that the data transfer(s) will end with the current transfer. This bit is used in streaming several data blocks in succession. SDS: The start data sequence bit accompanying the starting address indicates the start of a data transfer. This bit is used in streaming data from SBU memory to the drum when several blocks of SBU memory are transferred.
NOC-5	1 ↓ F	SA/TA/funct bit 2^{14} ↓ SA/TA funct bit 2^{0}	These bits serve three purposes. 1. Transmit terminating address to SBU. 2. Transmit starting address to SBU. 3. Transmit function code or bits to 7000 interface.
NOC-6	0 (2^{15})	Starting address SBU (7000-0 and 1)	This set/clear† bit is used to strobe the starting address, transmitted by the SCU into SBU memory for use with the selected 7000 interface.
NOC-6	1 (2^{14})	Terminating address SBU (7000-0 and 1)	This set/clear† bit is used to strobe the terminating address, transmitted by the SCU into SBU memory for use with the selected 7000 interface.
NOC-6	2 (2^{13})	Function	This set/clear† bit strobes the external function code sent by the SCU into the connected 7000 interface.
NOC-6	3 (2^{12})	Program clear	This bit is interpreted as a master clear.
NOC-6	4 (2^{11}) ↓ B (2^{4})		These bits of NOC-6 are not used with the 7000 interface.
NOC-6	C (2^{3}) ↓ F (2^{0})	Connect Code	These bits are used by the SCU to connect the 7000 interface in the SBU with the SCU normal channels (refer to Table 2-1 for codes).

† A set/clear bit is set and immediately cleared in order to perform its intended function.

PROGRAMMING SEQUENCES

The following paragraphs provide typical examples of the programming steps required to initiate and control data transfer operations through the 7000 interface. The example that follows is based upon an output data transfer to the drum.

CONNECTING TO 7000 INTERFACE

1. Place either connect code 2_{16} (7000 interface 0, drum station) or 1_{16} (7000 interface 1, drum station) on NOC-6, bits C through F. For the disk station, use 0_{16} for 7000 interface 0 or 1_{16} for 7000 interface 1.
2. If only the head selection procedure is required, proceed to Send Function in this section.

SEND STARTING OR TERMINATING ADDRESS SETS

Up to four sets of starting and terminating addresses may be sent to the 7000 interface. If a four-section data transfer is to be made, the SCU program must ensure that each address set is transferred in time to guarantee a continuous data transfer.

1. Place starting address of first data transfer on NOC-5, bits 0 through F. Set bit 0 of the first of the starting address words to inform the interface that this request is the start of a data sequence. (Refer to Figure 2-20.1.) For all other starting addresses, leave bit 0 clear.
2. Set and clear bit 0 (starting address SBU) of NOC-6 to strobe the starting address into the 7000 interface.
3. Place the terminating address of the data transfer corresponding to the starting address of step 2 on NOC-5, bits 0 through F. If this is the last terminating address of a data sequence, set bit 0 of the address. (Refer to Figure 2-20.1.) For all other terminating addresses, leave bit 0 clear.
4. Set and clear bit 1 (terminating address SBU) of NOC-6 to strobe the terminating address into SBU memory.
5. Repeat steps 1 through 4, as required.

TABLE 2-19.1 BIT DESCRIPTIONS FOR 7000 INTERFACE/
NOC-5, -6, AND -7 (DISK STATION)

Channel	Bit	Name	Function
NOC-5	0 (2^{15})	EDS/SDS	EDS: The end data sequence bit accompanying the terminating address indicates that the data transfer(s) will end with the current transfer. This bit is used in streaming several data blocks in succession. SDS: The start data sequence bit accompanying the starting address indicates the start of a data transfer(s). This bit is used in streaming data from SBU memory to the disk when several blocks of SBU memory are transferred.
NOC-5	1 (2^{14}) ↓ F (2^{0})	SA/TA/funct bit ↓ SA/TA funct bit	These bits serve three purposes: 1. Transmit starting address to SBU. 2. Transmit terminating address to SBU. 3. Transmit function code or bits to 7000 interface.
NOC-6	0 (2^{15})	SBU starting address (7000-0 and -1)	This set/clear† bit is used to strobe the starting address, transmitted by the SCU into SBU memory for use with the selected 7000 interface.
NOC-6	1 (2^{14})	SBU terminating address (7000-0 and -1)	This set/clear† bit is used to strobe the terminating address, transmitted by the SCU into SBU memory for use with the selected 7000 interface.
NOC-6	2 (2^{13})	Function	This set/clear† bit strobes the external function code sent by the SCU into the connected 7000 interface.
NOC-6	3 (2^{12})	Program clear	This bit is interpreted as a master clear.
NOC-6	4 (2^{11}) ↓ B (2^{4})		These bits of NOC-6 are not used with the 7000 interface.

†A set/clear bit is set and immediately cleared in order to perform its intended function.

TABLE 2-19.1 BIT DESCRIPTIONS FOR 7000 INTERFACE/
NOC-5, -6, AND -7 (DISK STATION) (Cont'd)

Channel	Bit	Name	Function
NOC-6	C (2^3) ↓ F (2^0)	Connect code	These bits are used by the SCU to connect the 7000 interface in the SBU with the SCU normal channels (refer to Table 2-1 for codes).
NOC-6	0 (2^{15})	EDS/SDS	Same function as that given above for NOC-5, bit 0.
NOC-7	1 (2^{14}) ↓ F (2^0)	SA/TA/funct bit ↓	Same three functions as those given above for NOC-5, bits 1 through F.

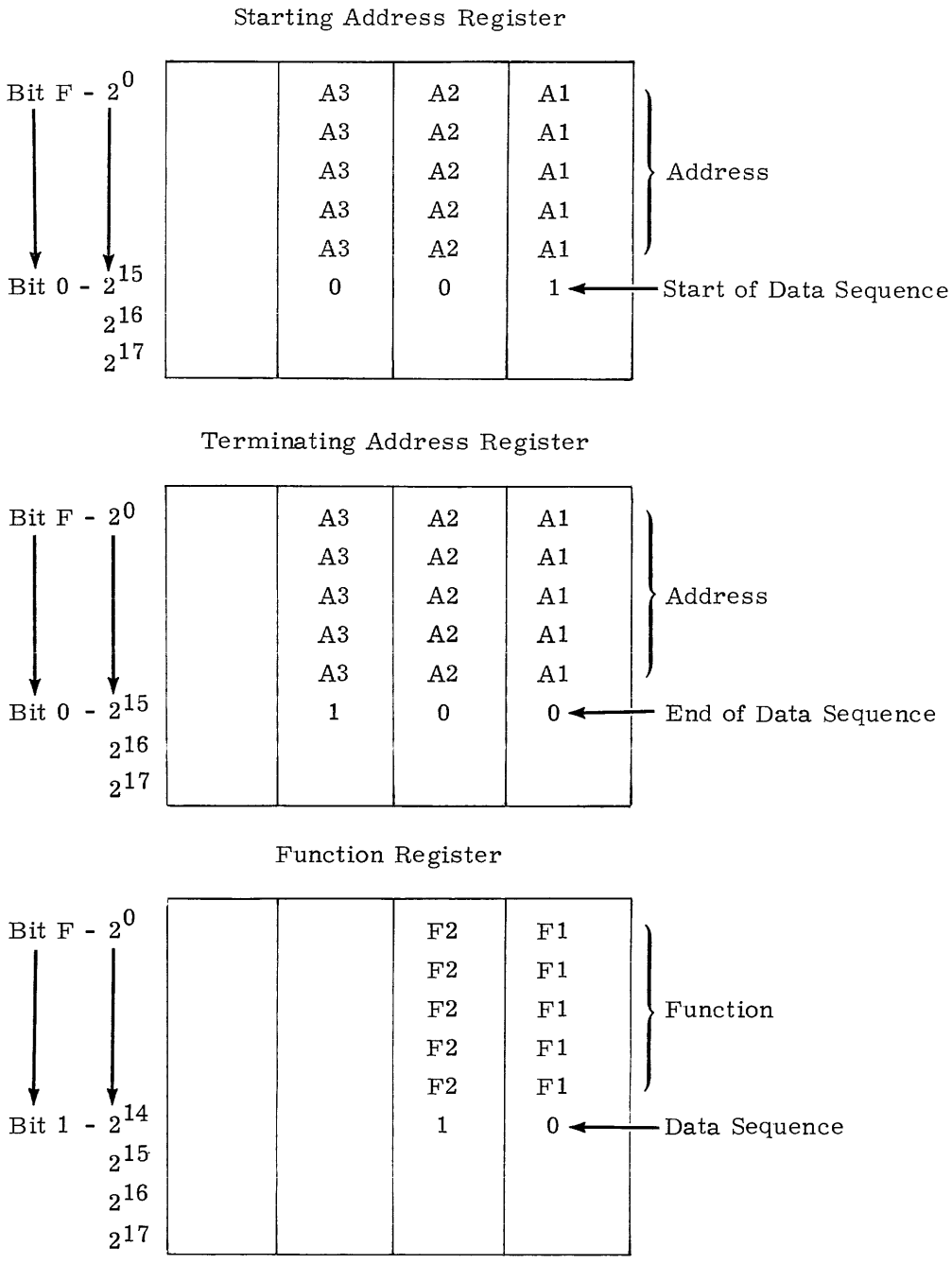


Figure 2-20.1 Typical Register Configuration For a Data Transfer To The Drum

SEND FUNCTION

1. Place the desired function code (such as read, write, or head select) on NOC-5, bits 0 through F. If the function is a read or write, set bit 1 of NOC-5. (Refer to Figure 2-20.1.)
2. Set and clear the function strobe bit (NOC-6, bit 2) to strobe the function into the function register of the 7000 interface. The 7000 channel interface is now primed with a request, and initiation of the requested operation starts immediately.

SETTING STATUS

1. Perform a connect function as described in Connecting to 7000 Interface.
2. Obtain the following status.

   Controller status (disk or drum)
   Sector address
   End of transfer - EOT 2^1 (NIC-5, bit C) (drum station)
   EOT 2^0 (although not scanned, appears in scanner)

INDEX COMPARE LOGIC

The SBU index compare (IC) logic provides a hardware search of a virtual address list contained in the even banks of SBU memory at the request of the SCU (Figure 2-21). In general, this procedure is accomplished by the IC logic searching a table in SBU memory for an entry that corresponds to conditions specified by the SCU program. Although each list entry is a 64-bit word, the IC logic processes it as four 16-bit words. Portions of each entry relate a virtual address (33 bits) and a 12-bit job identifier (called a key) to a specific 16-bit physical address. The IC logic simultaneously searches for two different entries that meet the requirements of a virtual address and the key as directed by the SCU.

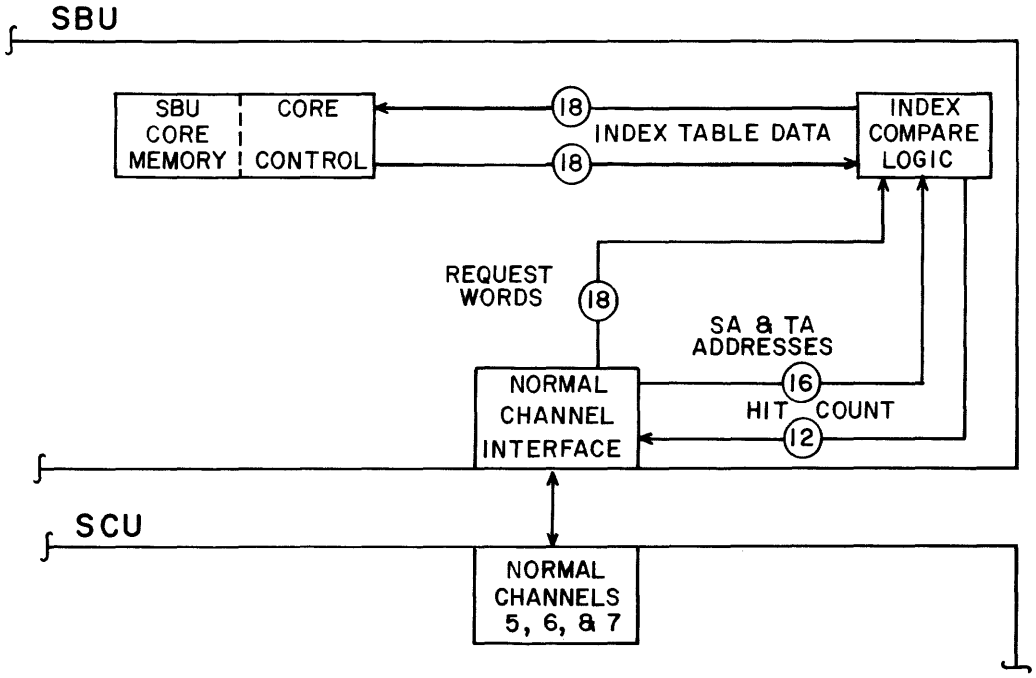


Figure 2-21. Index Compare System Relationship

To more fully understand the following paragraphs, the reader should be familiar with the operation of the SBU normal channel interface, the BC, and the organization of SBU memory.

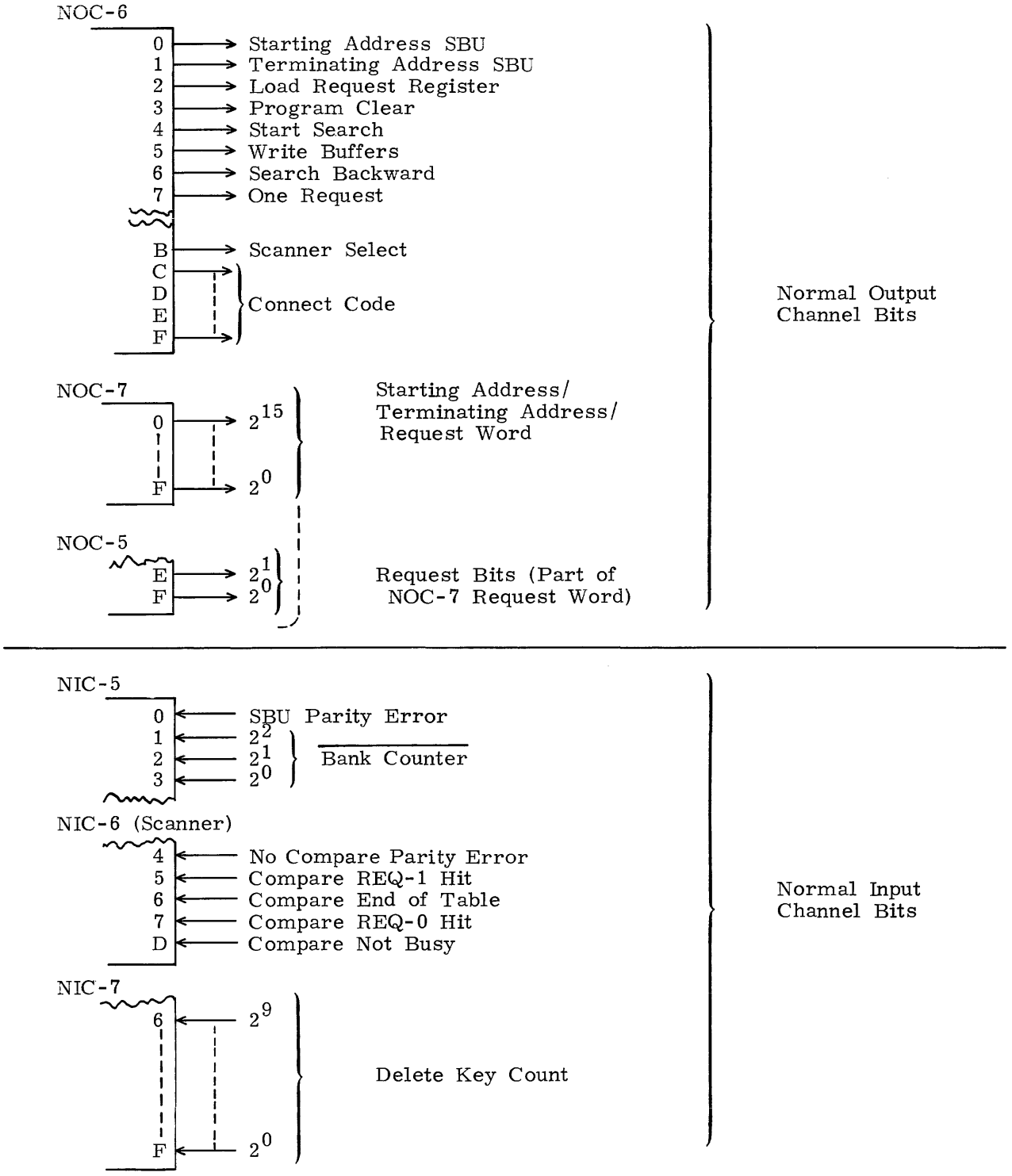


Figure 2-22. Normal Channel Bit Assignments, Index Compare Logic (Drum Station)

SCU CONTROL OF INDEX COMPARE LOGIC

The SCU controls the IC logic using control information issued on normal output channels 5, 6, and 7 (Figure 2-22). Also, the status of various operations occurring within the IC interface is monitored via normal input channels 5, 6, and 7.

There are three basic types of operations that the SCU may instruct the IC logic to perform: the key search, key and virtual address search, and delete key. Each of these operations search a table of 64-bit entries in SBU memory. Prior to the beginning of each type of operation, the SCU instructs the IC logic as to the objective of the search and defines the limits of the area in memory to be searched. The format of each 64-bit entry in the SBU memory table is shown in Figure 2-23. Since the table is in the even memory banks only, the first word is in bank 0, the second in bank 2, etc.

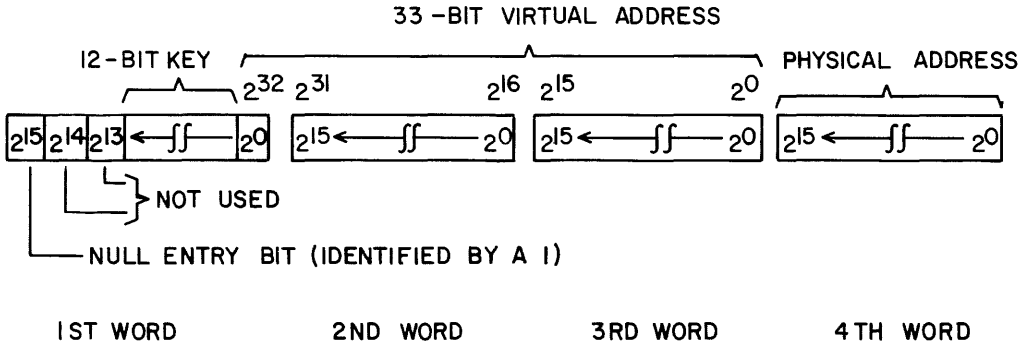


Figure 2-23. Memory Table Word Format

SCU SEARCH REQUESTS

The SCU may load as many as six 18-bit search request words into the IC logic which compares the request words against table entries from SBU memory. The SCU may issue several combinations of requests which may instruct the IC logic to search for a single key, two keys, or one or two keys and virtual address combinations as the objects of the search through memory. Special operations include a search backward through the table and delete key where a table entry is effectively deleted by setting the null bit (2^{15}) in the key. Typical request word formats are shown in Figure 2-24.

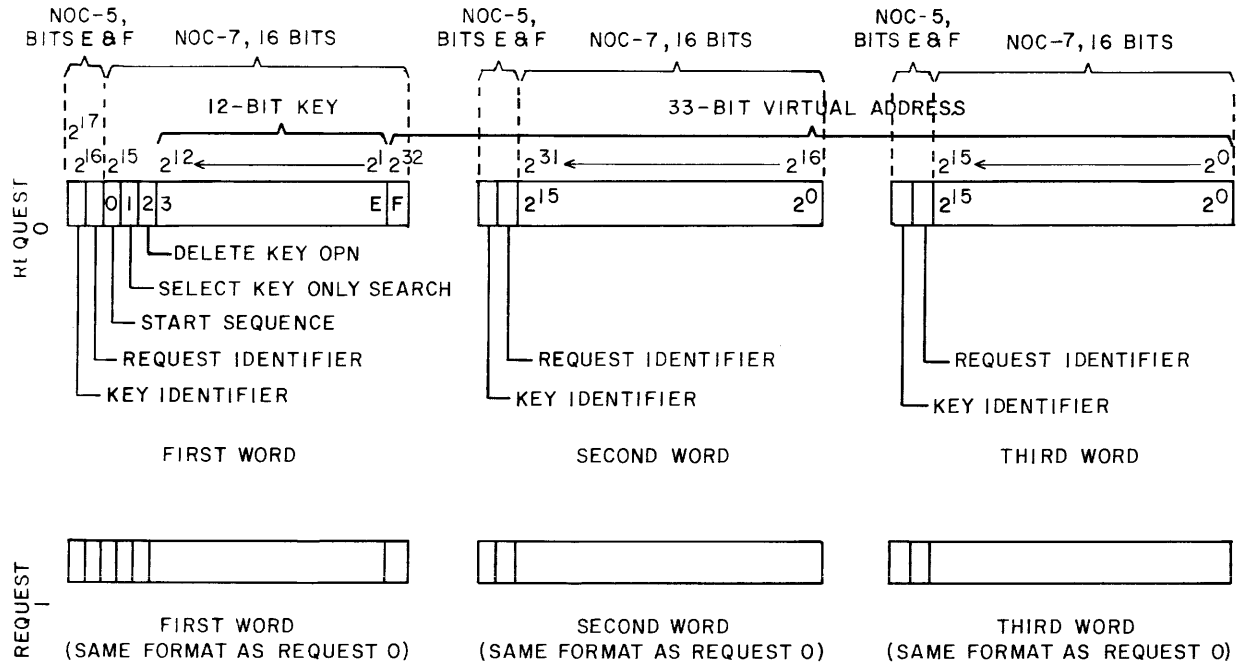
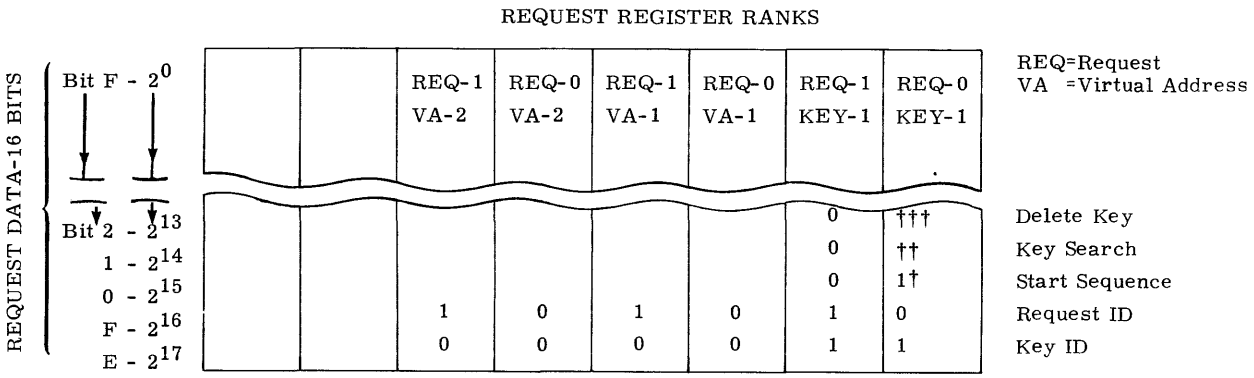


Figure 2-24. Typical SCU Request Word Formats

The request words illustrated in Figure 2-24 indicate that the IC logic searches for two keys and virtual address combinations. Note that only the first word of a request should contain a 12-bit key. This key is usable to the IC logic and SBU memory only if the key identifier bit (2^{17}) is set. Note also that the last two bits (2^{16} and 2^{17}) of all request words are transferred to the IC logic via NOC-5, bits E and F. The key identifier bit of the second and third words of request 0 must be cleared.

The request identifying bit (2^{16}) in each of the three request words must be the same for one request so that the individual words of a request can be identified as belonging to either request 0 or 1. When the SCU transfers request words to the IC logic, the individual words of the two requests (requests 0 and 1) are alternated; that is, first word of request 0, first word of request 1, second word of request 0, etc. Therefore, the request ID bit (2^{16}) must be changed each time another word of a request is sent to properly identify the request.

For different search operations, the request registers and control bits (2^{16} and 2^{17}) are set up in different manners. Figure 2-25 illustrates the proper setup for the operations as noted on the figure. It is important to note that the request registers must be loaded properly as shown (for the operations indicated) so that logic circuits within the IC logic function correctly.



Notes:
† When set, this bit (2^{15}) indicates the start of a sequence of requests. It must always be set in REQ-0 and KEY-1, and cleared in all other key entries.

†† When set, this bit (2^{14}) indicates a key search operation. Since a key search requires that it be the only request, all other entries in the request register are not used. The One Request bit (NOC-6, bit 7) and the Start Sequence bit (2^{15}) must both be set.

††† When set, this bit (2^{13}) indicates a Delete Key operation. The Key Search (2^{14}), Start Sequence (2^{15}), and One Request (NOC-6, bit 7) bits must all be set.

Figure 2-25. Request Register Loading

Once a search request from the SCU has been recognized and a type of search operation is in progress, only the following conditions may terminate the search.

1. Conditions occurring which satisfy the search before end of table is reached.
2. The search procedure reaching the end of the table.
3. A parity error on data read from SBU memory (called compare error).
4. A master or program clear.

KEY SEARCH OPERATION

Prior to the search of a key in SBU memory, the SCU loads the terminating address into the IC logic with the upper boundary memory address of the table. This is done via NOC-7. Setting bit 1 of NOC-6 strobes the address into the IC logic. The SCU then loads the starting address into the IC logic with the starting or lower boundary memory address of the table. This is also accomplished using NOC-7. Setting bit 0 of NOC-6 strobes this address into the IC logic. Note that the loading process may begin with either the terminating or starting address prior to issuing a request.

The next step the SCU performs in the key search operation is to issue one or two search requests depending upon the desired end result of the controlling program. Refer to Figure 2-25 and note that the proper bit or bits of the first word of any request must be set to initiate an operation. If only one search request is issued by the SCU, bit 7 of NOC-6 (one request bit) must be set so that the IC logic can search for a null list entry in the memory table (Figure 2-24).

If the SCU issues two requests for a key and virtual address search operation, the procedure is completed as follows:

1. The IC logic reads entries sequentially from the SBU memory table and compares that data with each of the two request entries received from the SCU. The comparison procedure begins by comparing the first word read from the memory table with the first word of request 0. Then the same memory table word is compared with the first word of request 1. The second word from the memory table is compared with the second word (first half of virtual address) of request 0. The comparison procedure continues through the end of the third word, alternating between the words of request 0 and request 1.
2. If the key/virtual address combination read from the memory table matches the key/virtual address combination of either request from the SCU, an equal-compare condition occurs. When this happens, that entire memory table entry is stored in a buffer of the IC logic and normal input channel 6, status bit 7 (compare request-0 hit) is set.
3. After the first equal-compare condition occurs, the search continues through the memory table for an entry that matches the second request entry from the SCU.

4. When a second equal-compare condition occurs, that memory table entry is stored in an IC logic buffer and normal input channel 6, status bit 5 (compare request-1 hit) is set.

5. Data that has been temporarily stored in the IC logic buffers is then stored in SBU memory at a starting address specified by the SCU and the issuance of the write buffers bit (NOC-6, bit 5) by the SCU.

The searching process through memory may proceed either forward or backward. To search forward, the SCU sets bit 4 of NOC-6 (start search) and the search proceeds sequentially through the memory table as specified by starting and terminating addresses issued by the SCU. To search backward through the memory table, the IC logic must be properly connected. Bit 6 of NOC-6 (search backward) must be set, and the SCU must set bit 4 of NOC-6 (start search) to start the search backwards. Also, the SCU issues the complement of the address of the last list entry to the IC logic as the starting address and the uncomplemented address of the top of the list as the terminating address. If the SCU issues only one request to the IC logic, it also sets bit 7 (one request) of NOC-6 to inform the interface that only one request was issued. The procedure followed is similar to that of a two-request key and virtual address search operation except that the IC logic searches the memory table for a null entry in place of the second request. A null entry is any entry in which bit 2^{15} of the first word is a 1 (Figure 2-24). When an equal-compare condition occurs, that memory table entry is stored in a buffer in the IC logic and a normal input channel status bit is set (either bit 5 or 7 of NIC-6).

DELETE KEY OPERATION

During a delete key operation, the IC logic searches the SBU memory table for entries containing a specified key. Each entry in the table containing that key is effectively deleted from the table by writing a null entry (set bit 2^{15} of the first word) in its place. All null entries are transferred to SBU memory except the first which remains in a data buffer in the IC logic. Each null entry increments the delete key hit counter which the SCU monitors via NIC-7, bits 6 through F. The delete key counter indicates the total number of entries altered to nulls and operates on only one delete key request at a time. If a second delete key request is required, it must be loaded into the IC logic after the first delete key operation is complete. Attempting to load one delete key request into request 0 and another into request 1 is an illegal operation.

Bit 2^{13} of the first request word of request 0 must be set to perform a delete key operation. In this operation, the search through the memory table does not stop even though a satisfying condition is found. Instead, the search continues through the table for all specified keys matching the SCU's request until the end of the table is reached. The other conditions which may terminate a delete key operation prior to reaching the end of the table are a parity error or master or program clear. Entries temporarily stored in the IC logic buffers may be stored back in SBU memory exactly as entries are stored for the key search operations.

NORMAL CHANNEL BIT ASSIGNMENTS

All of the normal channel bit assignments used to control the IC logic are defined either in the following paragraphs or in Tables 2-20 and 2-21. The explanations of bit assignments given apply to the drum peripheral station. If the IC logic is used in other stations, bit assignments may be different. Appendix A contains a complete set of normal channel bit assignments for all STAR stations.

The SCU uses normal output channel 7 to transfer the following classes of control information to the IC logic.

Starting address
Terminating address
IC logic function bits
Search keys

Control information placed on NOC-7 must be accompanied by the appropriate strobing signal on NOC-6. The strobe signal is used to identify the class of information being issued and to transfer the information from NOC-7 into the IC logic.

TABLE 2-20. BIT DESCRIPTIONS FOR INDEX COMPARE/NOC-6 AND 7 (DRUM STATION)

Channel	Bit	Signal Name	Function
NOC-5	E (2^1)	SCU key identifier	When set, this bit indicates that the first word of a request contains a 12-bit job identifier.
NOC-5	F (2^0)	SCU request identifier	This bit identifies the request (0 or 1) to which each 18-bit request word belongs.
NOC-6	0 (2^{15})	Starting address SBU	This bit gates the starting address on NOC-7 into the starting address register.
NOC-6	1 (2^{14})	Terminating address SBU	This bit gates the terminating address on NOC-7 into the terminating address register.
NOC-6	2 (2^{13})	Load request register	This bit is used for gating a request word from the SCU into the request register.
NOC-6	3 (2^{12})	Program clear	This bit is interpreted as a master clear.
NOC-6	4 (2^{11})	Start search	This bit enables the start of the search of the table in SBU memory.
NOC-6	5 (2^{10})	Write buffers	This bit enables the transfer of data from the buffer registers of the compare interface to SBU memory.
NOC-6	6 (2^9)	Search backward	Use of this bit enables the index compare logic to search backward through the table in core memory.
NOC-6	7 (2^8)	One request	The set condition of this bit initiates the one request circuit in the index compare logic and causes it to search for a null list entry in addition to an entry equal to a single request.
NOC-7	0 (2^{15}) ↓ F (2^0)	SA/TA/request bit 2^{15} ↓ SA/TA/request bit 2^0	These bits serve three purposes. 1. Transmit starting address to SBU memory via IC logic. 2. Transmit terminating address to SBU memory via IC logic. 3. Transmit bits of request words to IC logic.

TABLE 2-21. BIT DESCRIPTIONS FOR INDEX COMPARE/NIC-5, -6, AND -7

Channel	Bit	Name	Function
NIC-5	0	SBU parity error	This status bit indicates that a parity error occurred in SBU memory.
NIC-5	1 2 3	Bank ctr 2^2 2^1 2^0	When an SBU memory parity error is detected, the memory bank with the parity error is found by complementing the bank count status and then counting back four memory banks.
NIC-6	4	Compare not error	This scanner status bit indicates whether or not a parity error occurred on data as it was read from SBU memory.
NIC-6	5	Compare request-1 hit	This scanner status bit is set if a match was made with request 1.
NIC-6	6	Compare end-of-table	This status bit indicates that the current address in the IC logic matches the terminating address and therefore means that the table search is complete.
NIC-6	7	Compare request-0 hit	This scanner status bit is set if a match was made with request 0.
NIC-6	D	Compare not busy	This scanner status bit indicates whether or not the IC logic has completed an address compare procedure.
NIC-7	6 ↓ F	Delete key count bit 2^9 ↓ 2^0	These status bits indicate the total number of hits produced by altered table entries during a delete key operation (up to the limits of the counter).

PROGRAMMING SEQUENCES

The following procedures provide typical examples of the programming steps required to initiate and control address compare operations through the IC logic.

CONNECTING TO IC INTERFACE

1. Transmit the appropriate connect code (3_{16}) from the SCU to the IC logic using NOC-6, bits C through F.
2. Set and clear the program clear bit (NOC-6, bit 3).

TRANSMITTING KEY(S) FOR REQUEST(S)

1. Set or clear the request identifier bit (bit F of first word of request 0 or 1; refer to Figure 2-24). Set the key identifier bit.
2. Set the appropriate key bit (bit 2 for delete key operation or bit 1 for select key only search; refer to Figure 2-24). Set bit 0 for request 0, word 0 (start sequence).
3. Transmit the key bits (bits 2^1 through 2^{12} of first word of request 0 or 1) and bit 2^{32} of the virtual address to the IC logic via NOC-7.
4. Set and clear the load request register bit (NOC-6, bit 2).
5. Clear all bits set in step 2.
6. Repeat steps 1 through 5 to transmit a second request.

TRANSMITTING VIRTUAL ADDRESS PORTION(S) OF REQUEST(S)

1. Transmit second word of request via NOC-5, bits E and F, and all bits of NOC-7 to IC logic. Clear the key identifier bit via NOC-5.
2. Set and clear load request register bit via NOC-6, bit 2.
3. Repeat steps 1 and 2 for a second request (if used).
4. Transmit third word of request via NOC-5, bits E and F, and all bits of NOC-7 to IC logic.

5. Set and clear load request register bit via NOC-6, bit 2.
6. Repeat steps 4 and 5 for the third word of a second request (if used).
7. Set one request bit via NOC-6, bit 7, if only one request is made.

SET SEARCH LIMITS FOR COMPARATOR

1. Transmit starting address bits via NOC-7, bits 0 through 5.
2. Set and clear strobe starting address SBU bit (NOC-6, bit 0).
3. Transmit terminating address bits via NOC-7, bits 0 through 5.
4. Set and clear strobe terminating address SBU bit (NOC-6, bit 1).
5. Set and clear strobe start search bit (NOC-6, bit 4).

After completion of the previous procedures, the index compare logic initiates the address search. An exit is then made to the scanner program to wait for completion of the search. The comparator response routine then causes the comparator to check for a compare error.

844 INTERFACE

The interface controls data transfers between the 844 disk storage units and the SBU memory. The SBU in 844 storage/magnetic tape stations contains two 844 interfaces which may connect to the same eight disk storage units when the units are in dual channel configuration. Otherwise, each interface can control up to eight units but can transfer data to only one unit at a time. Each interface is assigned a different connect code for selection by the SCU. The two interfaces are completely independent and can transfer data simultaneously. However, the simultaneous data transfer must be to or from different units. In an 844 Service Station, the SBU contains only one 844 interface which can control up to three units but transfers data to only one unit at a time. Figure 2-26 shows the relationship between the interface and system elements with which it communicates.

Operation of the interface is controlled by the SCU through bits on normal channels 6 and 7. Once the SCU has initiated a data transfer operation, it need not remain connected to the interface except to check status.

Normally, the interface completes the operation without further SCU control. Principal control responsibilities of the SCU in relation to the interface are as follows:

Issues connect code for interface.
Issues starting address in SBU memory for data page transfer.
Selects unit, cylinder, and sector of the unit.
Issues read, write, and write tags commands to initiate data transfer operations.
Monitors status conditions.

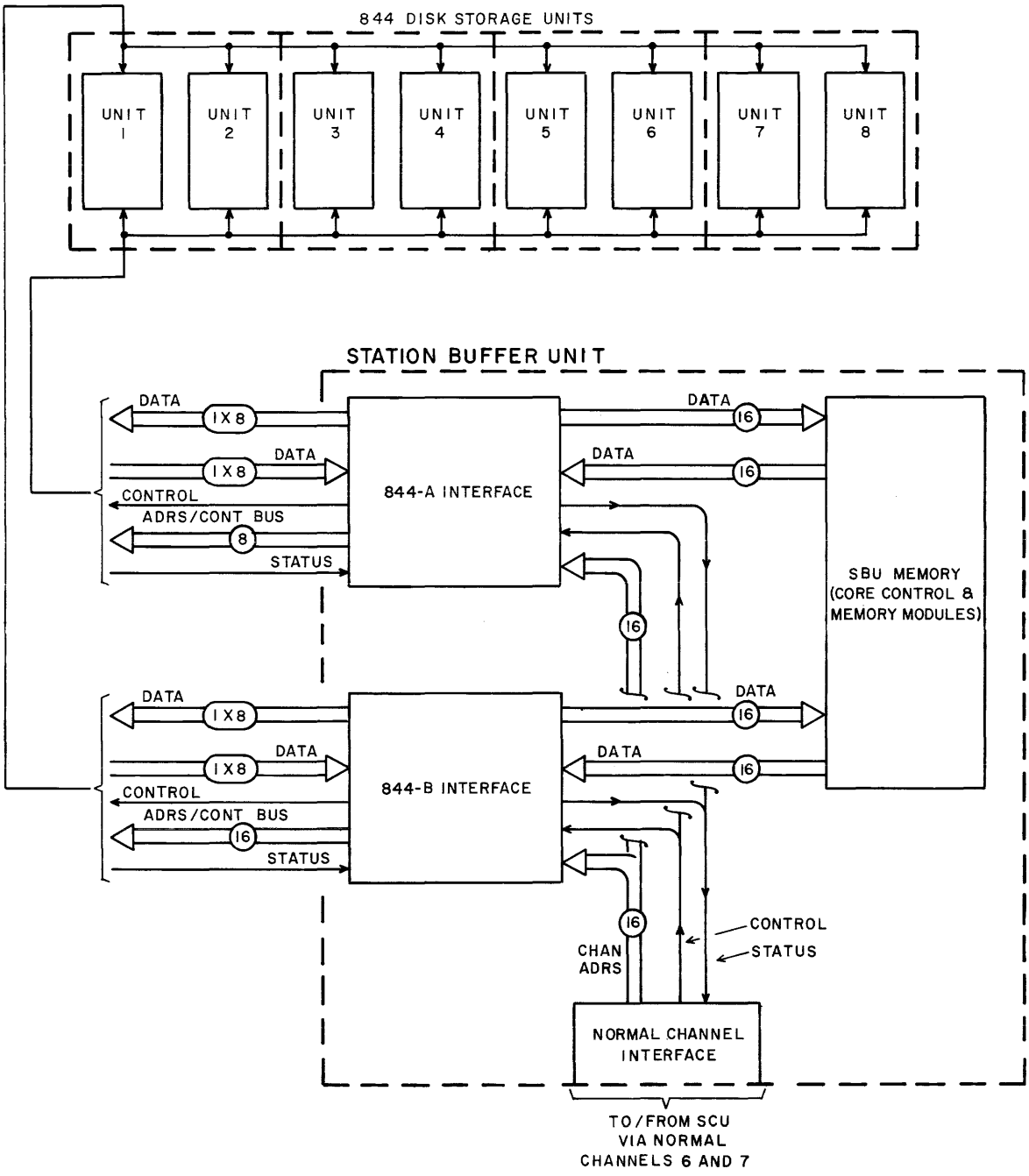


Figure 2-26. 844 Interface System Relationship

844 DISK STORAGE UNIT DESCRIPTION

The 844 disk storage unit is a high-speed, random access, disk storage device. Data is recorded on removable disk packs. Up to eight units are used on line with the 844 interface.

Each disk pack contains 12 disks mounted on a common vertical spindle. Only 10 disks are used for recording. The top and bottom disks are protective nonrecording disks. The access mechanism for each unit consists of 20 arms mounted on a movable carriage. Nineteen read/write heads and one track servo head are mounted on the ends of the arms. The track servo head and surface are used for the reading of prerecorded information that positions the heads to the desired tracks. Figure 2-27 shows the location of the heads in relation to the recording surfaces.

An initial seek operation is performed by loading a unit, closing the front panel, and pressing the START switch. The carriage then moves horizontally from a retracted position and positions the heads at the outermost track (000). The heads are now in a recording attitude near the surfaces of the disks.

The 19 read/write heads are aligned in a vertical line to enable access to corresponding circular tracks on the 19 disk surfaces. The 19 aligned circular tracks are called a cylinder. The access mechanism can be stopped at any one of the 404 track positions plus seven spare positions. This provides 404 cylinders of data or 7676 data tracks for each unit. The cylinders are numbered 000 to 403 from the outside ring to the inside ring. The address of an individual track in a given unit consists of the cylinder number and read/write head number.

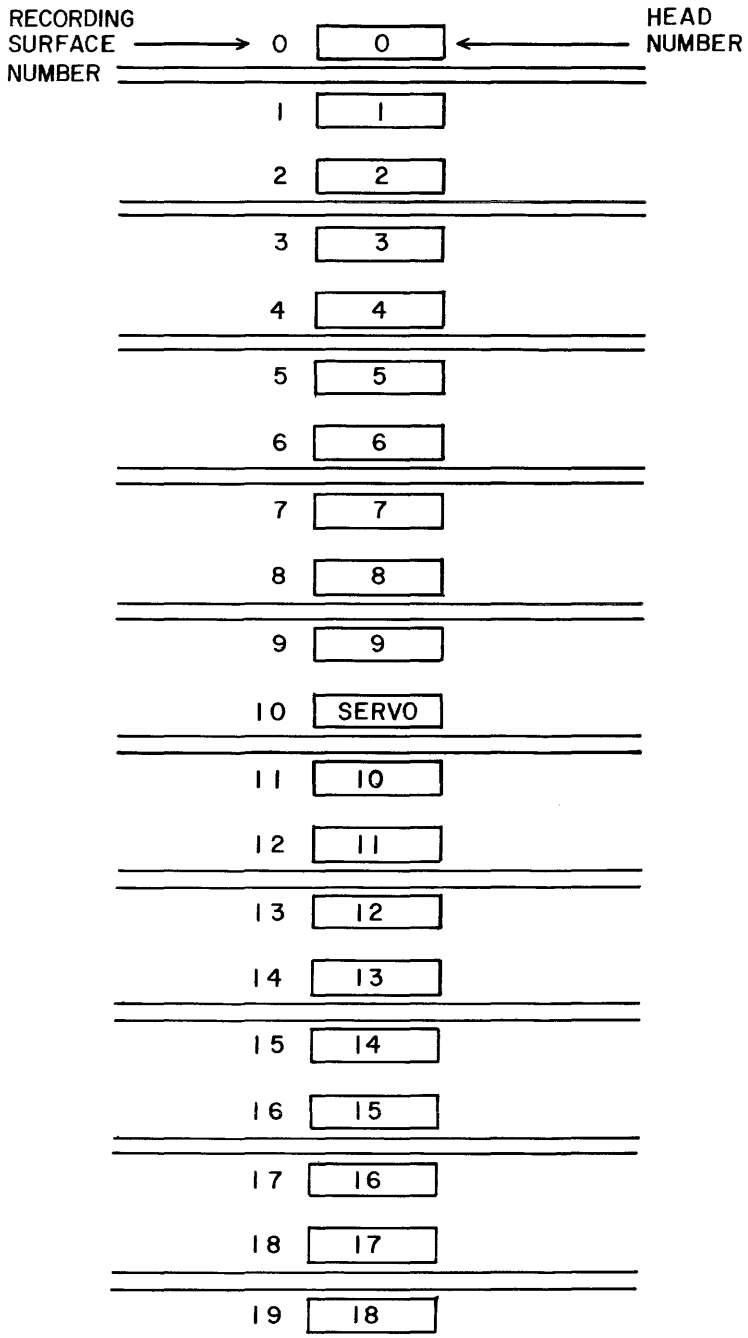


Figure 2-27. Read/Write Heads and Disk Recording Surfaces

844 SECTOR RECORDING

GENERAL

Each disk surface of a unit is divided into three equal-sized logical sectors. The logical sector is the smallest addressable unit on the disk. All read or write operations must start at the beginning of a logical sector. The interface transmits one page of data and its associated address tagging information to the unit during each one-third revolution of the disk. A data page is equal to 2048 16-bit words.

The unit records information serially bit by bit on the disk. The format for sector recording is shown in Figure 2-28.

The head gap, sync pattern, address tag word, and address check code are written on all sectors when a disk pack is first put into use. This is a write tags operation and is normally a one-time operation for each disk pack.

The second head gap, second sync pattern, preamble, data page, block check code, and end of record are the only areas on a sector that can be altered by a normal write operation.

HEAD GAP

The head gap is a block of all 0's which is equivalent to the physical distance of the read/write-to-erase gap on the disk.

SYNC PATTERN

The sync pattern consists of all 0's with a single 1 bit at the end. The 1 bit (sync bit) provides a known starting point in the bit stream that signals the beginning of useful data. A search for the sync bit takes place prior to a read or write operation.

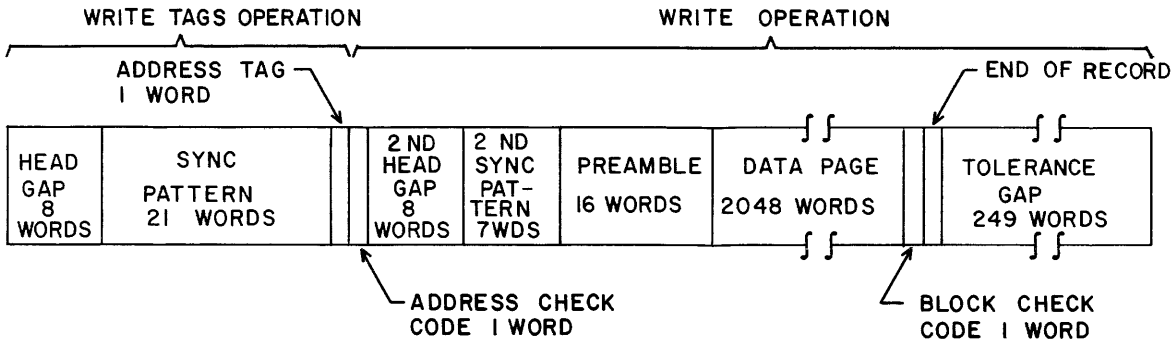


Figure 2-28. 844 Sector Recording

ADDRESS TAG WORD

The address tag word designates the area on the disk to be accessed. Each sector contains a unique address tag word that distinguishes it from all other sectors within a disk pack. Figure 2-29 shows a detailed breakdown of the address tag word. Bits 2^0 and 2^1 designate one of the three sectors on the track. When the sector code is 00, sector 0 on the disk is designated. A 01 code designates sector 8, and a 10 code designates sector 16 on the disk. This is because the disk is divided in three parts for data page recording. Bits 2^2 through 2^6 designate one of 19 read/write heads (tracks) numbered 0 through 18. Bits 2^7 through 2^{15} designate one of 404 cylinders numbered 0 through 403.

ADDRESS CHECK CODE

The address check code is a 16-bit word for detecting errors in the transmission of the address tags. During a write tags operation, the software generates a 16-bit check code that is unique to the address tag word. This check code is written on the disk immediately after the address tag word. Previous to a read or write operation, the address tag information is read from the disk into the interface. The information passes through a coder in the interface leaving a code. Now as the address check code is read from the disk, it should clear the coder. If not, transfer error and tag BCC error status bits are sent to the SCU.

PREAMBLE

The 16-word preamble provides software identification information of the data page that follows.

DATA PAGE

The data page contains 2048 16-bit words.

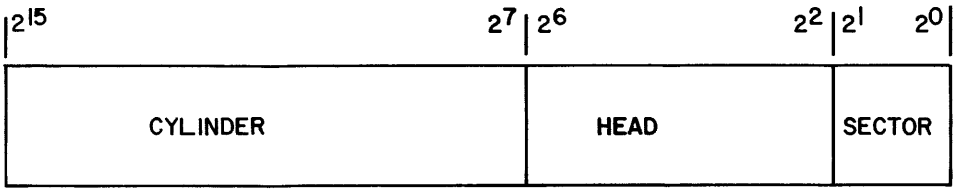


Figure 2-29. Address Tag Word

BLOCK CHECK CODE

The block check code (BCC) is a 16-bit word for detecting errors in the transmission of the address tag and data page. During an 844 write operation, the interface coder creates a BCC that is unique to the data page. The BCC is written on the disk after the data page. During an 844 read operation of the data page, the coder creates another BCC which is compared with the recorded BCC on the disk. If the two BCCs do not compare, transfer error and read error status bits are sent to the SCU.

END OF RECORD

The end of record is 16 zero bits written after the BCC before the write operation terminates.

TOLERANCE GAP

The tolerance gap is an unrecorded area on the disk that separates the BCC and the next sector mark.

DATA TRANSFER OPERATIONS

GENERAL

The SCU initiates and selects all operations through the normal channels. The types of transfer operations performed by the 844 interface include 844 write, 844 read, and write tags. The following initial procedure is always performed prior to any transfer operation.

INITIAL PROCEDURE

Upon detecting a transfer request, the SCU software performs these tasks.

Connects the appropriate 844 interface. Each of the two interfaces has its own connect code.

Selects a particular 844 unit.

Positions the read/write head assembly to the proper cylinder (cylinder select).

Selects the proper sector on the disk and waits for 75 microseconds after receiving on-sector status from unit.

Places header information including address tag word into page 0 of SBU memory. Formats differ between write tags operation and write/read operation and are given in the paragraphs for these operations. The tag word is identical to the tag word written at the beginning of the sector. It is used by the interface for head selection.

Loads starting address for SBU memory into 844 interface, sends full control bit, and selects write tags, write, or read operation at the same time. Interface should return a full status.

Provides a begin transfer command to the interface.

The 844 interface then reads the first word (address tag word) from memory. The head address portion of the address tag word is transmitted through the interface to the 844 unit. The unit then selects the addressed head. The interface now waits for the next sector mark signal from the unit.

Detection of the sector mark signal alerts the interface that address tag information (head gap, sync pattern, and address tag word) is being read from the sector. The interface now does a search for the 1 bit at the end of the sync pattern. Detection of the sync bit enables the interface to accept useful data from the disk.

After the sync bit, the address tag word is read from the sector into the interface. The interface does an address tag word compare against the address tag previously read from SBU memory. If the compare fails, a transfer error status bit is sent to the SCU.

Following the address tag word, the interface reads the address check code from the disk and performs a check. If the check fails, transfer error and tag BCC error status bits are sent to the SCU. If the check passes, the interface drops the full status.

WRITE TAGS OPERATION

The write tags operation is a one-time operation used to write address tag information at the beginning of all sectors in a disk pack (refer to Figure 2-28 for the sector recording format). These tags are read as addresses during read and write operations.

The cylinder and head portions of the address tag word correspond to the actual cylinder and head numbers of the disk. Sectors 0, 1, and 2 of the sector portion of the address tag word correspond to sectors 0, 8, and 16 of the disk.

Prior to selecting a write tags operation on the normal channels at the time the starting address is loaded, the SCU software uses the SCU coupler to place the address tags to be written on the disk into a 32-word section in page 0 of SBU memory. Page 0 is always reserved for address tags and does not contain data.

Page 0 is divided into 64 32-word sections. This format is intended to limit addressing to 32-word boundaries. Therefore, the addresses for the sections are 0000_{16}, 0020_{16}, 0040_{16}, 0060_{16}, 0080_{16}, $00A0_{16}$, etc. Only a few of the sections are used at any one time for address tags. Any of the 64 sections may be selected for the tags. Using the section that begins with address 0020_{16} as an example, the tags for one sector would be located in page 0 as follows:

Word 0020_{16} = address tag word (1 word)
Words 0021_{16} - 0028_{16} = head gap (8 words, all 0's)
Words 0029_{16} - $003D_{16}$ = sync pattern (21 words, all 0's except last bit is a 1)

Word $003E_{16}$ = address tag word (1 word, repeat of word 0020_{16})
Word $003F_{16}$ = address check code (1 word)

The first address tag word in memory is for head selection, as described in Initial Procedure, and is not written on the sector during a write tags operation.

Selection of the write tags operation enables the interface to bypass the address tag read and verify sequence described in Initial Procedure since none exists on the disk.

Following detection of the first sector mark after the on-sector status signal, the head gap, sync pattern, address tag word, and address check code are written at the beginning of the first sector.

After the tags for the first sector are written on the disk, a finished status bit is sent to the SCU. The SCU software now selects the next consecutive sector by selecting cylinder, selecting sector, loading starting address, and sending a begin transfer command. This procedure continues until tags are written for all sectors in a pack.

WRITE OPERATION

The SCU software selects a write operation at the time the starting address is loaded as described in Initial Procedure.

Prior to selecting a write operation on the normal channels, the SCU software uses the SCU coupler to place the address tag word, second head gap, second sync pattern, and preamble into page 0 of SBU memory (refer to Figure 2-28 for the sector recording format). As explained in Write Tags Operation, the page is divided into 32-word sections. Using the section that begins with address 0020_{16} as an example, the header information for a sector for a write or read operation would be located in page 0 as follows:

Word 0020_{16} = address tag word (1 word)
Words 0021_{16} - 0028_{16} = second head gap (8 words, all 0's)
Words 0029_{16} - $002F_{16}$ = second sync pattern (7 words, all 0's except last bit is a 1)
Words 0030_{16} - $003F_{16}$ = preamble (16 words)

The address tag word is the one used for head selection and address tag compare, mentioned in Initial Procedure, and is not written on the sector during a write operation.

After the address check code word is read from the sector, as described in Initial Procedure, a write operation enables the second head gap, second sync pattern, and preamble to be written in the sector.

Once the header information is completely written on the sector, the full status drops and the interface addresses a data page in memory (1 through 15) and writes the entire data page in the sector. The memory page addressed is specified by the starting address loaded during the initialization procedure.

The interface also generates a block check code (BCC) and writes this after the data page. The unit then writes an end of record as the last word in the sector. This completes the write operation, and a finished status bit is sent to the SCU.

The remaining space on the disk before the next sector is an unwritten tolerance gap. After the software detects the finished status, it may write the next consecutive sector by sending a begin transfer command (provided header information was placed in SBU memory and a starting address was loaded when the full status dropped).

READ OPERATION

The SCU software selects a read operation at the time the starting address is loaded as described in Initial Procedure.

Prior to selecting a read operation on the normal channels, the SCU software uses the SCU coupler to place the address tag word, second head gap, and second sync pattern into page 0 of SBU memory. Refer to Write Operation for memory location of this header information.

After the address check code word is read from the sector, as described in Initial Procedure, a read operation enables the interface to skip over the second head gap and begin a search for the 1 sync bit in the second sync pattern.

After the sync bit is read, the preamble is read from the disk into page 0 to complete the 32-word section.

Once the header information is completely read from the sector, the full status drops and the interface addresses a data page in memory (1 through 15) and transfers the entire data page from the sector to memory. The memory page addressed is specified by the starting address loaded during the initialization procedure.

Next, the interface does a check when the BCC is read from the sector. If the check fails, a read error status bit is sent to the SCU. If the check passes, a finished status bit is sent to the SCU.

After the software detects the finished status, it may read the next consecutive sector by sending a begin transfer command (provided header information was placed in SBU memory and a starting address was loaded when the full status dropped).

SBU MEMORY ADDRESSING

The SCU software must issue a starting address to the 844 interface at the time a read or write operation is initiated. The starting address specifies the location of the data page and also the location of the 32-word block of header information (address tags and preamble) for the data page. Since the data page is confined to only 2048 words of data, the software must reserve a separate page for the accompanying 32-word block of header information.

The ODD/EVEN/OFF banks switch on the interface main timing module allows selection of odd half banks, even half banks, or full banks addressing mode. When the switch is set to ODD, only odd memory banks (1, 3, 5, and 7) are selected. The EVEN position of the switch selects only even memory banks (0, 2, 4, and 6). The OFF position of the switch selects full banks (0 through 7). The software usually reserves page 0 for header information when full banks mode is selected. (Any page may be wired for the header information but the software must correspond.) Any one of the even pages (0, 2, 4, etc.) may be reserved for header information when even banks mode is selected or any one of the odd pages (1, 3, 5, etc.) when odd banks mode is selected. Figures 2-30 and 2-31 show the formats for full banks and half banks addressing.

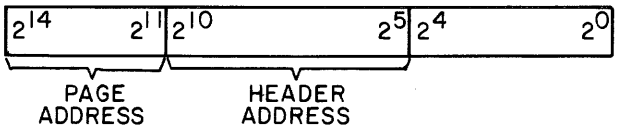


Figure 2-30. Full Banks Address Format, 844 Interface

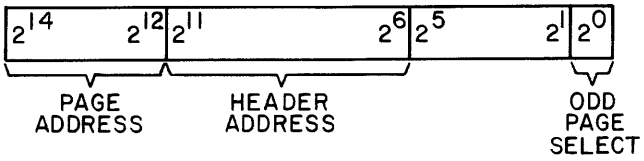


Figure 2-31. Half Banks Address Format, 844 Interface

The software places the starting address into the address register of the interface via the normal channels. Only page (whole sector) transfers take place between the interface and unit so that terminating addresses are not required. Bits 2^1 through 2^4 are hardwired to ground in the core control address register, and therefore, are always 0's. This condition restricts the starting address of the header to 32-word boundaries. During full banks mode, bit position 2^0 is forced clear, so that the boundaries would be 0000_{16}, 0020_{16}, 0040_{16}, etc.

When either odd or even half banks mode is selected by the ODD/EVEN/OFF banks switch, bit 2^0 is no longer used as part of the 32-word header address boundary. A 0 must be placed in bit position 2^5 to maintain the 32-word boundary. This shifts the header address one bit to the left, leaving only three bits for the page address. These three page address bits select one of eight odd pages (1, 3, 5, etc.) if a 1 is placed in bit position 2^0 or one of eight even pages (0, 2, 4, etc.) if a 0 is placed in bit position 2^0.

PARITY

A parity check is made by the interface on all data that is read from SBU memory. If the parity check fails, read error and transfer error status bits are sent to the SCU.

844 INTERFACE NORMAL CHANNEL SIGNALS

Tables 2-22 and 2-23 describe the normal input and output channel bits used with the 844 interface. Bit assignments in the table are for the 844 storage station. Interface bit assignments may be different in other stations. Refer to appendix A for a complete list of normal channel bit assignments for all stations.

The interface must remain connected to the SCU in order to receive control information from the SCU. The interface remains connected as long as the SCU holds the unique connect code, assigned to the interface, on output channel 6, bits C through F.

TABLE 2-22. NIC-6 AND -7 BIT DESCRIPTIONS FOR 844 INTERFACE

Channel	Bit	Name	Function
NIC-6	$0(2^{15})$	Unit busy	This status bit from the disk unit indicates that the unit selected by one of select unit bits (NOC-7) is reserved by the other 844 interface.
NIC-6	$1(2^{14})$ ↓ $9(2^{6})$	Units 1 through 9 selected	Each of these status bits from the units indicates that the unit selected by one of select unit bits (NOC-7) is available and reserved.
NIC-6	$A(2^{5})$	Read error	This is a status bit from the interface that results from a parity error or a BCC error. The parity error may result on either the lower or upper half of a 16-bit word during a read operation from SBU memory. The BCC error results when the 16-bit BCC word read from the disk at the end of a page does not match the BCC word generated when the data page is read from the disk. The read error bit also sets the transfer error bit (NIC-7).
NIC-6	$B(2^{4})$	Tag BCC error	This interface status bit is used to indicate an address check code error when set in conjunction with the transfer error bit (NIC-7). The address check code error results when the 16-bit address check word read from the disk after the address tag word does not match the address check word generated when the address tag word is read from the disk.
NIC-6	$C(2^{3})$	Timing chain	This interface status bit indicates that it is too late to start a data transfer for the next consecutive disk sector. A retry should be attempted after a revolution of the disk. If a read error bit (NIC-6) occurs in conjunction with the timing chain bit, it indicates a parity error on the address tag word read from SBU memory.

TABLE 2-22. NIC-6 AND -7 BIT DESCRIPTIONS FOR 844 INTERFACE (Cont'd)

Channel	Bit	Name	Function
			If a read error bit (NIC-6) occurs without the timing chain bit, it indicates a BCC error on the data page read from the disk.
NIC-6	D(2^2) ↓ F(2^0)	SBU bank count 2^2 through 2^0	These three status bits give the not code of one of the eight SBU memory banks that was being accessed during a data transfer when a read error bit (NIC-6) was set. The memory bank with the error is found by complementing the bank count status code and counting back four memory banks.
NIC-7	0(2^{15})	Finished	This interface status bit indicates that the data transfer terminated normally.
NIC-7	1(2^{14}) ↓ 9(2^6)	Units 1 through 9 on sector	Each of these status bits from the selected unit indicates that the addressed sector is one sector away from the heads. The signal is one sector in length.
NIC-7	A(2^5)	Ready	This status bit from the disk unit indicates that the selected unit has completed its first seek. The unit does not accept seek commands until ready.
NIC-7	B(2^4)	Transfer error	This interface status bit sets as a result of a read error (NIC-6), an address check code error (refer to tag BCC error, NIC-6), or an address tag compare error. The address tag compare error results when the address tag word read from SBU memory does not match the address tag word read from the disk.
NIC-7	C(2^3)	On cylinder	This status bit from the disk unit indicates that the selected unit has positioned the read/write heads at the addressed cylinder.

TABLE 2-22. NIC-6 AND -7 BIT DESCRIPTIONS FOR 844 INTERFACE (Cont'd)

Channel	Bit	Name	Function
NIC-7	$D(2^2)$	Seek error	This status bit from the disk unit indicates that the selected unit was unable to complete a move within 500 milliseconds or that the carriage has moved to a position outside the recording field.
NIC-7	$E(2^1)$	Pack unsafe	This status bit from the disk unit indicates that the selected unit has a fault condition.
NIC-7	$F(2^0)$	Full	This interface status bit indicates that new SBU memory address bits (NOC-7) and full bit (NOC-7) have been loaded into the interface address register. The software loads another address when the full status drops. It also indicates that a data page transfer has not yet begun. This marks a point of reference for determining whether an error occurred during address tag or data page transfer.

TABLE 2-23. NOC-6 AND -7 BIT DESCRIPTIONS FOR 844 INTERFACE

Channel	Bit	Name	Function
NOC-6	$0(2^{15})$ ↓ $6(2^{9})$	Not used	
NOC-6	$7(2^{8})$	Load SBU address	This bit is used to gate the SBU memory starting address bits (NOC-7), write tag bit (NOC-7), write bit (NOC-7), full bit (NOC-7), and odd bit (NOC-7) into the interface address register.
NOC-6	$8(2^{7})$	Release	This bit in conjunction with function bit (NOC-6) enables a release signal to all units to clear the reserve status of the selected unit or units.
NOC-6	$9(2^{6})$	Function	This bit is used to gate the address and control bits 0 through 8 (NOC-7), difference select bit (NOC-7), clear fault bit (NOC-7), control select bit (NOC-7), sector select bit (NOC-7), and cylinder select bit (NOC-7) through the interface to the units.
NOC-6	$A(2^{5})$	Begin transfer	This bit initiates a read or write operation in the interface.
NOC-6	$B(2^{4})$	Not used	
NOC-6	$C(2^{3})$ ↓ $F(2^{0})$	Connect code 2^{3} through 2^{0}	These bits connect either 844-A or 844-B interface to the SCU. (Refer to Table 2-1 for 844 interface connect codes.)
NOC-7	$0(2^{15})$	Not used	
NOC-7	$1(2^{14})$	Select unit 1 †	This bit or one of select unit 2 through 9 bits (NOC-7) is used to select one of the nine units. The bit must be cleared to deselect the unit. Only one select unit bit should be set; otherwise, more than one unit would be selected.

See footnotes at end of table.

TABLE 2-23. NOC-6 AND -7 BIT DESCRIPTIONS FOR 844 INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-7	$2(2^{13})$	Select unit 2 †	Refer to select unit 1 bit (NOC-7).
		or	
		Difference select † †	This bit is sent to all units and indicates that the address and control bits 0 through 8 (NOC-7) contain positioning information (difference between unit's present cylinder address and new interface address) for the selected unit.
		or	
		Write tag † † †	This bit enables the interface main timing to write address tags on the disk. The address tags include head gap, sync pattern, address tag word, and address check code. The write bit (NOC-7) must also be set for a write tags operation.
NOC-7	$3(2^{12})$	Select unit 3 †	Refer to select unit 1 bit (NOC-7).
		or	
		Clear fault † †	This bit is sent to all units and clears the fault condition in the selected unit.
		or	
		Write † † †	This set bit signals the interface to write an SBU memory data page onto the disk. A clear bit enables the interface to read a data page from the disk.
NOC-7	$4(2^{11})$	Select unit 4 †	Refer to select unit 1 bit (NOC-7).
		or	
		Control select † †	This bit is sent to all units and indicates that the address and control bits 0 through 8 (NOC-7) contain control information.
		or	
		Full † † †	This bit is sent to the interface address register along with the SBU memory address bits 2^5 through 2^{14} (NOC-7). It enables a full status in the interface. The full status clears in the interface when the page address is sent to the SBU memory.

See footnotes at end of table.

TABLE 2-23. NOC-6 AND -7 BIT DESCRIPTIONS FOR 844 INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-7	5(2^{10})	Select unit 5 †	Refer to select unit 1 bit (NOC-7).
		or	
		Sector select † †	This bit is sent to all units and indicates that the address and control bits 0 through 8 (NOC-7) contain the address of the sector which generates the next on-sector signal. Only sectors 0, 8, and 16 are selected in order to divide the disk in three parts for full page data recording.
		or	
		Odd † † †	This bit is used to select odd data pages (1, 3, 5, etc.) when set or even data pages (0, 2, 4, etc.) when clear. This function is applicable only during half banks mode (ODD/EVEN/OFF banks switch on interface main timing module set to either ODD or EVEN). NOTE The switch selects odd or even banks while the odd bit selects odd or even data pages.
NOC-7	6(2^{9})	Select unit 6 †	Refer to select unit 1 bit (NOC-7).
		or	
		Cylinder select † †	This bit is sent to all units and indicates that the address and control bits 0 through 8 (NOC-7) contain the new cylinder address.
		or	
		SBU memory address bit 2^{14} † † †	This bit and SBU memory address bits 2^{13} through 2^{5} specify the starting address in SBU memory for a page of data and its accompanying header information.

See footnotes at end of table.

TABLE 2-23. NOC-6 AND -7 BIT DESCRIPTIONS FOR 844 INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-7	$7(2^8)$	Select unit 7†	Refer to select unit 1 bit (NOC-7).
		or	
		Address and control bit 8††	This bit and address and control bits 7 through 0 are sent to all units. Information for these bits is given in Table 2-24 and is determined by the accompanying difference select bit (NOC-7), control select bit (NOC-7), sector select bit (NOC-7), or cylinder select bit (NOC-7).
		or	
		SBU memory address bit 2^{13}†††	Refer to SBU memory address bit 2^{14} (NOC-7).
NOC-7	$8(2^7)$ ↓ $9(2^6)$	Select units 8 and 9†	Refer to select unit 1 bit (NOC-7).
		or	
		Address and control bits 7 and 6††	Refer to address and control bit 8 (NOC-7).
		or	
		SBU memory address bits 2^{12} and 2^{11}†††	Refer to SBU memory address bit 2^{14} (NOC-7).
NOC-7	$A(2^5)$ ↓ $F(2^0)$	Address and control bits 5 through 0††	Refer to address and control bit 8 (NOC-7).
		or	
		SBU memory address bits 2^5 through 2^0†††	Refer to SBU memory address bit 2^{14} (NOC-7).

†This bit is set, provided the load SBU address bit (NOC-6) and function bit (NOC-6) are set.

††This bit is set, provided the load SBU address bit (NOC-6) is not set and function bit (NOC-6) is set.

†††This bit is set, provided the load SBU address bit (NOC-6) is set.

ADDRESS AND CONTROL BUS

The address and control bus in the interface consists of nine lines which transmit normal channel information to the selected unit. Information on these lines is determined by an accompanying normal channel tag line signal. Table 2-24 gives the relationship of the address and control bus and the tag line signals.

TABLE 2-24. ADDRESS AND CONTROL BUS FUNCTIONS

Tag Line				
Address/ Control Bus	Difference Select or Cylinder Select	Sector Select	Control Select	
Bit 0 (NOC-7, bit F)	1	1	Write Gate	A 1 input on this line enables the write drivers.
Bit 1 (NOC-7, bit E)	2	2	Read gate	A 1 input on this line enables the digital read data lines.
Bit 2 (NOC-7, bit D)	4	4	Seek forward†	A 1 input on this line initiates forward carriage movement.
Bit 3 (NOC-7, bit C)	8	8	Not used	
Bit 4 (NOC-7, bit B)	16	16	Erase gate	A 1 input on this line enables the erase driver to pass current through the head erase coil.
Bit 5 (NOC-7, bit A)	32	Not used	Seek reverse††	A 1 input on this line initiates reverse carriage movement.
Bit 6 (NOC-7, bit 9)	64	Not used	Return to zero	A 1 input on this line initiates carriage movement to cylinder 00.
Bit 7 (NOC-7, bit 8)	128	Not used	Data strobe early	A 1 input on this line moves the data strobe to the early margin position.
Bit 8 (NOC-7, bit 7)	256	Not used	Data strobe late	A 1 input on this line moves the data strobe to the late margin position.

†If this bit is set when the difference of the cylinder address is 0, a forward offset occurs (heads move toward spindle by 400 microinches).
††If this bit is set when the difference of the cylinder address is 0, a reverse offset occurs (heads move away from spindle by 400 microinches).

PROGRAMMING SEQUENCES

The following procedures are to be used to aid the programmer during the preparation of the program for conducting data transfers between the SBU memory and disk unit via the 844 interface. Normal channel bits are those assigned to the 844 storage station.

CONNECTION PROCEDURE

1. Place the appropriate connect code in bits C through F of NOC-6 to select 844-A or 844-B interface. (Refer to Table 2-1 for proper code.)
2. Set appropriate select unit bit (NOC-7, bits 1 through 9) to connect desired unit to interface.
3. Set load SBU address bit (NOC-6, bit 7).
4. Set function bit (NOC-6, bit 9).
5. Clear function bit (NOC-6, bit 9).
6. Clear select unit bit (NOC-7, bits 1 through 9).
7. Clear load SBU address bit (NOC-6, bit 7).

CYLINDER POSITIONING

1. Set appropriate address and control bits (NOC-7, bits 7 through F) to select new cylinder address.
2. Set function bit (NOC-6, bit 9).
3. Set and clear cylinder select bit (NOC-7, bit 6).
4. Calculate difference between present cylinder address and new cylinder address. Set this difference in address and control bits (NOC-7, bits 7 through F).
5. Set and clear difference select bit (NOC-7, bit 2).
6. Clear address and control bits (NOC-7, bits 7 through F).
7. Set seek reverse or seek forward bit (NOC-7, bit A or bit D).
8. Set and clear control select bit (NOC-7, bit 4).
9. Clear seek reverse or seek forward bit (NOC-7, bit A or bit D).
10. Clear function bit (NOC-6, bit 9).

SECTOR SELECTION

1. Repeat steps in Connection Procedure.
2. Set sector address in address and control bits (NOC-7, bits 7 through F).
3. Set function bit (NOC-6, bit 9) and clear load SBU address bit (NOC-6, bit 7).
4. Set and clear sector select bit (NOC-7, bit 5).
5. Clear function bit (NOC-6, bit 9).
6. Clear address and control bits (NOC-7, bits 7 through F).

DATA TRANSFER PREPARATION

1. Repeat steps in Connection Procedure.
2. Write sync pattern and preamble in 844 table entry.
3. Calculate address tag word and address check code and write it in 844 table entry.
4. Check status of on-sector bit (NIC-7, bits 1 through 9).
5. Wait 75 microseconds to ensure proper operation of interface timing chain after detecting on-sector bit (on-sector status is received one sector early).
6. Clear function bit (NOC-6, bit 9).
7. Prepare to load starting address of SBU memory by setting or clearing write bit (NOC-7, bit 3, 1 = write, 0 = read), setting memory address bits (NOC-7, bits 6 through F) as required, setting or clearing odd bit (NOC-7, bit 5) if selecting odd or even pages (1 = odd, 0 = even), and setting full bit (NOC-7, bit 4).
8. Set and clear load SBU address bit (NOC-6, bit 7).
9. Check status of timing chain bit (NIC-6, bit C). If set, do not proceed with step 1 of Data Transfer. Wait one revolution of the disk, and repeat steps 4 through 8.

1. Set and clear begin transfer bit (NOC-6, bit A).
2. Check status of full bit (NIC-7, bit F). When clear, the data transfer is in process.
3. Check status of finished bit (NIC-7, bit 0). When set, the data transfer of the first sector is complete.
4. Write contiguous sectors in a cylinder by setting and clearing begin transfer bit (NOC-6, bit A) each time finished bit (NIC-7, bit 0) sets.

MULTIPLE DISK DRIVE (MDD) INTERFACE

The MDD interface controls data transfers between 841 MDD disk packs (units) and the SBU memory. Each MDD interface controls up to eight units but can transfer data to only one unit at a time. The SBU usually contains two MDD interfaces. Each interface is assigned a different connect code for selection by the SCU. Both interfaces connect to the same nine 841 MDD units (eight units on line, one standby). The two interfaces are completely independent and can transfer data simultaneously. However, the simultaneous data transfer must be to or from different units. Figure 2-32 shows the relationship between the MDD interface and the system elements with which it communicates.

Operation of the MDD interface is controlled by the SCU through bits on normal channels 5, 6, and 7. Once the SCU has initiated a data transfer operation, it need not remain connected to the interface except to check status.

Normally, the interface completes the operation without further SCU control. Principal control responsibilities of the SCU in relation to the MDD interface are as follows:

Issues connect code for interface.
Issues starting address in SBU memory for data page transfer.
Selects MDD unit, cylinder, and sector of the unit.
Issues read, write, and write tags commands to initiate data transfer operations.
Monitors status conditions.

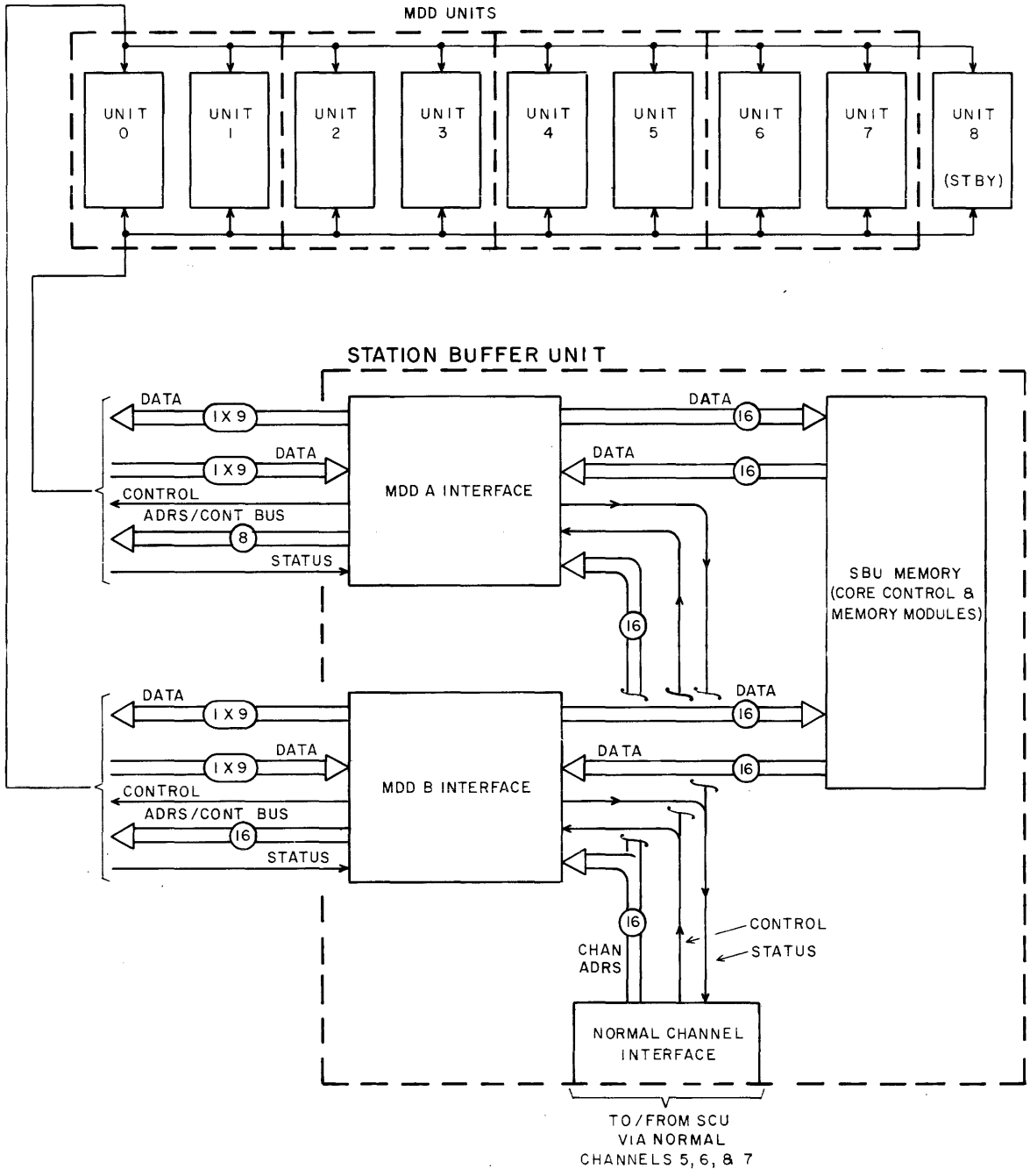


Figure 2-32. MDD Interface System Relationship

MDD DESCRIPTION

The 841 MDD consists of a cabinet that contains one or two MDD units, associated circuitry, and drive mechanism. Up to eight units are used on line with the MDD interface while the ninth unit is on standby.

Each unit contains 11 disks mounted on a common vertical spindle. The access mechanism for each unit consists of 20 arms mounted on a movable carriage. A read/write head is mounted on the end of each arm. Each of the 20 arms contacts a disk surface. Recording is not done on the top side of the upper disk or the bottom side of the lower disk.

An initial seek operation is performed by loading a unit, closing the drawer, and pressing the START switch. The carriage then moves horizontally from a retracted position and positions the heads at the outermost track at the edge of each disk. After a delay to permit purging of air, the carriage moves the heads to the innermost track and then withdraws the heads to the outermost track. The heads are now in a recording attitude near the surfaces of the disks.

The 20 heads are aligned in a vertical line to enable access to corresponding circular tracks on the 20 disk surfaces. The 20 aligned circular tracks are called a cylinder. The access mechanism can be stopped at any one of the 180 track positions (modified from 200). This provides 180 concentric cylinders of data or 3600 data tracks for each unit. The 20 tracks on a cylinder and associated read/write heads are numbered 0 to 19 from top to bottom. The 180 concentric cylinders are numbered 000 to 179 from outside ring to inside ring. The address of an individual track in a given unit consists of the cylinder number and read/write head number.

Data is read or written consecutively from track 0 to track 19 on each cylinder. Electronic switching from one read/write head to the next provides minimum selection time.

The units are modified to rotate at 2184 rpm while maintaining a recording frequency of 2.52 MHz for STAR station operation.

MDD SECTOR RECORDING

GENERAL

Each disk surface of an MDD unit is divided into 14 equal-sized sectors by 14 sector marks numbered 0 through 13. The MDD unit generates a sector mark signal each time a mark is sensed on the disk. The MDD interface ignores all but sector marks 0 and 7, dividing the disk in half for the data page recording scheme. The sector is the smallest addressable unit on the disk. All read and write operations must start at the beginning of a sector. The interface transmits one page of data and its associated address tagging information to the MDD unit during each half revolution of the disk. A data page is equal to 2048 16-bit words.

The MDD unit records information serially bit by bit on the disk. The format for sector recording is shown in Figure 2-33.

The head gap, sync pattern, address tag word, and address check code are written on all sectors when a disk pack is first put into use. This is a write tags operation and is normally a one-time operation for each disk pack.

The second head gap, second sync pattern, preamble, data page, block check code, and end of record are the only areas on a sector that can be altered by a normal write operation.

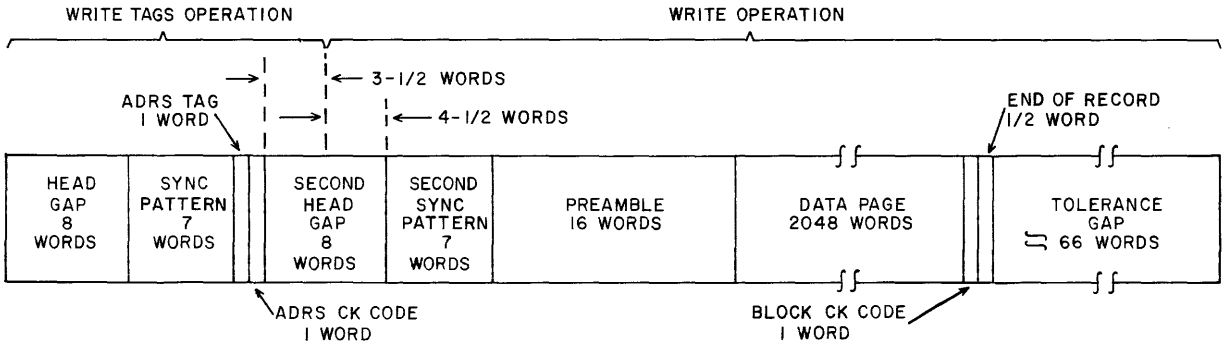


Figure 2-33. MDD Sector Recording

HEAD GAP

The head gap is a block of all 1's which is equivalent to the physical distance of the read/write to erase gap of the MDD unit.

SYNC PATTERN

Each sync pattern consists of all 0's with a single 1 bit at the end. The 1 bit (sync bit) provides a known starting point in the bit stream that signals the beginning of useful data. A search for the sync bit takes place prior to an MDD read or write operation.

ADDRESS TAG WORD

The address tag word designates the area on the disk to be accessed. Each sector contains a unique address tag word that distinguishes it from all other sectors within a disk pack. Figure 2-34 shows a detailed breakdown of the address tag word. Bit 2^0 indicates that the recording surface under the selected head is marginal and should not be used. Bit 2^1 indicates that the recorded data in the sector should not be destroyed. Bit 2^2 designates sector 0 on the disk when it is a 0 or sector 7 when it is a 1. This is because the disk is divided in half for data page recording. Bits 2^3 through 2^7 designate one of the 20 read/write heads numbered 0 through 19. Bits 2^8 through 2^{15} designate one of 180 cylinders numbered 0 through 179.

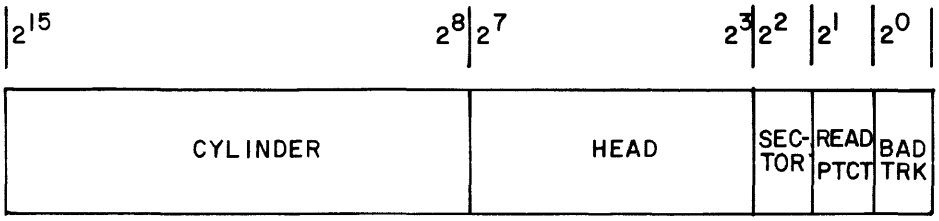


Figure 2-34. MDD Address Tag Word

ADDRESS CHECK CODE

The address check code is a 16-bit word for detecting errors in the transmission of the address tags. During an MDD write tags operation, the software generates a 16-bit check code that is unique to the address tag word. This check code is written on the disk immediately after the address tag word. Previous to a read or write operation, the address tag information is read from the disk into the interface. The information passes through the coder leaving a code in the coder. Now as the address check code is read from the disk, it should clear the coder. If not, MDD error and tag BCC error status bits are sent to the SCU.

PREAMBLE

The 16-word preamble provides software identification information of the data page that follows.

DATA PAGE

The data page contains 2048 16-bit words.

BLOCK CHECK CODE

The BCC is a 16-bit word for detecting errors in the transmission of the preamble and data page. During an MDD write operation, the interface coder creates a BCC that is unique to the preamble and data page. The BCC is written on the disk after the data page. During a read operation of the data page, the coder creates another BCC which is compared with the recorded BCC on the disk. If the two BCCs do not compare, MDD error and read error status bits are sent to the SCU.

END OF RECORD

The end of record is eight zero bits written after the BCC before the write operation terminates.

TOLERANCE GAP

The tolerance gap is an unrecorded area on the disk that separates the BCC and the next sector mark.

DATA TRANSFER OPERATIONS

GENERAL

The SCU initiates and selects all operations through the normal channels. The types of transfer operations performed by the MDD interface include MDD write, MDD read, and write tags. The following initial procedure is always performed prior to any transfer operation.

INITIAL PROCEDURE

Upon detecting a transfer request, the SCU software performs the following tasks.

Connects the appropriate MDD interface. Each of the two interfaces has its own connect code.

Selects a particular MDD unit.

Positions the read/write head assembly to the proper cylinder (cylinder select).

Selects the proper sector on the disk and waits for receipt of the on-sector status from the unit.

Places header information including address tag word into page 0 of SBU memory. Formats differ between write tags operation and write/read operation and are given in the paragraphs for these operations. The tag word is identical to the tag word written at the beginning of the sector. It is used by the interface for head selection.

Loads starting address for SBU memory into MDD interface, sends full control bit, and selects write tags, write, or read operation at the same time. Interface should return a full status.

Provides a begin transfer command to the interface.

The MDD interface then reads the first word (address tag word) from memory. The head address portion of the address tag word is transmitted through the interface to the MDD unit. The unit then selects the addressed head. The interface now waits for the next sector mark signal from the unit.

Detection of the sector mark signal alerts the interface that address tag information (head gap, sync pattern, and address tag word) is being read from the sector. The interface now does a search for the 1 bit at the end of the sync pattern. Detection of the sync bit enables the interface to accept useful data from the disk.

After the sync bit, the address tag word is read from the sector into the interface. The interface does an address tag word compare against the address tag previously read from SBU memory. If the compare fails, an MDD error status bit is sent to the SCU.

Following the address tag word, the interface reads the address check code from the disk and performs a check. If the check fails, MDD error and tag BCC status bits are sent to the SCU. If the check passes, the interface drops the full status.

WRITE TAGS OPERATION

The write tags operation is a one-time operation used to write address tag information at the beginning of all sectors in a disk pack (refer to Figure 2-33 for the sector recording format). These tags are read as addresses during read and write operations.

The cylinder and head portions of the address tag word correspond to the actual cylinder and head numbers of the disk. Sectors 0 and 1 of the sector portion of the address tag word correspond to sectors 0 and 7 of the disk.

Prior to selecting a write tags operation on the normal channels at the time the starting address is loaded, the SCU software uses the SCU coupler to place the address tags to be written on the disk into a 32-word section in page 0 of SBU memory. Page 0 is always reserved for address tags and does not contain data.

Page 0 is divided into 64 32-word sections. This format is intended to limit addressing to 32-word boundaries. Only 24 words in a section are used for a write tags operation while all 32 words are used in a read or write operation. Therefore, the addresses for the sections are 0000_{16}, 0020_{16}, 0040_{16}, 0060_{16}, 0080_{16}, $00A0_{16}$, etc. Only a few of the sections are used at any one time for address tags. Any of the 64 sections may be selected for the tags. Using the section that begins with address 0020_{16} as an example, the tags for one sector would be located in page 0 as follows:

Word 0020_{16} = address tag word (1 word)

Words 0021_{16} - 0028_{16} = head gap (8 words, all 1's)

Words 0029_{16} - $002F_{16}$ = sync pattern (7 words, all 0's except last bit is a 1)

Word 0030_{16} = address tag word (1 word, repeat of word 0020_{16})

Word 0031_{16} = address check code (1 word)

Words 0032_{16} - 0037_{16} = part of second head gap (6 words, all 1's)

The first address tag word in memory is for head selection as described in Initial Procedure, and is not written on the sector during a write tags operation.

Selection of the write tags operation enables the interface to bypass the address tag read and verify sequence described in Initial Procedure since none exist on the disk.

Following detection of the first sector mark after the on-sector status signal, the head gap, sync pattern, address tag word, address check code, and three and one-half words of the second head gap are written at the beginning of the first sector.

After the tags for the first sector are written on the disk, an end of page status bit is sent to the SCU. The SCU software now selects the next consecutive sector by selecting cylinder, selecting sector, loading starting address, and sending a begin transfer command. This procedure continues until tags are written for all sectors in a pack.

WRITE OPERATION

The SCU software selects a write operation at the time the starting address is loaded as described in Initial Procedure.

Prior to selecting a write operation on the normal channels, the SCU software uses the SCU coupler to place the address tag word, second head gap, second sync pattern, and preamble into page 0 of SBU memory (refer to Figure 2-33 for the sector recording format). As explained in Write Tags Operation, the page is divided into 32-word sections. Using the section that begins with address 0020_{16} as an example, the header information for a sector for a write or read operation would be located in page 0 as follows:

Word 0020_{16} = address tag word (1 word)

Words 0021_{16} - 0028_{16} = second head gap (8 words, all 1's)

Words 0029_{16} - $002F_{16}$ = second sync pattern (7 words, all 0's except last bit is a 1)

Words 0030_{16} - $003F_{16}$ = preamble (16 words)

The address tag word is the one used for head selection and address tag compare, mentioned in Initial Procedure, and is not written on the sector during a write operation.

After the address check code word is read from the sector, as described in Initial Procedure, a write operation enables the remainder of the second head gap (four and one-half words), second sync pattern, and preamble to be written in the sector. Since the first three and one-half words of the second head gap are written during a write tags operation, the interface reads these words from memory but prevents the MDD from writing until this area of the sector has passed.

Once the header information is completely written on the sector, the full status drops and the interface addresses a data page in memory (1 through 15) and writes the entire data page in the sector. The memory page addressed is specified by the starting address loaded during the initialization procedure.

The interface also generates a BCC and writes this after the data page. The unit then writes the eight-bit end of record after the BCC. This completes the write operation and an end of page status bit is sent to the SCU.

The remaining space on the disk before the next sector is an unwritten tolerance gap.

After the software detects the end of page status, it may write the next consecutive sector by sending a begin transfer command (provided header information was placed in SBU memory and a starting address was loaded when the full status dropped).

READ OPERATION

The SCU software selects a read operation at the time the starting address is loaded as described in Initial Procedure.

Prior to selecting a read operation on the normal channels, the SCU software uses the SCU coupler to place the address tag word, second head gap, and second sync pattern into page 0 of SBU memory. Refer to Write Operation for memory location of this header information.

After the address check code word is read from the sector, as described in Initial Procedure, a read operation enables the interface to skip over the second head gap and begin a search for the 1 sync bit in the second sync pattern.

After the sync bit is read, the preamble is read from the disk into page 0 to complete the 32-word section.

Once the header information is completely read from the sector, the full status drops and the interface addresses a data page in memory (1 through 15) and transfers the entire data page from the sector to memory. The memory page addressed is specified by the starting address loaded during the initialization procedure.

Next, the interface does a check when the BCC is read from the sector. If the check fails, a read error status bit is sent to the SCU. If the check passes, an end of page status bit is sent to the SCU.

After the software detects the end of page status, it may read the next consecutive sector by placing the header and sending a begin transfer command (provided header information was placed in SBU memory and a starting address was loaded when the full status bit dropped).

SBU MEMORY ADDRESSING

The SCU software must issue a starting address to the MDD interface at the time a read or write operation is initiated. The starting address specifies the location of the data page and also the location of the 32-word block of header information (address tags and preamble) for the data page. Since the data page is confined to only 2048 words of data, the software must reserve a separate page for the accompanying 32-word block of header information.

The ODD/EVEN/OFF banks switch, on the interface main timing module, allows selection of odd half banks, even half banks, or full banks addressing mode. When the switch is set to ODD, only odd memory banks (1, 3, 5, and 7) are selected. The EVEN position of the switch selects only even memory banks (0, 2, 4, and 6). The OFF position of the switch selects full banks (0 through 7). The software reserves page 0 for header information when even half banks mode or full banks mode is selected. Page 1 is reserved for header information when odd half banks mode is selected. Figures 2-35 and 2-36 show the formats for full banks and half banks addressing.

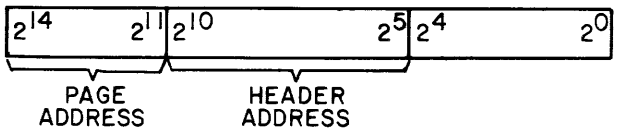


Figure 2-35. Full Banks Address Format, MDD Interface

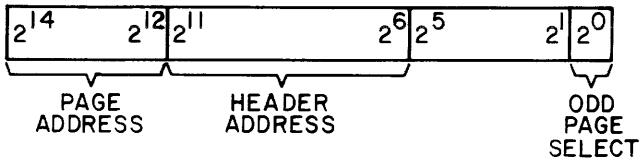


Figure 2-36. Half Banks Address Format, MDD Interface

The software places the starting address into the address register of the interface via the normal channels. Only page (whole sector) transfers take place between the interface and unit so that terminating addresses are not required. Bits 2^1 through 2^4 are hardwired to ground in the interface address register, and therefore, are always 0's. This condition restricts the starting address of the header to 32-word boundaries. During full banks mode, bit position 2^0 is forced clear so that the boundaries are 0000_{16}, 0020_{16}, 0040_{16}, etc.

When either odd or even half banks mode is selected by the ODD/EVEN/OFF banks switch, bit 2^0 is no longer used as part of the 32-word header address boundary. A 0 must be placed in bit position 2^5 to maintain the 32-word boundary. This shifts the header address one bit to the left leaving only three bits for the page address. These three page address bits select one of eight odd pages (1, 3, 5, etc.) if a 1 is placed in bit position 2^0 or one of eight even pages (0, 2, 4, etc.) if a 0 is placed in bit position 2^0.

PARITY

A parity check is made by the interface on all data that is read from SBU memory. If the parity check fails, read error and transfer error status bits are sent to the SCU.

MDD INTERFACE NORMAL CHANNEL SIGNALS

Tables 2-25 and 2-26 describe the normal input and output channel bits used with the MDD interface. Bit assignments in the table are for the STAR-1B service station. MDD interface bit assignments may be different in other stations. Refer to appendix A for a complete list of normal channel bit assignments for all stations.

The MDD interface must remain connected to the SCU in order to receive control information from the SCU. The interface remains connected as long as the SCU holds the unique connect code, assigned to the interface, on output channel 6, bits D, E, and F.

TABLE 2-25. NIC-6 AND -7 BIT DESCRIPTIONS FOR MDD INTERFACE

Channel	Bit	Name	Function
NIC-6	$0(2^{15})$ ↓ $8(2^{7})$	Unit 0 selected through unit 8 selected	Each of these status bits from the MDD units indicates that the unit selected by the unit select and logic no. bits (NOC-5) is available and reserved NOTE When the standby unit replaces another unit on line, it takes the select code of the replaced unit but has its own unit selected status line.
NIC-6	$9(2^{6})$	Full	This interface status bit indicates that new SBU address bits (NOC-5) and full bit (NOC-5) have been loaded into the interface address register. The software loads another address when the full status bit drops. It also indicates that a data page transfer has not yet begun. This marks a point of reference for determining whether an error occurred during address tag or data page transfer.
NIC-6	$A(2^{5})$	Read error	This is a status bit from the interface that results from a parity error or a BCC error. The parity error may result on either the lower or upper half of a 16-bit word during a read operation from SBU memory. The BCC error results when the 16-bit BCC word read from the disk unit at the end of a data page does not match the BCC word generated when the data page is read from the disk. The read error bit also sets the MDD error bit (NIC-7).

TABLE 2-25. NIC-6 AND -7 BIT DESCRIPTIONS FOR MDD INTERFACE (Cont'd)

Channel	Bit	Name	Function
NIC-6	B(2^4)	Tag BCC error	This interface status bit is used to indicate an address check code error when set in conjunction with the MDD error bit (NIC-7). The address check code error results when the 16-bit address check word read from the disk after the address tag word does not match the address check word generated when the address tag word is read from the disk.
NIC-6	C(2^3)	End of page	This interface status bit indicates that the transfer of a data page between the SBU memory and disk unit (in either direction) has been completed and the interface is ready to process the next data page.
NIC-6	D(2^2) ↓ F(2^0)	SBU $\overline{\text{bank count}}$ 2^2 through 2^0	These three status bits give the not code of one of the eight SBU memory banks that was being accessed during a data transfer when a read error bit (NIC-6) was set. The memory bank with the error is found by complementing the bank count status code and counting back four memory banks.
NIC-7	0(2^{15}) ↓ 8(2^7)	Unit 0 through 8 on sector or seek error	Each of these status bits from the MDD units indicates that the unit has completed the seek for the addressed sector (on sector) or is unable to complete the seek for the addressed sector (seek error). The on-sector signal is one sector in length, occurs immediately preceding the addressed sector, and is gated by the on-cylinder bit(NIC-7). The 1 pulse for an on sector occurs once per revolution of the disk pack until cleared.
NIC-7	9(2^6)	Busy	This status bit from the MDD units indicates that the unit selected by unit select and logic number bits (NOC-5) is reserved by the other MDD interface.

TABLE 2-25. NIC-6 AND -7 BIT DESCRIPTIONS FOR MDD INTERFACE (Cont'd)

Channel	Bit	Name	Function
NIC-7	A(2^5)	Ready	This status bit from the MDD units indicates that the unit selected by unit select and logic number bits (NOC-5) meets all conditions to transmit data.
NIC-7	B(2^4)	Seek error	This status bit from the MDD units indicates that the selected unit was unable to complete a seek operation of the addressed sector in 600 milliseconds.
NIC-7	C(2^3)	On cylinder	This status bit from the MDD units indicates that the selected unit has positioned the read/write heads at the addressed cylinder.
NIC-7	D(2^2)	Pack unsafe	This status bit from the MDD units indicates that the selected unit has one or more fault conditions that inhibit write and erase currents to the heads.
NIC-7	E(2^1)	MDD error	This status bit from the interface indicates a transfer error. The transfer error is the result of read error (NIC-6), address check code error (refer to tag BCC error, NIC-6), or address tag compare error. The address tag compare error results when the address tag word read from SBU memory does not match the address tag word read from the disk.
NIC-7	F(2^0)	$\overline{\text{Finished}}$	This status bit from the interface indicates a not finished condition of the data transfer when set.

TABLE 2-26. NOC-5 AND -6 BIT DESCRIPTIONS FOR MDD INTERFACE

Channel	Bit	Name	Function
NOC-5	0(2^{15})	Release †	This bit is sent to all MDD units to release the selected unit by clearing the reserve status of the unit.
		or	
		Write tags † †	This bit enables the main timing in the interface to write address tags on the disk. The address words include head gap, sync pattern, address tag word, and address check code. The write bit (NOC-5) must also be set for a write tags operation.
NOC-5	1(2^{14})	Sector select †	This bit is sent to all MDD units and indicates that the address and control bits 0 through 7 (NOC-5) contain the address of the sector which generates the next on-sector signal. Only sectors 0 and 7 are selected in order to divide the disk in half for full-page data recording.
		or	
		Full † †	This bit is sent to the interface address register along with page and header address bits (NOC-5). It enables a full status in the interface. The full status clears in the interface when the page address is sent to the SBU memory.
NOC-5	2(2^{13})	Difference select †	This bit is sent to all MDD units and indicates that the address and control bits 0 through 7 (NOC-5) contain positioning information (difference between unit's present cylinder address and new interface address) for the selected unit.
		or	
		Proceed † †	This bit is sent to the interface address register along with page and header address bits (NOC-5). It is used in conjunction with the begin transfer bit (NOC-6) to initiate a read or write operation in the interface.

See footnotes at end of table.

TABLE 2-26. NOC-5 AND -6 BIT DESCRIPTIONS FOR MDD INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-5	$3(2^{12})$	Clear †	This bit is sent to all MDD units and unreserves any unit previously selected by the interface regardless of present status.
		or	
		Odd † †	This bit is used to select odd data pages (1, 3, 5, etc.) when set or even data pages (0, 2, 4, etc.) when clear. This function is applicable only during half banks mode (ODD/EVEN/OFF banks switch, on interface main timing module, set to either ODD or EVEN). NOTE The switch selects odd or even banks while the odd bit select odd or even data pages.
NOC-5	$4(2^{11})$	Logic number 2^2 †	This bit in conjunction with logic number 2^1 and 2^0 bits (NOC-5) provide a unit select code. They are sent to all MDD units along with unit select bit (NOC-5) to select a unit.
		or	
		Page address 2^{14} † †	This bit and page address 2^{13}, 2^{12}, and 2^{11} bits (NOC-5) are part of the starting address and specify 1 of 15 data page addresses in the SBU memory.
NOC-5	$5(2^{10})$	Logic number 2^1 †	Refer to logic number 2^2 (NOC-5).
		or	
		Page address 2^{13} † †	Refer to page address 2^{14} (NOC-5).
NOC-5	$6(2^9)$	Logic number 2^0 †	Refer to logic number 2^2 (NOC-5).
		or	
		Page address 2^{12} † †	Refer to page address 2^{14} (NOC-5).

See footnotes at end of table.

TABLE 2-26. NOC-5 AND -6 BIT DESCRIPTIONS FOR MDD INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-5	7(2^8)	Unit select †	Refer to logic number 2^2 (NOC-5).
		or	
		Page address 2^{11} † †	Refer to page address 2^{14} (NOC-5).
NOC-5	8(2^7)	Address and control bit 7 †	This bit along with address and control bits 6 through 0 (NOC-5) are sent to all MDD units. Information in the address and control bits is given in Table 2-24 and is determined by the accompanying sector select bit (NOC-5), control select bit (NOC-6), or difference select bit (NOC-5).
		or	
		Write † †	This set bit signals the interface to write an SBU memory data page onto the disk. A clear bit enables the interface to read a data page from the disk.
NOC-5	9(2^6)	Address and control bit 6 †	Refer to address and control bit 7 (NOC-5).
NOC-5	A(2^5)	Address and control bit 5 †	Refer to address and control bit 7 (NOC-5).
		or	
		Header address 2^{10} † †	This bit and header address bits 2^9 through 2^5 specify the address in page 0 of the header information that accompanies a data page.
NOC-5	B(2^4) ↓ F(2^0)	Address and control bits 4 through 0 †	Refer to address and control bit 7 (NOC-5).
		or	
		Header addresses 2^9 through 2^5 † †	Refer to header address 2^{10} (NOC-5).
NOC-6	0(2^{15}) ↓ 7(2^8)	Not used	
NOC-6	8(2^7)	Begin transfer	This bit, along with the proceed bit (NOC-5), initiates a read or write operation in the interface.

See footnotes at end of table.

TABLE 2-26. NOC-5 AND -6 BIT DESCRIPTIONS FOR MDD INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-6	$9(2^6)$	Clear end page	This bit is used to set a not end page status in the interface at the beginning of a read or write operation. After the read or write of a data page is completed, an end of page status bit (NIC-6) results.
NOC-6	$A(2^5)$	Load SBU address	This bit is used to gate the SBU memory starting address (page and header address bits, NOC-5). Write tags bit (NOC-5), full bit (NOC-5), proceed bit (NOC-5), odd bit (NOC-5), and write bit (NOC-5) into the interface.
NOC-6	$B(2^4)$	Control select	This bit is sent to all MDD units and indicates that the address and control bits 0 through 7 (NOC-5) contains control information.
NOC-6	$C(2^3)$	Function	This bit is used to gate the address and control bits 0 through 7 (NOC-5), release bit (NOC-5), sector select bit (NOC-5), difference select bit (NOC-5), clear bit (NOC-5), logic number bits (NOC-5), and unit select bit (NOC-7) through the interface to the units.
NOC-6	$D(2^2)$ ↓ $F(2^0)$	Connect code 2^2 through 2^0	These bits connect either MDD-A or MDD-B interface to the SCU (refer to Table 2-1 for MDD interface connect codes).

†This bit is set, provided the function bit (NOC-6) is set.
††This bit is set, provided the load SBU address bit (NOC-6) is set.

ADDRESS AND CONTROL BUS

The address and control bus in the interface consists of eight lines which transmit normal channel information to the selected MDD unit. Information on these lines is determined by an accompanying normal channel tag line signal. Table 2-27 gives the relationship of the address and control bus and the tag line signals.

TABLE 2-27. ADDRESS AND CONTROL BUS FUNCTIONS

Address/ Control Bus	Tag Line			
	Difference Select	Sector Select	Control Select	
Bit 0 (NOC-5, bit F	1	1	Write gate –	A 1 input on this line enables the write drivers.
Bit 1 (NOC-5, bit E)	2	2	Read gate –	A 1 input on this line enables the digital read data lines.
Bit 2 (NOC-5, bit D)	4	4	Seek forward –	A 1 input on this line initiates forward carriage movement.
Bit 3 (NOC-5, bit C)	8	8	Not used	
Bit 4 (NOC-5, bit B)	16	Not used	Erase gate –	A 1 input on this line enables the erase driver to pass current through the head erase coil.
Bit 5 (NOC-5, bit A)	32	Not used	Seek reverse –	A 1 input on this line initiates reverse carriage movement.
Bit 6 (NOC-5, bit 9)	64	Not used	Return to zero –	A 1 input on this line initiates carriage movement to cylinder 00.
Bit 7 (NOC-5, bit 8)	128	Not used	Not used	

PROGRAMMING SEQUENCES

The following procedures are to be used to aid the programmer during the preparation of the program for conducting data transfers between the SBU memory and the MDD via the MDD interface. Normal channel bits are those assigned to STAR-1B service station.

INTERFACE CONNECTION

Place the appropriate connect code in NOC-6, bits D, E, and F to select MDD-A or MDD-B interface (refer to Table 2-1 for proper code).

MDD UNIT SELECTION

1. Set the appropriate logic number 2^2, 2^1, and 2^0 bits (NOC-6, bits 4, 5, and 6) to select one of the units 0 through 7.
2. Set unit select bit (NOC-5, bit 7).
3. Set function bit (NOC-6, bit C).
4. Clear function bit (NOC-6, bit C).
5. Clear unit select bit (NOC-5, bit 7).

CYLINDER POSITIONING

1. Calculate difference between present cylinder address and new cylinder address. Set this difference in address and control bits (NOC-5, bits 8 through F).
2. Set function bit (NOC-6, bit C).
3. Set and clear difference select bit (NOC-5, bit 2).
4. Clear address and control bits (NOC-5, bits 8 through F).
5. Set seek reverse or seek forward bit (NOC-5, bit A or bit D).
6. Set and clear control select bit (NOC-6, bit B).
7. Clear seek reverse or seek forward bit (NOC-5, bit A or bit D).
8. Clear function bit (NOC-6, bit C).

SECTOR SELECTION

1. Perform MDD Unit Selection procedure.
2. Set sector address in address and control bits (NOC-5, bits 8 through F).
3. Set function bit (NOC-6, bit C).
4. Set and clear sector select bit (NOC-5, bit 1).
5. Clear function bit (NOC-6, bit C).
6. Clear address and control bits (NOC-5, bits 8 through F).

DATA TRANSFER PREPARATION

1. Perform MDD Unit Selection procedure.
2. Write sync pattern and preamble in 841 table entry.
3. Calculate address tag word and address check code and write it in 841 table entry if writing tags.
4. Check status of on-sector bit (NIC-7, bits 0 through 8).
5. Wait 75 microseconds to ensure proper operation of interface timing chain after detecting on-sector bit (on-sector status is received one sector early).
6. Clear function bit (NOC-6, bit C).
7. Prepare to load starting address of SBU memory by setting or clearing write bit (NOC-5, bit 8, 1 = write, 0 = read), setting page address bits (NOC-5, bits 4 through 7) as required, setting header address bits (NOC-5, bits A through F) as required, setting or clearing odd bit (NOC-5, bit 3) if selecting odd or even pages (1 = odd, 0 = even), and setting full bit (NOC-5, bit 1) and proceed bit (NOC-5, bit 2).
8. Set and clear load SBU address bit (NOC-6, bit A).

DATA TRANSFER

1. Set and clear begin transfer bit (NOC-6, bit 8).
2. Check status of full bit (NIC-6, bit 9). When clear, the data transfer is in process.
3. Check status of end of page bit (NIC-6, bit C). When set, the data transfer of the sector is complete.
4. Write contiguous sectors in a cylinder by setting and clearing begin transfer bit (NOC-6, bit 8) each time end of page bit (NIC-6, bit C) sets.

NETWORK TRUNK INTERFACE

The communications network trunk interface (NTI) is an SBU interface unlike all other SBU interfaces in that it is controlled by a remote terminal via modems (modulator/demodulator) rather than by the BC in the SCU. Figure 2-37 illustrates the relationship of the NTI to the SBU and remote terminal.

The NTI may transmit or receive up to 5 megabits per second to or from a modem, transceiver, or data set, depending upon the timing of the type of device used with the communications line. At the NTI/modem interface, data is in serial format and 16-bit parallel format at the SBU memory interface. The mode of operation in the NTI is governed completely by control information received from the remote terminal. There are no normal channel connections from the NTI to the SCU. This being the case, the information contained in the remaining paragraphs of this interface description explains some general communications procedures used and message sequences between the NTI and a remote terminal.

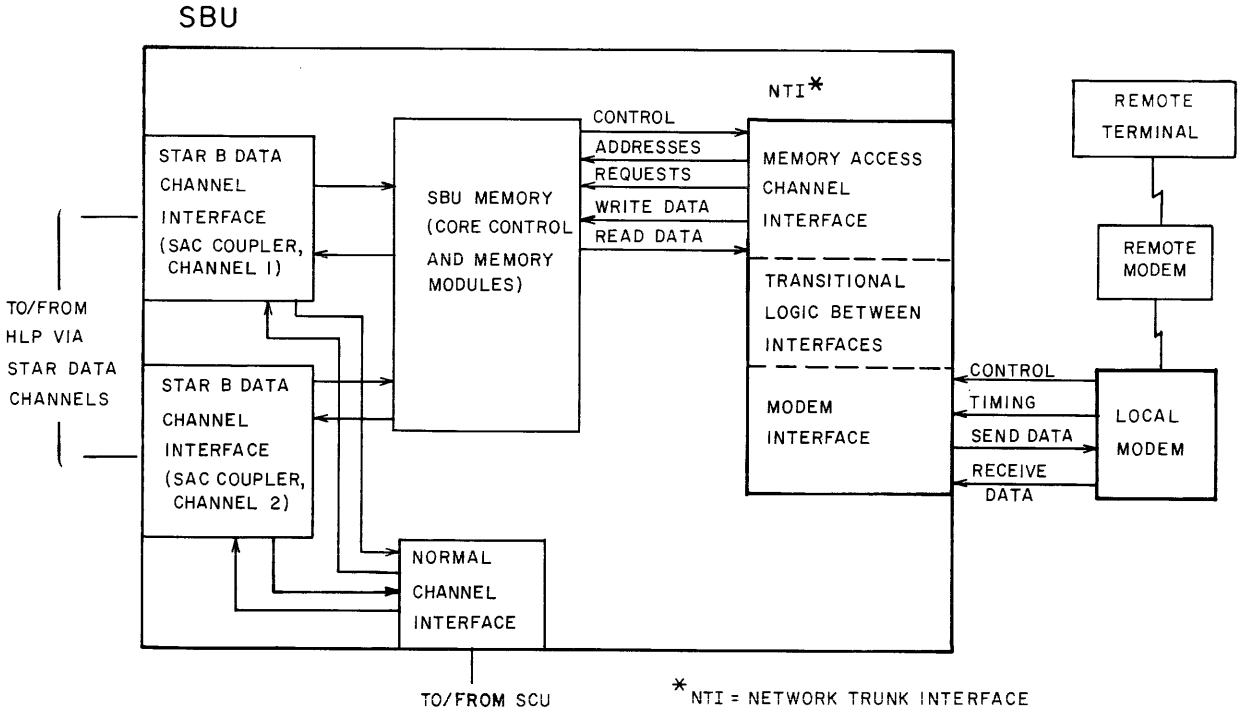


Figure 2-37. Network Trunk Interface System Relationship

ADCCP MESSAGE FORMATS

Message format conforms to USA Standard for Advanced Data Communication Control Procedures (ADCCP) with several exceptions. Serial data transmission from the modem starts with highest order bit first. The all parties address (00_{16}) is not implemented. On a party line, a modem in a terminal desiring to transmit a message after another terminal has been communicating with the NTI must wait between 50 and 300 microseconds after the carrier signal drops before initiating a request to send procedure. This delay allows the NTI time to generate the appropriate response.

In ADCCP procedures, all transmissions are in messages which conform to one of the formats shown in Figure 2-38.

FLAG	ADDRESS	CONTROL FIELD	INFO (DATA) FIELD	CRC†	FLAG

View A: Standard frame format

FLAG	ADDRESS	CONTROL FIELD	CRC†	FLAG

View B: Short frame format (used only when there is no information or data field)

†$\overline{\text{CRC}}$ = cyclic redundancy check

Figure 2-38. ADCCP Message Frame Formats

Explanations of the various blocks of the message frame formats are as follows:

FLAG FIELD

All transfers start and end with a flag sequence. The flag sequence is one 0-bit, six 1-bits, and a 0-bit (0 11 11 11 0) and is used for synchronization purposes. All stations attached to the system data link continuously search for this sequence. The flag sequence is prohibited from occurring in all but the first and last fields of a frame so that transparent operation can be achieved and maintained in all other fields. Transparent operation simply means that information is transmitted exactly as it was received (even if in error). To achieve transparency within a block between starting and ending flags, the transmitting equipment inserts a 0-bit after five 1-bits and the receiving equipment deletes the 0-bit. If seven 1-bits are received, the receiving equipment interprets the condition as an abort and clears itself.

ADDRESS FIELD

Each terminal is assigned a unique address. The address field is eight bits and may contain the addresses of 0 to 255 terminals.

CONTROL FIELD

This is an 8-bit field used for functions, responses, and sequence numbers. The upper four bits contain either the function or response code. The lower four bits contain the sequence number. Function codes and response codes are defined in later paragraphs of this section. Sequence numbers are used to number transfers and to ensure that a transfer is not overlooked or duplicated. The NTI and the remote terminal must maintain the same sequence numbers. At the terminal, the control program generates the sequence number. In the NTI, logic circuits increment the sequence number after each message and store the number in SBU memory. Function and response bits are also stored in SBU memory. The upper eight bits of the memory location contain the terminal address.

INFORMATION (DATA) FIELD

This field is used only when data is to be transmitted. It is used for transmitting a bit stream of transparent data in which no specific character length is implied.

CRC FIELD

The cyclic redundancy check (CRC) field contains a 16-bit character used for data checking. Subsequent paragraphs in this section describe the CRC generator as it is used in the NTI.

NTI MESSAGE SEQUENCES

There are three types of message sequences used with the NTI excluding error conditions: read, write, and reset.

READ SEQUENCE

A read sequence means that the NTI must extract data from a specified location or locations in SBU memory. The procedure is initiated by the remote terminal issuing a four-bit function code of 0101_2 (within the read request message sequence) to the NTI. After processing the read request, the NTI responds to the remote terminal with the read data message.

The order of events in a read request message sequence are given in Table 2-28 and occur as listed. When the sequence number in the read message sequence has been received, the NTI then receives the SBU address information. At the end of the address field, both an even parity word and a CRC word are received from the remote terminal and are used to check all information received except the synchronizing flags. After the CRC character, a final flag character is received to indicate the end of the transmission.

After the NTI processes the read request, it transmits a request to send signal to the local modem and then waits for a carrier on signal from the local modem. After a 50-microsecond delay, the carrier on signal becomes active, the NTI begins transmitting synchronizing flags, and the remote terminal sends the read data message response which follows the sequence listed in Table 2-29.

TABLE 2-28. REMOTE TERMINAL READ REQUEST MESSAGE SEQUENCE

No. of Bits	Name	Definition or Function
8	Flag	Synchronizing character
8	Terminal address	Address of the remote terminal to receive the frame
4	Function	Read
4	Sequence number	Count of the number of transfers (messages) the NTI has transmitted
16	SBU starting address	Any address in SBU memory from which data is extracted
16	Complement of SBU starting address	Ones complement of the SBU starting address
16	Complement of SBU terminating address	Ones complement of the SBU terminating address for the data field in a read data message
16	Even parity check	Checkword generated on the previous 64 bits by the remote terminal and checked in the NTI
16	CRC	Cyclic redundancy check generated by the remote terminal on the previous 80 bits and checked in the NTI (not including inserted zeros)
8	Flag	Synchronizing character used to terminate a message

TABLE 2-29. READ DATA MESSAGE RESPONSE SEQUENCE

No. of Bits	Name	Definition or Function
8	Flag	Synchronizing character
8	Terminal address	Unique address number of the terminal to receive the frame
4	Response	Answer the NTI sends to the terminal requesting data; it can be either an accept with data to follow or a reject
4	Sequence number	Count of the number of transfers (messages) the NTI has processed
16(n)	Data field	Data read from SBU memory and transmitted to remote terminal in 16-bit multiples, two words minimum and up to 2048 words, maximum
16	CRC	Cyclic redundancy checkword generated by the NTI and checked by the remote terminal
8	Flag	Synchronizing character used to terminate the message

WRITE SEQUENCE

A write sequence operation is initiated when a four-bit function code of 1101_2 is sent by the remote terminal to the NTI. The code is transferred to the NTI via the write request message sequence (Table 2-30). After processing the write request, the NTI responds to the remote terminal with the appropriate response sequence (Table 2-31).

After the NTI decodes the write function, it receives SBU address location information and performs an even parity check on all information except the synchronizing flags. The data field follows the parity word and can be any size (in increments of 16-bit words) up to one page (2048 words). The 16-bit CRC character checks all information received from the remote terminal. The remote terminal then turns the carrier off and the NTI then sends an accept after a 50-microsecond delay to the remote terminal, provided no errors were detected in the data received. The accept (or control information) is followed by a 16-bit CRC character for all information transmitted (except synchronizing flags) plus a synchronizing flag to end the transmission.

For each message received from the remote terminal, the NTI must send an appropriate response as to the status of the message. Table 2-31 lists the steps in order for a response sequence.

RESET SEQUENCE

If the remote terminal detects a sequence error either by receiving a reject from the NTI or by analyzing received information, it issues the reset function (0110_2) to the NTI. Table 2-32 lists the steps in order for a reset message sequence.

After the NTI receives and decodes the reset function, it receives SBU address information which is ignored. At the end of the received information, the NTI performs an even parity check and a CRC character check. After the flag is received to indicate the end of transmission, the remote terminal turns the carrier off in the remote modem. The NTI activates request to send, and after a 50-microsecond delay, begins sending synchronizing flags. The NTI then transmits the accept response for the reset function, clears the sequence number, and sends a CRC character and a synchronizing flag to the remote terminal to end the transmission. Format of the acknowledgement message sent to the remote terminal in response to the reset function is listed in Table 2-33.

TABLE 2-30. REMOTE TERMINAL WRITE REQUEST MESSAGE SEQUENCE

No. of Bits	Name	Definition or Function
8	Flag	Synchronizing character
8	Terminal address	Unique address number of the terminal sending the frame.
4	Function	Write
4	Sequence number	A count of the number of transfers (messages) the NTI has processed. This count in the incoming message is compared to the count in SBU memory. If they are not equal, a sequence error is recorded.
16	SBU starting address	Any address in SBU memory where the data field is written.
16	Complement of SBU starting address	Ones complement of the SBU starting address. It verifies that the SBU address register contains the correct address.
16	Complement of SBU terminating address	Ones complement of the SBU terminating address.
16	Even-parity data check	Checkword generated on the previous 64 bits from the remote terminal and checked in the NTI.
16	Data field	Multiples of 16-bit words up to 2048 words in length from the remote terminal.
16	CRC	Cyclic redundancy checkword generated by the remote terminal and checked by the NTI.
8	Flag	Synchronizing character to terminate the incoming message.

TABLE 2-31. NTI RESPONSE SEQUENCE TO WRITE REQUEST MESSAGE

No. of Bits	Name	Definition or Function
8	Flag	Synchronizing character
8	Terminal address	Unique address of the remote terminal receiving the acknowledgement
4	Response	Answer that the NTI sends to the remote terminal sending the data; it can be either an accept or reject.
4	Sequence number	Transfer (message) count of the acknowledgement
16	CRC	Cyclic redundancy checkword generated by the NTI and checked by the remote terminal
8	Flag	Synchronizing character used to terminate the message

TABLE 2-32. REMOTE TERMINAL RESET MESSAGE SEQUENCE

No. of Bits	Name	Definition or Function
8	Flag	Synchronizing character
8	Terminal address	Unique address of the remote terminal sending the frame
4	Function	Reset
4	Sequence number	Count of the number of transfers the NTI has processed
16	SBU starting address	Any address in SBU memory
16	Complemented SBU starting address	Ones complement of the SBU starting address
16	Complemented SBU terminating address	Ones complement of the SBU terminating address
16	Even parity check	Check performed on the previous 64 bits
16	CRC	Cyclic redundancy check performed by the NTI
8	Flag	Synchronizing character used to end the message

TABLE 2-33. NTI RESET ACKNOWLEDGE MESSAGE SEQUENCE

No. of Bits	Name	Definition or Function
8	Flag	Synchronizing character
8	Terminal address	Unique address of remote terminal to receive the frame
4	Response	Answer that the NTI sends to the connected terminal; it may be either an accept or a reject.
4	Sequence number	Reset to zero
16	CRC	Cyclic redundancy check generated by the NTI and checked by the remote terminal
8	Flag	Synchronizing character used to terminate the message

NTI RESPONSES

In response to a function, the NTI can respond with an accept code, an accept code with a data field to follow, or a reject code if an error was detected.

ACCEPT CODE (0001_2)

This response is transmitted to the remote terminal as a four-bit code and indicates that no error conditions were detected during the previous transmission from the remote terminal to the NTI.

ACCEPT WITH DATA FIELD CODE (1001_2)

This response is transmitted to the remote terminal during a read sequence. After the NTI receives the appropriate control information, it transmits this accept response followed by the transmission of serial data.

REJECT CODE (0011_2)

The NTI transmits this response as a four-bit function code after a transmission from the remote terminal resulted in an error condition. The response is sent to the remote terminal as serial data. The seven error conditions which cause a reject are:

Illegal function

Memory parity error on read previous sequence

Sequence compare error

Starting address and complement starting address error

Header parity error

Page boundary reached on write function

Cyclic redundancy error

ERROR CONDITIONS

Although there are many possible error conditions that can cause the NTI to send a reject code or to terminate a transmission, 12 representative error conditions are listed and explained in Table 2-34. Since no status is available, detection of an error is confirmed only by a reject or no response at the terminal. No attempt is made by the hardware to differentiate between the types of errors.

TABLE 2-34. NTI ERROR CONDITIONS

Type of Error	Meaning	Response to Terminal
Illegal function	A function other than read, write, or reset was decoded from the four function bits.	Reject
Memory parity error on read previous sequence	A parity error was detected while reading the sequence number from memory for that terminal.	Reject
Sequence compare error	A lost message is indicated because the sequence number of the message and the sequence number in SBU core were not equal.	Reject
Starting address and complement starting address error	After the starting address is loaded into the SBU starting address register, it is returned to the NTI in ones complement form (return address). This complemented starting address is compared with the return address from the SBU. If they are not equal, an error has occurred.	Reject
Header parity error	This error occurs during an even parity check on the 64 serial data bits following the flag.	Reject
Page boundary reached on write function	This error occurs when a page boundary is reached before the current address equals the terminating address.	Reject
Cyclic redundancy error	This type of error occurs if data or a reject response is transferred incorrectly.	Reject
Carrier drop during address sequence	The address sequence contains the terminal and the SBU addresses both of which reside in the 64 bits following the synchronizing flag. If the carrier from the modem drops during this sequence, an error occurs.	No response, and NTI terminates operation
Insufficient data on write function	If the data stream stops or a flag is sensed before the current SBU address equals the terminating SBU address, an error occurs.	No response
Seven ones detected during an input operation	This error is caused by a malfunction at the transmitting terminal since seven 1's are never to be sent as data. The NTI clears its synchronized condition and timing chain.	No response

TABLE 2-34. NTI ERROR CONDITIONS (Cont'd)

Type of Error	Meaning	Response to Terminal
Memory parity error during read function	If an SBU memory parity error is detected while the NTI is transmitting data to a terminal, transmission is stopped immediately and the CRC character is deleted.	No response, and NTI terminates transmission
Page boundary reached on read function	Data field is limited to one page. If the read data field crosses a page boundary, transmission terminates with a CRC as if the terminating address was reached.	No response, and NTI terminates transmission

CYCLIC REDUNDANCY CHECK

Data checking procedures are based upon the transmission of redundant information with each transmission to enable the receiving party to detect errors. The CRC accumulation starts with the first bit following the flag sequence. All bits except the inserted 0-bit following the five one-bits are included in the accumulation. After the last information bit is received, the contents of the shift register represents the coefficient of the remainder polynomial. The checkword can be shifted out serially and appended to the data field. When data is received, data and the checkword are processed through the CRC generator. If the data is correct, the remainder is zero.

STATION CONTROL UNIT INTERFACES 3

GENERAL

This section describes the functions and programming characteristics of the STAR station control units. Each programmable unit within the SCU is described with the exception of the BC. Since the BC is used in many applications, it is covered in a separate BC reference manual (refer to preface for publication number).

MICRODRUM SUBSYSTEM AND CRT DISPLAY

The microdrum subsystem (Figure 3-1) consists of a rotating magnetic memory (microdrum) supported by the drum and display logic which in turn communicates with the BC via normal input and output channels. Although not a part of the microdrum subsystem as such, the CRT display is inseparable from the microdrum in that the CRT is controlled by the same logic and the microdrum stores the data which is displayed on the screen of the CRT.

Operations of the microdrum are governed by a number of control signals and normal channel bits exchanged with the BC.

ORGANIZATION OF DATA ON MICRODRUM

Data is recorded serially on the microdrum, bit by bit, on one track at a time. Each track is divided into 18 sectors each of which may contain or store 36 or 64 16-bit words per sector. Normally, a sector is the smallest addressable unit of data. Thirty-four tracks (not including a track for autoloading and a track for CRT refreshing) are available for general purpose recording. Therefore, each track may contain 1152 16-bit words except the CRT refreshing track which may contain a maximum of 648 16-bit words. Each sector of the CRT refreshing track is displayed as a separate line on the CRT. Generally, track 2B is used for the memory refreshing track, track 2A is used for device status and other maintenance data, and track 28 usually contains the map of how the function keys are defined. A brief summary of microdrum track assignments is as follows:

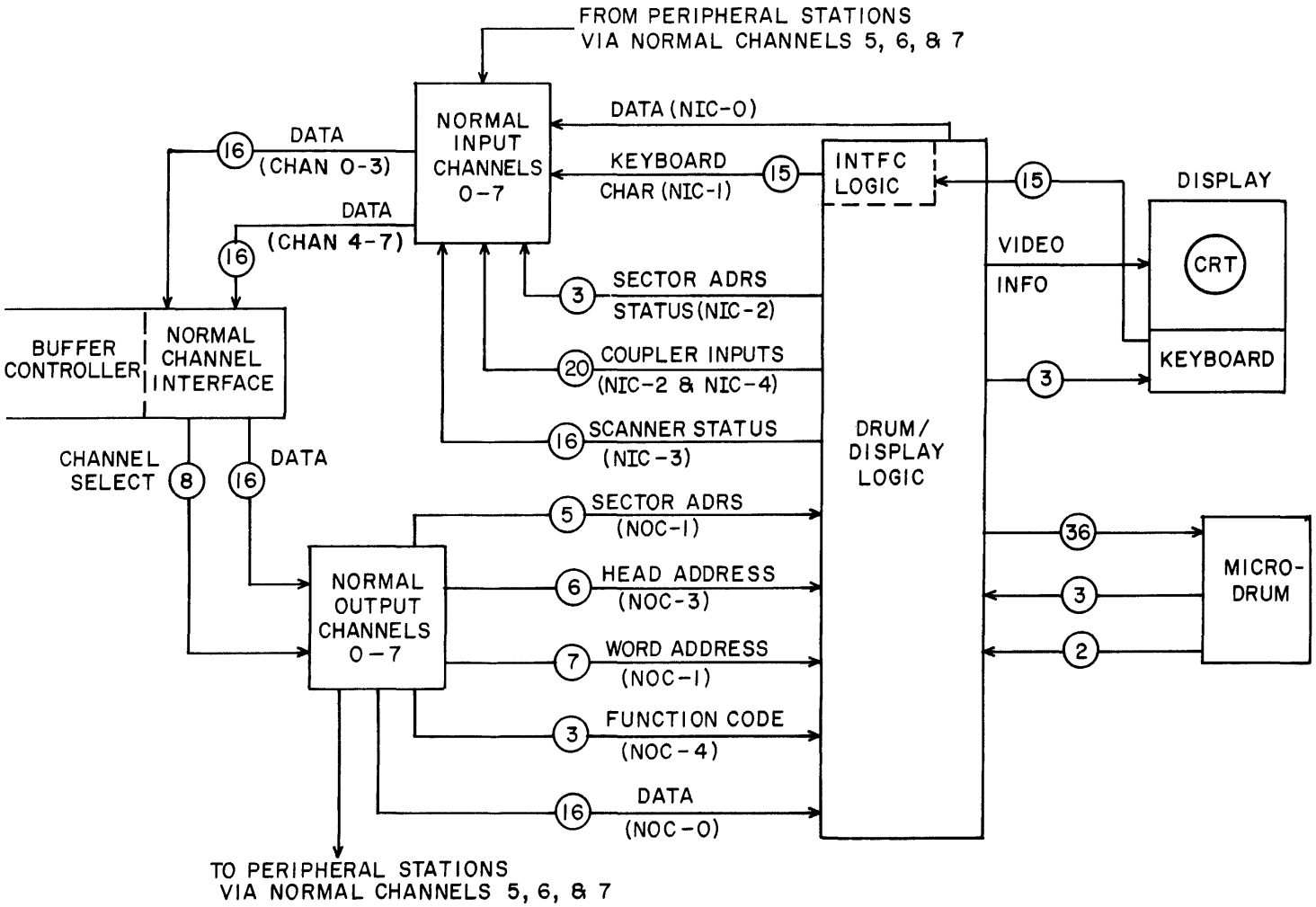


Figure 3-1. Microdrum Subsystem - System Relationship

34 general purpose recording tracks, 1152† words/track	=	39,168
1 CRT refresher track, 648 words/track	=	648
1 autoload program track, 1152 words/track	=	1,152
36 addressable tracks	Total 16-bit words	40,968
2 spare tracks		
2 control tracks		
40 total tracks per microdrum		

Generally, the four principal functions of the microdrum are as follows:

- Provides auxiliary storage for the BC (34 tracks available).
- Provides an autoload track reserved for permanent storage of a program that can be automatically transferred to BC memory.
- Provides for the storage of diagnostic programs.
- Provides a CRT refresher track to enable a continuous display of data on the CRT.

The average access time is 8.4 milliseconds and the maximum access time is 16.7 milliseconds. The transfer rate is 1 megabit per second.

MICRODRUM/BC SIGNALS

The following signals or data bits are exchanged between the normal input and output channels of the BC, the drum and display logic, the microdrum, and the CRT display. Refer to Figure 3-2.

OUTPUT DATA

Sixteen bits of data are sent by the BC over NOC-0 to the drum and display logic and ultimately to the microdrum.

† Based on 18 sectors/track; 64 16-bit words/sector = 1152 words/track.

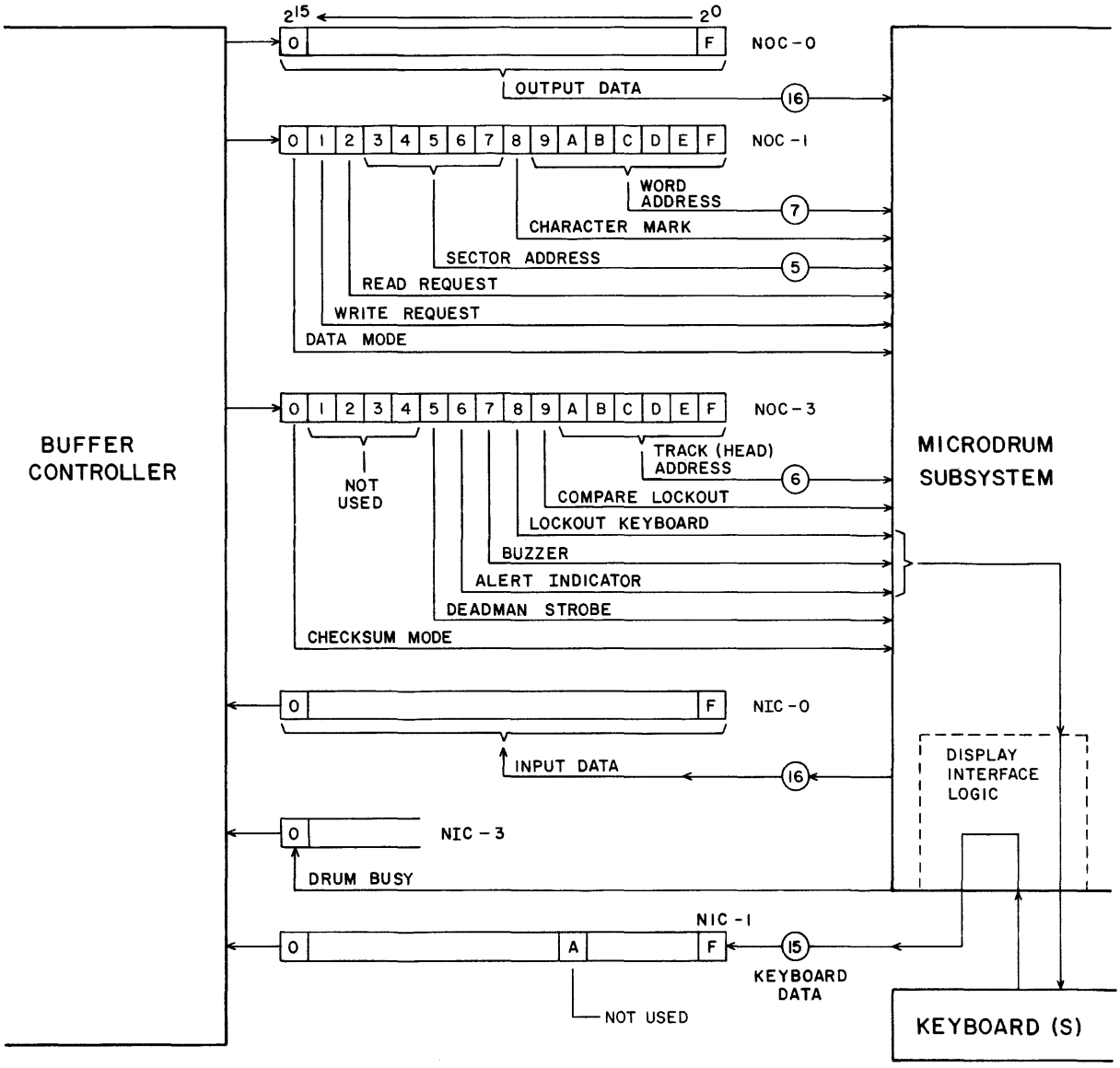


Figure 3-2. Word Formats Between the BC and Microdrum Subsystem and Keyboard(s)

DATA MODE

This signal is sent to the drum and display logic via NOC-1, bit 0 and determines whether the microdrum subsystem will operate in the display mode or data mode. If the BC puts a 1 on this line, the data mode is specified and a 0 indicates the display mode. The two modes of operation are discussed in more detail in subsequent paragraphs of this section.

WRITE REQUEST

This signal (NOC-1, bit 1) is a pulse which is used by the BC to transfer each output word to the microdrum.

READ REQUEST

This signal is a pulse sent to the drum and display logic via NOC-1, bit 2 and informs the microdrum subsystem that the BC is ready to accept a data word from the microdrum.

SECTOR ADDRESS

These five bits (NOC-1, bit 3 through 7) are used by the drum and display logic to select a particular sector on one of the microdrum tracks.

CHARACTER MARK

This signal (NOC-1, bit 8) specifies the placement of the cursor (or entry mark) under either the odd or even character of the word specified by the address lines. The cursor is placed under character 0 (even) if this line is a 0 or under character 1 (odd) if this line is a 1.

WORD ADDRESS

These seven bits (NOC-1, bits 9 through F) indicate the particular word within a sector that will be read from or recorded on the microdrum. This signal is used primarily in the display mode of operation.

CHECKSUM MODE

This signal (NOC-3, bit 0) indicates that a checksum word can be written on the end of a record. It also enables the busy signal to switch states.

DEADMAN STROBE

This signal (NOC-3, bit 5) clears the time-out circuit started by the index mark signal on each drum revolution. If the timeout expires, the BC is stopped, the normal channels are cleared, and the ALERT light and the audible buzzer on the keyboard are activated.

ALERT INDICATOR

This signal (NOC-3, bit 6) sent by the BC, lights the ALERT indicator on the keyboard and indicates that the BC has a message ready for transmission to the display terminal.

BUZZER

This signal (NOC-3, bit 7) sent by the BC to the keyboard, sounds the audible alarm on the keyboard.

LOCKOUT KEYBOARD

This signal (NOC-3, bit 8) originates in the BC or from the keyboard lockout switch on the station control panel and lights the LOCKOUT indicator on the keyboard. It informs the operator that the keyboard may not be used to enter data into the system. The lockout occurs by a combination of logic circuit and program functions and is, therefore, electronic and not a physical disengaging of keys on the keyboard.

COMPARE LOCKOUT

This signal (NOC-3, bit 9) disables the drum address compare circuits allowing continuous reading or writing on the microdrum. The BC must assume complete control during this mode of operation.

TRACK (HEAD) ADDRESS

These six bits (NOC-3, bits A through F) are sent by the BC to select any one of 36 tracks on the microdrum.

INPUT DATA

Sixteen bits of data are collected from the microdrum via NIC-0.

DRUM BUSY

This pulse, sent to the BC via NIC-3, bit 0, indicates that the microdrum is preparing to perform an operation such as read, write, or autoload; it, therefore, cannot accept additional commands until the current operation is completed. While this pulse is active, there must be no changes on any line from the BC to the drum and display logic. The drum busy pulse becomes active each time a new operation is begun.

MICRODRUM OPERATIONS

The following operations can be performed under control of the BC.

WRITE

Records data from the BC into a specified sector or word location on the microdrum. The sector or word location is specified by an address sent from the BC at the beginning of the write operation.

A write lockout feature can be used to inhibit writing on 24 (00_8 through 27_8) of the 36 tracks, if desired. The procedure is initiated by the operator pressing the WRITE LOCKOUT switch on the SCU control panel.

The write operation can be performed in either the data or display mode. In the data mode, bits of information are packed (no spaces between words) on the microdrum and a full sector (64 16-bit words) is the smallest practical unit of data that is recorded.

In the display mode two spaces equal to six characters each form a parenthesis around each word. Therefore, a single 16-bit word can be stored within a sector without disturbing the remainder of the sector. The CRT refresher track is the only track that uses the display mode in a write operation.

Data cannot be recorded on the autoload track from the BC in the normal manner. A special procedure is used to write into this track from the paper tape reader on the portable maintenance console.

During a write operation, data is sent to the microdrum from the A register in the BC. A separate output-from-A instruction is executed to transfer each 16-bit word.

READ

This operation is the transfer of data from any sector or word on the microdrum to BC. In either the data or display mode, a read operation can be specified to begin at any word position within a sector.

During a read operation, input data flows to the A register in the BC. A separate input-to-A instruction is executed for each 16-bit word transferred.

In either a read or write operation, the BC must supply a starting address at the beginning of the operation. Words are then read or written from sequential locations beginning with the starting address supplied by the BC.

AUTOLOAD

The microdrum subsystem provides two methods for autoloading its host station; local or remote. In the local autoload mode, data is read from track 0 on the microdrum and loaded into the first 1152 locations (0000 through 480_{16}) of BC memory. In remote

autoload mode, data is transferred from a higher level processor into the same first 1152 locations of BC memory. In both modes, data is transferred to the BC through the block transfer channel. Selection of the autoload mode of operation is made via LOCAL/REMOTE and AUTOLOAD switches on the SBU and SCU, respectively. The AUTOLOAD MATRIX and HEAD SELECT switches on the SCU control panel may be set so that the autoload data may be loaded into BC memory from tracks 0, 1, 2, 3, 20, 21, 22, or 23.

CRT DISPLAY/KEYBOARD INTERFACE

The drum and display logic provides the interface between the BC and the display/keyboard. Data which is to be displayed is formatted by the logic and written (in display mode) on the microdrum usually from refreshing track 2B. The data is then read back and used to continuously refresh the display. Other tracks not available for general recording are 2A which contains maintenance information and device status and track 28 which contains a map of how function keys are defined. Operator inputs from the keyboard are read into the BC via the normal input channels, written on the CRT refreshing track on the microdrum, and then displayed on the CRT. The following signals are used at this interface.

DISPLAY SIGNALS

There is one line which carries signals from the drum/display logic to the CRT (refer to Figure 3-1).

KEYBOARD SIGNALS

There are 15 lines which carry keyboard data codes and function signals from the keyboard(s) to the BC. These signals pass through the drum/display logic to the BC via NIC-1. The BC uses NOC-3, bits 6, 7, and 8 to send control information to the selected keyboard (refer to Figure 3-3).

DISPLAY OPERATIONS

The BC program must contain the proper instructions to write data on the microdrum or to display data on the CRT. There are two general modes (data and display) of operation available to the users of peripheral subsystems such as those described in section 1 of this manual. The various aspects of each mode are described in the following paragraphs.

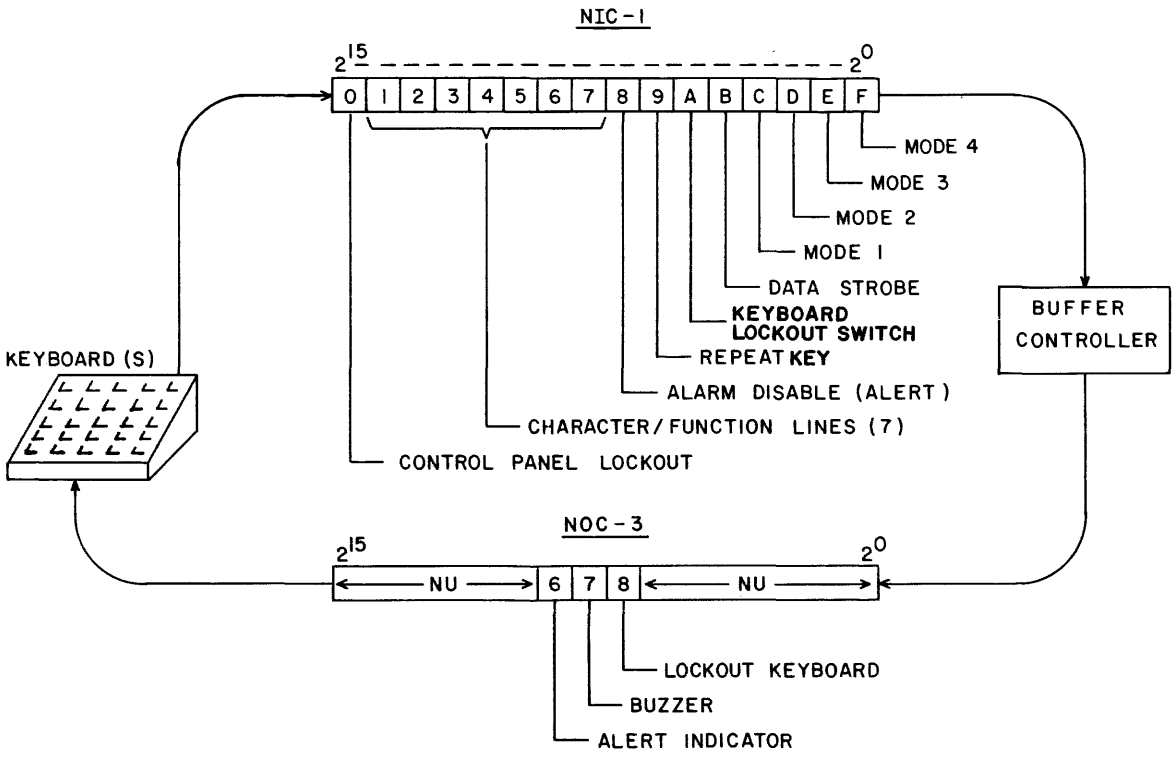


Figure 3-3. Keyboard/BC Word Formats

DATA MODE ADDRESSING

The data mode of operation is initiated by an instruction from the BC which sets the data mode line (NOC-1, bit 0) to a 1. Normally, the smallest segment of data addressed in the data mode is a sector. To accomplish this function, the BC places a head (or track) address, a word address (all zeros), and a sector address on the address lines and then sets the address compare lockout bit to a 1 (NOC-3, bit 9).

In the data mode, 16-bit bytes are stored end-to-end on the drum. That is, there are no spaces left between words. If word addressing is used, the word immediately preceding the addressed word is destroyed. If data is not transferred to the end of the sector, all data previously recorded on the remaining portion of the sector will be of no meaning.

DATA MODE WRITE SEQUENCE

To perform this sequence, the BC places the address information on the address lines, sets the data mode line (NOC-1, bit 0) to a 1, places data on the data lines, and sends a write request (NOC-1, bit 1) to the microdrum. The drum busy status bit (NIC-3, bit 0) then goes to a 1. When the drum busy status goes to 0, the BC may drop its write request, place new data and address information on the normal channel lines, and renew its write request. The write operation is terminated if the BC does not renew its write request within 14 microseconds in data mode, and within 24 microseconds in display mode. If the complete sector is not filled with data, the data writer stays on until sector time and destroys any data previously recorded on that portion of the microdrum.

DATA MODE READ SEQUENCE

The read sequence is identical to the write sequence except that the BC sends a read request (NOC-1, bit 2) to the microdrum instead of a write request for each 16-bit word read in from the microdrum. The drum and display logic then sends the drum busy signal to the BC. When the desired data is on the data lines, the drum busy signal goes to a 0. When the BC has accepted the data, it drops its read request signal to the drum. Therefore, it must reactivate read request and update the address lines in order to collect the next data word from the drum.

If only sector addressing is used, the address compare signal is generated at sector or index time and the bit counter is preset to 14_8. Also, the drum busy signal is not dropped until the current operation is complete.

DISPLAY MODE ADDRESSING

The display mode is initiated by setting the data mode line (NOC-1, bit 0) to a 0. Display mode addressing is identical to that for data mode addressing except that there are only 36 words per sector. Words are separated by 12 bit times so that each word (two characters) is separately addressable. As in the data mode, the address for a read operation is always one greater than the address for a write operation. When reading a complete sector, address 0 is inhibited so that the original contents of the shift register is discarded. Word 0 (address 1) contains the active mark and the last three words overlap the trace time. These four words are not displayed but can be used to store information on the microdrum.

DISPLAY MODE WRITE SEQUENCE

To perform this sequence, the BC places the address information on the address lines (NOC-1), sets the data mode line to a 0, places a 16-bit data word on the data lines, and sends a write request (NOC-1, bit 1) to the microdrum. Address compare lockout (NOC-3, bit 9) can also be set to a 1 to select word addressing if desired. If word addressing is selected the BC must send a word address in addition to a sector address. If word addressing is not selected, the BC sends only the sector address. Head selection in the display mode is identical to that used in the data mode.

If data from the keyboard is being written on the drum, the BC program must read in each 8-bit character from the keyboard, pack two 8-bit characters into a 16-bit word, and then write each word on the drum. The characters appear on the display as they are written on the drum. To indicate visually that a line is being written, the BC program can set bit 8 of word 00 in the sector being written. This will cause an active mark (a period) to be displayed at the end of the corresponding line on the CRT.

The BC can also set the character mark signal (NOC-1, bit 8). If this signal is a 0, the cursor is placed under the even character (character 0 of the two-character word) of the word being written and displayed. If the signal is a 1, the cursor is placed under the odd character (character 1 of the two-character word) being written and displayed.

DISPLAY MODE READ SEQUENCE

The read sequence is identical to the write sequence except that the BC sends a read request to the drum rather than a write request.

DISPLAY MODE CHARACTER GENERATION

Data from the microdrum display track is translated by the logic circuits into characters of the ASCII 64-character subset. Each of the 16-bit words on the microdrum contains two 8-bit characters. The display will hold 18 lines of data with 64 characters per line. Since each line is equivalent to one sector on the drum track, the display will hold one complete track of data. However, the first word and the last three words of each sector are not displayed. Tracks usually used for display purposes are all tracks except 2A, 2B, and 28.

TYPICAL PROGRAMMING SEQUENCES

The following procedures indicate an acceptable sequence of programming steps for the microdrum. The first few steps apply to the general set-up procedure prior to transferring data in either a read or a write operation.

SELECTION OF DESIRED MICRODRUM TRACK (HEAD)

1. If desired head is same as current head selected, proceed to Establishing Starting Sector and Word Address For Data Transfer. If not, continue to step 2.
2. Select head 00_8 using NOC-3, bits A through F (10 through 15) to activate head change delay in microdrum.
3. Select desired head (track on microdrum) using NOC-3, bits A through F.

ESTABLISHING STARTING SECTOR AND WORD ADDRESS FOR DATA TRANSFER

1. Adjust address if in read and/or display mode.
2. Transfer sector address to microdrum via NOC-1, bits 3 through 7, and transfer word address to microdrum via NOC-1, bits 9 through F.

DATA TRANSFER

1. If in checksum mode, proceed to Checksum Read or Write Mode.
2. If in word mode, proceed to Reading or Writing in Word Mode.

3. If data mode is desired set NOC-1, bit 0; set NOC-3, bit 9 (compare lockout), and proceed to Reading or Writing in Data Mode.

4. If in display mode for either read or write proceed to Reading or Writing in Data Mode.

READING OR WRITING IN DATA MODE

1. If in write mode, proceed to step 7.
2. Clear and set read request via NOC-1, bit 2.
3. Check that drum busy is equal to zero via NIC-3, bit 0.
4. Transfer data to BC via NIC-0, bits 0 through 15.
5. If read operation is not complete, repeat steps 2 and 4.
6. Proceed to Terminating Microdrum.
7. If in write mode, transfer data to microdrum via NOC-0, bits 0 through 15.
8. Clear and set write request via NOC-1, bit 1.
9. Check that drum busy is equal to zero via NIC-3, bit 0.
10. If write operation is not complete, repeat steps 7 through 9.
11. Proceed to Terminating Microdrum.

CHECKSUM READ OR WRITE MODE

1. Set checksum mode NOC-3, bit 0 to 1, set checksum word address in memory to zero. If in data mode, set data mode via NOC-1, bit 0 and set compare lockout via NOC-3, bit 9.
2. If in write mode, proceed to Checksum Write Mode.
3. Set read request via NOC-1, bit 2.
4. Check that drum busy is equal to one via NIC-3, bit 0.
5. Collect data from microdrum and add amount of data collected to checksum word total.
6. If data transfer is complete, proceed to step 12. If not, continue with step 7.
7. Check that drum busy is equal to zero via NIC-3, bit 0.
8. Collect data from microdrum and add amount of data collected to checksum word total.

9. If data transfer is not complete go back to step 4.
10. Wait for drum busy to equal 1 (NIC-3, bit 0).
11. Proceed to step 13.
12. Wait for drum busy to equal 0 (NIC-3, bit 0).
13. Transfer checksum word from microdrum to BC.
14. Compare transferred checksum word with generated checksum word.
15. Proceed to Terminating Microdrum.

CHECKSUM WRITE MODE

1. Transfer data (NOC-0, bits 0 through 15) to microdrum and add amount of data transferred to checksum word total.
2. Set write request via NOC-1, bit 1.
3. Wait for drum busy to equal 1 (NIC-3, bit 0).
4. Transfer data via NOC-0, bits 0 through F and add amount of data transferred to checksum word total.
5. If data transfer is complete, proceed to step 11.
6. Wait for drum busy to equal 0 (NIC-3, bit 0).
7. Transfer data via NOC-0, bits 0 through F and add amount of data transferred to checksum word total.
8. If data transfer is complete proceed to step 9. If not repeat steps 3 through 7.
9. Wait for drum busy to equal 1 (NIC-3, bit 0).
10. Proceed to step 12.
11. Wait for drum busy to equal 0 (NIC-3, bit 0).
12. Transfer checksum word to microdrum via NOC-0, bits 0 through 15.
13. Wait for logic circuits to write checksum on microdrum (14.4 milliseconds).
14. Proceed to Terminating Microdrum.

READING OR WRITING IN WORD MODE

1. If in display mode, proceed to step 11.

2. If in write mode, proceed to Terminating Microdrum (Attempting to do a write operation in the word mode is illegal).
3. For read mode, set read request (NOC-1, bit 2) and data mode (NOC-1, bit 0).
4. Set return from monitor or control program to continue with step 6.
5. Exit to monitor or control program.
6. Collect data (NIC-0, bits 0 through F).
7. If data collection is complete, proceed to Terminating Microdrum. If not, continue with step 8.
8. Update microdrum sector and word address.
9. Transfer sector address (NOC-1, bits 3 through 7) and word address (NOC-1, bits 9 through F) to microdrum.
10. Clear and set read request (NOC-1, bit 2). Repeat steps 4 through 7.
11. If in read mode, set read request. Repeat steps 4 through 7.
12. Transfer data to microdrum.
13. Set write request (NOC-1, bit 1).
14. When returning from monitor or control program continue with step 16.
15. Exit to monitor or control program.
16. Clear write request.
17. If data transfer is complete, proceed to Terminating Microdrum.
18. Update sector and word addresses (NOC-1, bits 3 through 7 and 9 through F, respectively).
19. Transfer sector and word address to microdrum.
20. Transfer 16 data bits (NOC-0, bits 0 through F) to microdrum. Repeat steps 13 through 20.

TERMINATING MICRODRUM

1. Clear, read, write, and data mode requests (NOC-1, bits 0, 1, 2).
2. Clear checksum mode select (NOC-3, bit 0) and compare lockout (drum) select (NOC-3, bit 9).

DISPLAY SETUP

1. Select display track (NOC-3, bits A through F).
2. Reset current head selected. If desired head is same as current head selected, proceed to Establishing Starting Sector and Word Address for Data Transfer.
3. Transfer sector address (NOC-1, bits 3 through 7), word address (NOC-1, bits 9 through F, and character mark (NOC-1, bit 8) to microdrum.
4. Exit to program that originated display request.

ENTRY KEYBOARD

The keyboard used in the SCU is illustrated in Figure 3-4, and generates character codes for the 64 characters in columns 2, 3, 4, and 5 (refer to Table 3-1) of the ASCII character coding set. Figure 3-4 also lists the normal channels used for the transfer of each bit.

The keyboard also generates function codes for the function keys listed in Table 3-2. Either character or function codes are transferred to the BC via NIC-1, bits 1 through 7. Codes for characters sent to the BC are accompanied by the data strobe signal (NIC-1, bit B).

KEYBOARD CONTROL KEYS

Each of the keyboard control keys places an octal code on the character/function lines when pressed, and each key is used for a specific purpose in program control. Table 3-2 lists each key, its hexadecimal equivalent, and an application pertaining to program control.

KEYBOARD INTERFACE SIGNALS (TO NIC) (Refer to Figure 3-4)

CHARACTER FUNCTION/CODES

These seven lines (bits 1 through 7) are used to carry the function codes or character codes from the keyboard to normal input channel 1.

ALARM DISABLE (ALERT)

This signal is transmitted when the ALERT pushbutton switch is pressed.

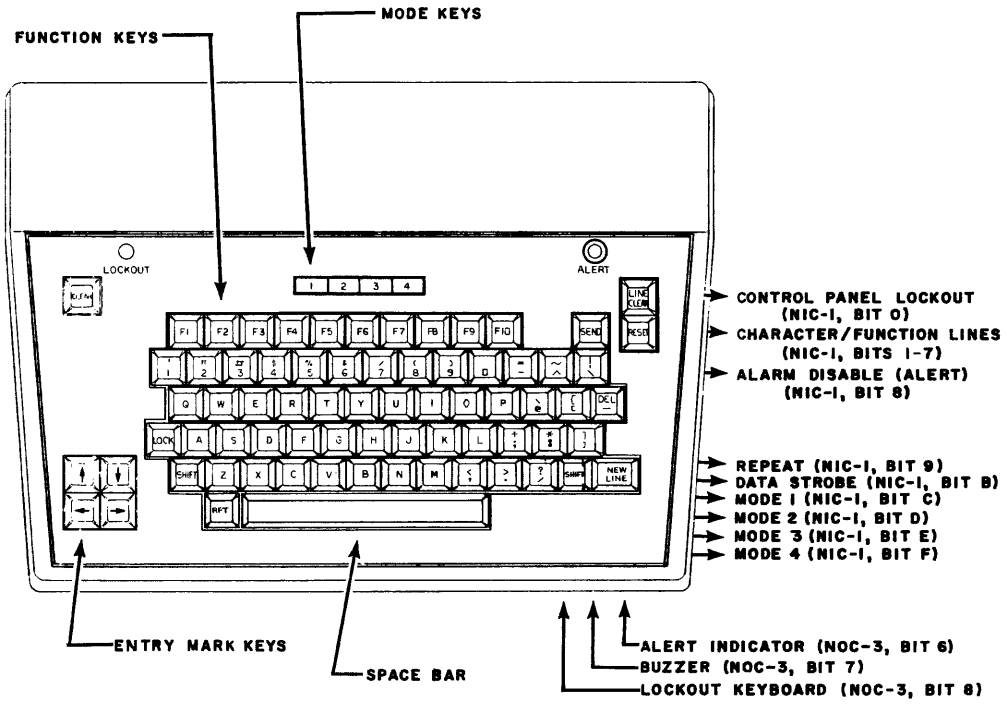


Figure 3-4. SCU Keyboard and I/O Signals

TABLE 3-1. ASCII CODING SET FOR KEYBOARD

b7 →						0	0	0	0	1	1	1	1
b6 →						0	0	1	1	0	0	1	1
b5 →						0	1	0	1	0	1	0	1
Bits	b4	b3	b2	b1	Column	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	CS	-	=	M	]	m	}
	1	1	1	0	14	SO	RS	.	>	N	^	n	~
	1	1	1	1	15	SI	US	/	?	O	—	o	DEL
						Not Used						Not Used	

TABLE 3-2. KEYBOARD CONTROL KEY FUNCTION CODES

Function	Hex Code	Software Application
CLEAR	00	Clears display and places entry marker in first character position of line without affecting displayed area.
LINE CLEAR	03	Clears an entry line and places entry marker in first character position of line without affecting displayed area.
INTER (Interrupt)	02	Signals BC to read active lines on display and to remove active marks.
NEW LINE	0A	Signals the software that the keyboard entry is complete.
RESET	1C	Places entry marker in first character position of data field without affecting display data.
Entry mark keys		
↑	08	No response
→	0B	These keys advance the cursor one character position left or right at a fixed rate without affecting the displayed area.
←	11	
↓	12	No response
Function keys		Functions unassigned; however, functions such as insert, delete, etc., can be used.
F1	71	
F2	72	
F3	73	
F4	74	
F5	75	
F6	76	
F7	77	
F8	78	
F9	79	
F0	70	
Mode switches 1 2 3 4		Functions unassigned; however, modes such as off-line, on-line, etc., could be assigned.

CONTROL PANEL LOCKOUT

This signal indicates the position of the key-operated PANEL LOCKOUT switch on the SCU control panel. A 1 indicates the switch is in the ON position. Refer to section 5 of this manual for a list of SCU control panel switches affected by this switch.

KEYBOARD LOCKOUT SWITCH

When the KEYBOARD LOCKOUT switch on the SCU control panel is ON, this signal sets bit A of NIC-1. The program then locks out that portion of the display keyboard which is to be protected in the connected station.

REPEAT KEY

Bit 9 of NIC-1 sets when this key is pressed. Repeats previous entry every 500 milliseconds when pressed. If there is no previous entry, the repeat command flag is set.

DATA STROBE

This bit accompanies the character codes listed in Table 3-2 when a character key is pressed.

MODE SIGNALS

These four lines transmit the signals assigned to the individual MODE switches when the switches are pressed.

KEYBOARD INTERFACE SIGNALS (FROM NIC) (Refer to Figure 3-4)

ALERT

This signal lights the ALERT indicator on the keyboard. This signal can originate in a BC normal output channel or as the result of a deadman logic timeout (refer to Start Deadman Timeout for details).

BUZZER

This signal sounds the audible alarm in the keyboard and can originate in a BC normal output channel or as the result of a deadman logic timeout.

LOCKOUT KEYBOARD

This signal lights the LOCKOUT indicator on the keyboard and indicates that the keyboard cannot be used. The signal originates in the BC or at the keyboard lockout switch on the station control panel.

SCU B COUPLER INTERFACE

The B coupler in a SCU provides an interface between the BC block transfer channel in the SCU and a STAR data channel. The coupler forms the B (active) end of the STAR channel. Figure 3-5 shows the relationship of the SCU B coupler to other elements in the SCU.

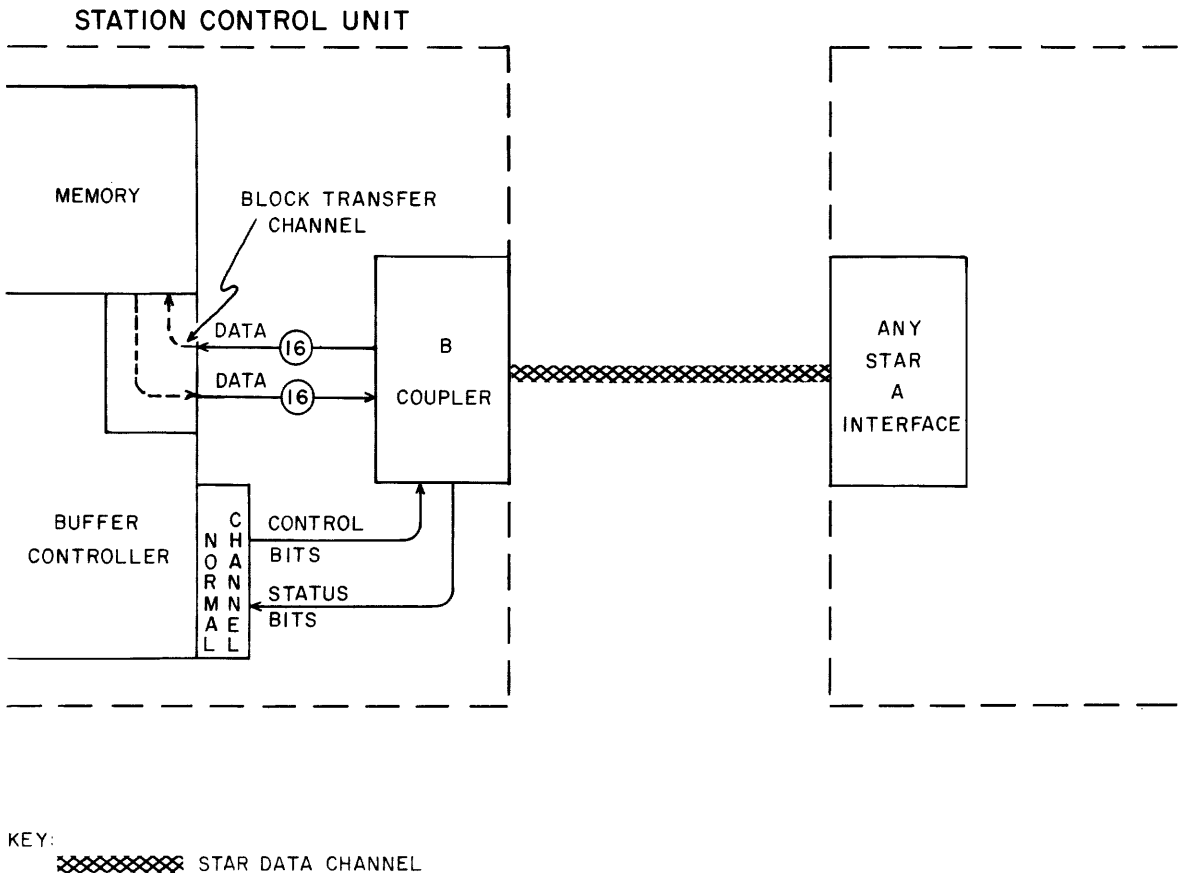


Figure 3-5. SCU B Coupler System Relationship

In SCU-only stations (such as the unit record station), the B coupler provides a bi-directional data path to an HLP as shown in Figure 3-6. This arrangement allows the SCU to read or write memory in the HLP.

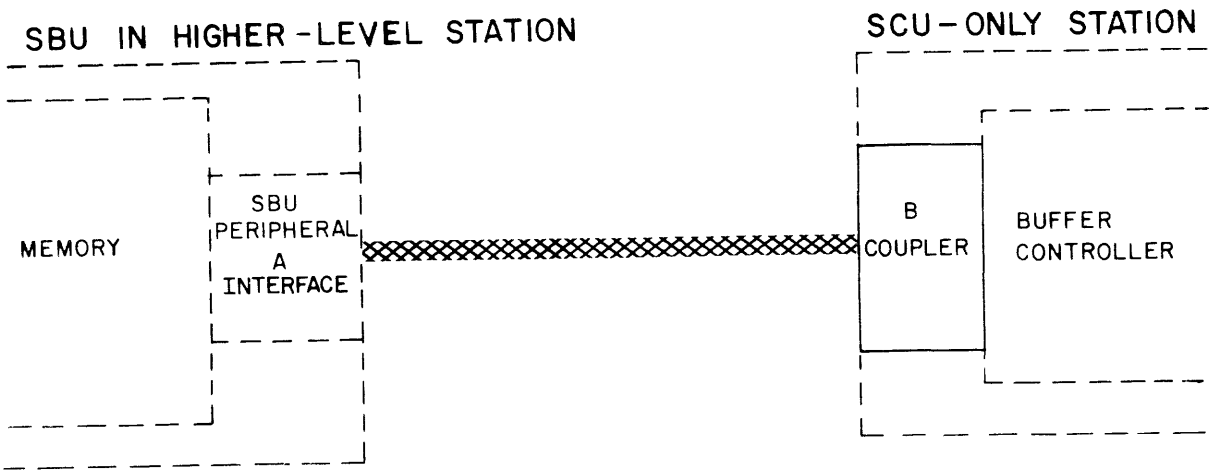


Figure 3-6. B Coupler Used in SCU-Only Station

In SCU/SBU stations (examples: paging station, disk station) the B coupler provides a path to the SBU that allows the SCU to write into or read from SBU memory. Figure 3-7 shows this arrangement.

One or more B couplers are present in all SCUs. Up to four B couplers can be installed in a SCU, however, only one can transfer data at a given time. A select code, issued on the normal channels, allows the BC, in the SCU, to select one of the couplers.

In SCU-only stations a secondary function of the B coupler is to allow autoloading of the SCU memory across the STAR data channel.

Operation of the B coupler is controlled by the BC via normal channel bits. The main control responsibilities of the SCU are:

- Issue a memory address to the A end of the channel. This address specifies the beginning of the area in A memory that the SCU wishes to read or write.
- Issue a function code specifies the type of transfer operation (read or write).
- Initiate data transfer by executing a block transfer instruction.
- Monitor status conditions.

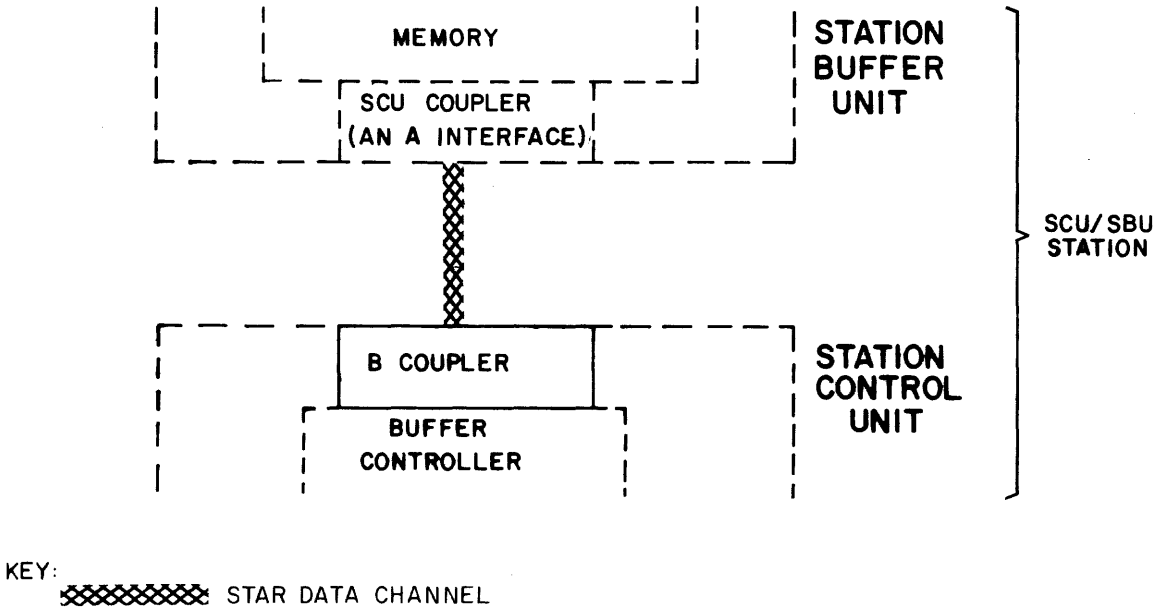


Figure 3-7. B Coupler Used in SCU/SBU Station

DATA TRANSFER OPERATIONS

The B coupler together with an A interface at the distant end of the channel, can perform the following data transfer operations.

• Write	Transfer a block of 32-bit data words (two 16-bit data words) from SCU memory to memory at the A end of the channel.
• Block read	Transfer a block of 32-bit words from A memory to the SCU memory.

- Special function† — Transfer one 32-bit word from a specified location in A memory to SCU memory, then store all zeros in this location in A memory.

All transfers on the STAR data channel should be multiples of 32-bit words. Each 32-bit word consists of two consecutive 16-bit words.

Data transfers on the STAR data channel must be initiated by the B end of the channel. To prepare the A end of the channel for a data transfer operation the BC sends the following information to A via the B coupler.

1. A function code that specifies the type of operation to be performed (write, block read, or destructive read).
2. A 21-bit starting address (sent as two 16-bit words) that specifies the beginning of the source or destination area in memory at the A end of the channel.

Then, the BC must execute an input-block transfer instruction or output-block transfer instruction to begin the data transfer.

For write and block read operations, the number of words transferred is determined by the word count of the block transfer instruction. The operation ends when the block transfer instruction terminates.

The sequence of programming steps required to initiate a data transfer are described later in this section.

FUNCTION CODES

Two types of function codes control activities on the STAR channel:

1. Operation function codes, sent from the B coupler to the A end of the channel.
2. System control functions, sent from A to the B coupler.

OPERATION FUNCTION CODES

The BC issues function codes (Table 3-3) codes to the B coupler via normal channel bits. The coupler forwards all function codes to the A end of the channel.

† Some A interfaces cannot perform this operation.

TABLE 3-3. B COUPLER OPERATION FUNCTION CODES

Write	010_2
Block read	101
Read	001
Special function	011
Data	100
End of operation	111
Null	000
Illegal code	110

Write (010_2) — This code conditions the A end of the channel for transfer of a block of words from B to A.

Block read (101_2) — This code conditions the A end of the channel for transfer of a block of words from A to B.

Read (001_2) — This code is redundant. Its effect is identical in all respects to that of block read (101_2).

Special function (011_2) — This code causes the B end of the channel to perform a predetermined special function other than normal input/output. Not all A interfaces can perform this operation.

Data (100_2) — This code must be issued after a write (010_2) function code. It must remain on the normal output channel throughout the write operation.

End of operation (111_2) — This code should be issued at the end of a data transfer operation to inform the A end of the channel that the operation is complete (clears the A end of the channel).

Null (000_2) — This code does not specify an operation. It is transmitted along with the lower half of an address or output (write) data word to inform A that the lower part of a 32-bit word is on the channel. The B coupler automatically transmits this code. From a programming standpoint, it is transparent.

Illegal code (110_2)

This code should not be used. If issued, no operation results and the A end of the channel returns an illegal response bit that sets a status bit that can be sensed by the BC.

Some types of A interfaces also treat code 011_2 (destructive read) as an illegal code.

SYSTEM CONTROL CODES

The codes listed in Table 3-4 provide a means for the A end of the channel to send control information back to the B coupler.

TABLE 3-4. B COUPLER SYSTEM CONTROL CODES

Name	Code
Channel flag	01_2
External flag	10
Suspend	11
Invalid	00

NOTE

When the B coupler is used for the path between the SCU and SBU in SBU/SCU stations, these codes do not apply. The A interface in the SBU (the SCU coupler) cannot send system control codes.

Channel flag (01_2)

This code sets the channel flag status bit in the B coupler. It indicates that a message for B has been placed in a prearranged area in memory at the A end of the channel. The SCU should respond by initiating a read operation to obtain the message.

External flag (10_2)

This code forces the SCU to master clear and then autoload from the microdrum. The autoload operation reads 1152 16-bit words from the drum and enters this data into the first 1152 locations of SCU memory. This code also sets the external flag status bit in the B coupler.

Suspend (11_2)	This code causes the B coupler to end a data transfer operation and forces the BC to terminate the block transfer instruction. This code also sets the suspend status bit in the B coupler.
Invalid (00_2)	This code sets the invalid status bit in the B coupler. The A end of the channel does not transmit this code. It appears on the channel only as the result of a transmission fault. Like a suspend (11_2) code, it causes a data transfer operation to terminate. When this condition occurs, the SCU software should send an interrupt to inform the A end of the channel (or other SCUs) of the malfunction.

In addition to setting individual status bits, all of the system function codes, except channel flag (01_2), also sets the B coupler fault status bit. All of these status bits can be sensed by the BC on one of the normal input channels.

STARTING ADDRESS FORMAT

Before the start of a data transfer operation the SCU must issue a 21-bit starting address (Figure 3-8) by performing a two-word output-block transfer operation. The B coupler forwards the two address words to the A end of the channel. The starting address is the lowest address of the area in A memory read or written during the data transfer operation.

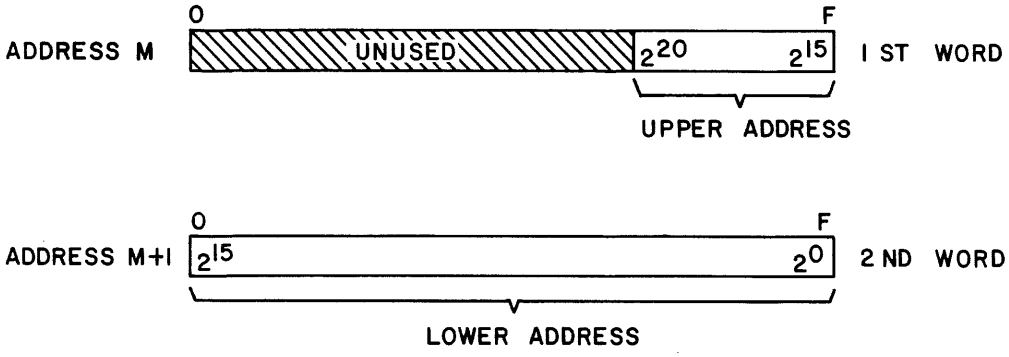


Figure 3-8. B Coupler Starting Address Word Formats

The 21-bit address provides for addressing up to 2,098,208 words of memory. For smaller memories not all of the address bits are required. For example, if the A memory is a standard SBU containing the equivalent of 16K 32-bit words (actually 32K of 16-bit words), only address bits 2^{14} through 2^{0} are used. The upper 7 address bits are not significant in this case but must be sent to the SBU.

PARITY CHECKING

Three types of parity errors are reported to the BC via B coupler status bits on the normal input channels.

OPERATION FUNCTION CODE PARITY

The B coupler appends an odd parity bit to a 3-bit function code sent to the A end of the channel. If A detects a parity error, it ignores the function code and returns signals to the B coupler that set (1) the fault status bit, (2) the parity error fault status bit, and (3) the illegal status bit.

A MEMORY PARITY

During block read operations, data read from A memory is checked for correct parity. If a parity error occurs, A sends a signal that sets (1) the fault status bit and (2) the parity error fault status bit.

TRANSMISSION PARITY

During block read operations, the B coupler parity checks data coming in from the STAR channel. If a parity error occurs on the STAR channel, the following status bits set: (1) fault status and (2) read parity error status.

Any of these parity errors will terminate the data transfer operation in progress.

TERMINATING CONDITIONS

Several conditions will cause the B coupler to terminate a data transfer operation.

- **Block transfer operation complete** — A data transfer operation ends when the number of words specified in the word count for an input or output-block transfer instruction is transferred. This is the normal terminating condition. The condition bit in the BC sets to indicate that the block transfer instruction has terminated normally.
- **Parity error** — Any of the three parity errors described previously stops the data transfer operation and forces the block transfer instruction to terminate.
- **Control function from A** — Any one of the three control functions listed below causes the coupler to end a data transfer operation and forces the BC block transfer instruction to terminate.
  1. Suspend (code 11_2)
  2. External flag (code 10_2)
  3. Invalid (code 00)
- **Coupler timeout** — If a fault at either end of the channel causes the B coupler to stop transferring data, a time-out circuit in the coupler and the drum and display logic forces the block transfer operation to terminate. The coupler fault and time-out status bits are set when this condition occurs.

Any time a data transfer operation is terminated before the block transfer instruction word count is satisfied, the BC condition bit is left in the false state.

INTERRUPTING THE A INTERFACE

The B coupler allows the SCU to send an interrupt to A by setting a normal channel bit. At the A end of the channel, the interrupt sets a flag that must be sensed by software in SCU-only stations. This condition is not true in SCU-SBU configurations where the A interface is an SCU coupler.

SCU CONTROL OF THE B COUPLER

Figure 3-9 shows the normal channel bits available to the BC for control of the B coupler. The bit assignments shown apply to all of the STAR stations.

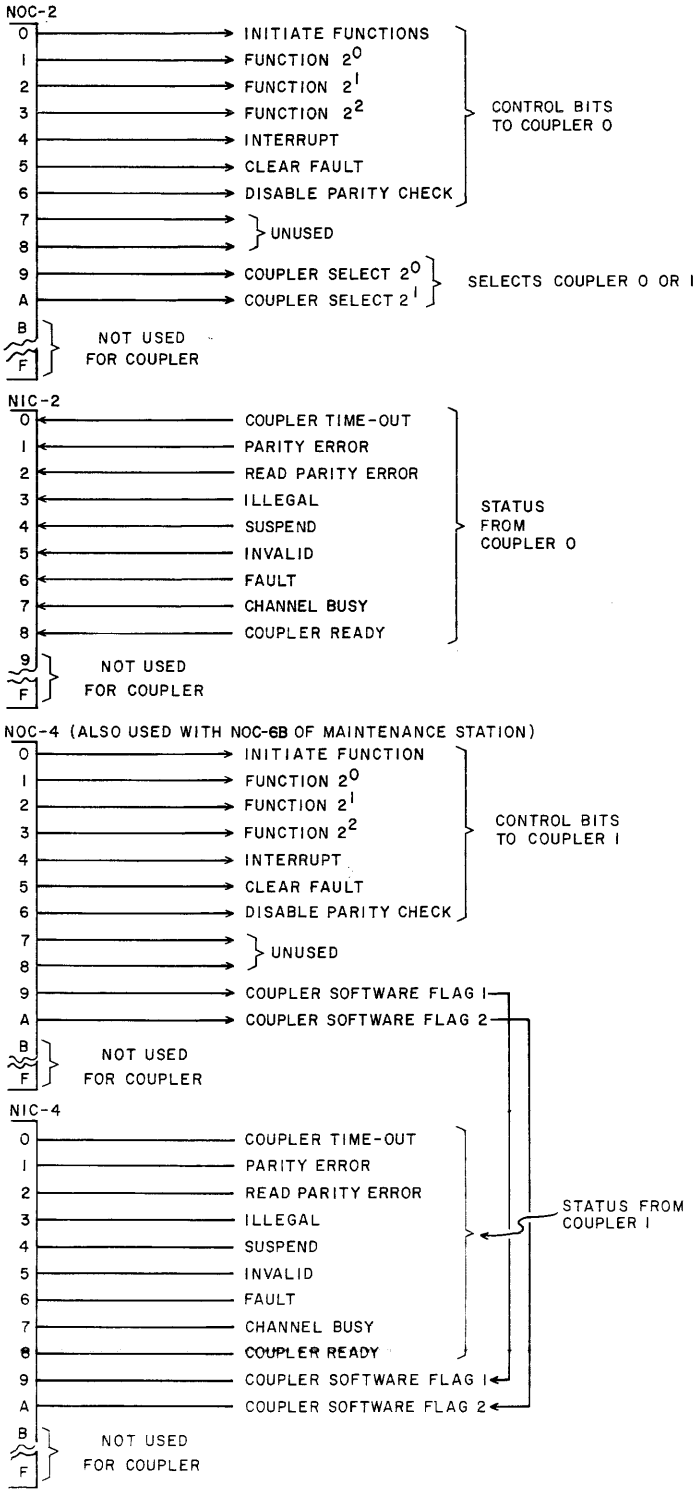


Figure 3-9. Coupler Normal Channel Bits

Table 3-5 lists and explains the functions of the normal channel bits used with the SCU B coupler.

TABLE 3-5. NORMAL CHANNEL BIT DESCRIPTIONS, SCU B COUPLER

Channel	Bit	Name	Function
NOC-2,-4	0	Initiate functions	When 1, this bit gates the function code (NOC-2,-4, bits 1, 2, and 3) into the B coupler. It must be used to issue all operation function codes, except the data code (100_2). It should be set at the beginning of a data transfer operation and cleared after the two starting address words are issued.
NOC-2,-4 NOC-2,-4 NOC-2,-4	1 2 3	Function bit 2^0 2^1 2^2	These bits are used to issue operation function codes.
NOC-2,-4	4	Interrupt	Setting this bit sends an interrupt signal to the A end of the channel. The bit may be cleared immediately after it is set.
NOC-2,-4	5	Clear fault	Setting and clearing this bit clears the following status bits. • Coupler time-out status • Parity error fault status • Read parity error status • Illegal status • Suspend status • Invalid status • Fault status

TABLE 3-5. NORMAL CHANNEL BIT DESCRIPTIONS, SCU B COUPLER (Cont'd)

Channel	Bit	Name	Function
NOC-2,-4	6	Disable parity check	When set, this bit disables the transmission parity check that the B coupler performs on data sent from A. It also prevents the fault bit from setting as a result of other parity errors detected.
NOC-2	9	Coupler select 2^1	These bits carry the coupler select code. Code 00 selects B coupler 0 Code 01_2 selects B coupler 1 Code 10_2 unused Code 11_2 standard stations The select code must remain on the normal channels continuously to keep a coupler selected.
NOC-2	A	Coupler select 2^0	
NOC-4	9	Coupler software flag 1	These two bits do not affect operation of the B coupler in any way. They are simply wired directly back to NIC-4, bits 9 and A. They are available for use by the software to record which coupler is selected.
NOC-4	A	Coupler software flag 2	
NIC-2,-4	0	Coupler timeout status	When set, this bit indicates that the B coupler failed to transfer a starting address or data in response to an input or output block transfer instruction. This condition is the result of a hardware malfunction at either end of the STAR channel.
NIC-2,-4	1	Parity error fault status	When set, this bit indicates that the A end of the channel has detected: 1. A parity error in a function code sent by the B coupler, or 2. A parity error in data read from A memory during block read.

TABLE 3-5. NORMAL CHANNEL BIT DESCRIPTIONS, SCU B COUPLER (Cont'd)

Channel	Bit	Name	Function
NIC-2,-4	2	Read parity error status	When set, this bit indicates that a transmission fault has occurred on the STAR channel during a block read operation.
NIC-2,-4	3	Illegal status	When set, this bit indicates that the A end of the channel has: 1. Received an illegal control function code from the B coupler. 2. Detected a parity error in a control function code sent from the B coupler.
NIC-2,-4	4	Suspend status	When set, this bit indicates that the A end of the channel has sent a suspend control function code.
NIC-2,-4	5	Invalid status	When set, this bit indicates that an invalid code has appeared on the control function lines from A.
NIC-2,-4	6	Fault	When set, this bit indicates that one of the following conditions has occurred. • Coupler timeout • Transmission parity error during read • Control function parity error • A memory parity error during read • Illegal operation function code to A • Suspend control function code from A • External flag control function code from A
NIC-2,-4	7	Channel busy status	When set, this bit indicates that the B coupler is transferring a data or address word. The bit sets each time the B coupler requests a half-word from A. The bit clears each time transfer of a word is completed.

TABLE 3-5. NORMAL CHANNEL BIT DESCRIPTIONS, SBU B COUPLER (Cont'd)

Channel	Bit	Name	Function
NIC-2, -4	8	Coupler ready status	When set, this bit indicates that the B coupler is ready to perform a data transfer operation. The bit goes to zero when a data transfer is initiated. If a transfer terminates early because of a fault, this bit remains clear. It goes to 1 when the fault status bit is cleared. If the SCU is in off-line mode (OFF LINE switch ON) the ready status bit is forced to zero.
NIC-4	9	Coupler software flag 1	These bits do not indicate any conditions in the B coupler hardware. They are wired directly back to NOC-4, bits 9 and A. They are available for use by software to record which coupler is selected.
NIC-4	A	Coupler software flag 2	

PROGRAMMING SEQUENCES

The following paragraphs illustrate a recommended programming sequence for driving the SCU B coupler. Normal channel bits cited are for coupler 0.

1. Select coupler
   a. Place select code for coupler 0 on NOC-2, bits 9 and A.
   b. Test NIC-2, bit 8 (ready status).
      1) If 0, go to step 5
      2) If 1, go to step 2.
2. Issue function code and starting address
   a. Clear initiate functions bit (NOC-2, bit 0).
   b. Place operation function code on NOC-2, bits 1, 2, and 3.

	Bit 1	Bit 2	Bit 3
Write	0	1	0
Block read	1	0	1

c. Set initiate functions bit (NOC-2, bit 0).

d. Issue starting address.

Execute two-word output-block transfer instruction to transmit address.

e. Test BC condition bit. (Indicates normal completion of block transfer instruction.)

1) If condition bit = 1 (normal), go to step 2f.
2) If condition bit = 0 (abnormal), go to step 5.

f. Test status bits (NIC-2) for errors. If there are errors, go to step 5.

g. Wait for ready status (NIC-2, bit 8).

h. Clear function select bits (NOC-2, bits 1, 2, and 3) and function initiate bit (NOC-2, bit 0). Go to step 3.

3. Initiate data transfer.

a. If write selected, issue data function code (set NOC-2, bit 1).

b. Start data transfer.

1) If write, execute output-block transfer instruction.

2) If read or destructive read, execute input-block transfer instruction.

(The block transfer instruction word count determines length of the data transfer. For destructive read, the word count must be limited to two 16-bit words).

c. Test BC condition bit.

(Indicates normal completion of block transfer instruction.)

1) If condition bit = 1 (normal), go to step 3d.
2) If condition bit = 0 (abnormal), go to step 5.

d. Test status bits (NIC-2). Go to step 5, if there are errors.

e. Go to step 4.

4. End of operation

a. Place end-of-operation function code (111_2) on NOC-2, bits 1, 2, and 3.

b. Set initiate function bit (NOC-2, bit 0).

c. Test status bits (NIC-2) for errors and wait for ready status.

1) If no errors, exit to calling program.
2) If an error exists, go to step 5.

5. Recovery sequence

   a. Clear function select bits (NOC-2, bits 1, 2, and 3) and function initiate bit (NOC-2, bit 0).

   b. Place end-of-operation function code (111_2) on NOC-2, bits 1, 2,and 3.

   c. Set function initiate bit (NOC-2, bit 0).

   d. Set and clear fault bit (NOC-2, bit 5).

   e. Check coupler time-out status (NIC-2, bit 0).

      1) If time-out status = 1, a nonrecoverable hardware fault has occurred. Exit to calling program with an abort indication.

      2) If time-out status = 0, go to step 5f.

   f. Test status bits (NIC-2) for errors.

      1) If errors exist after third retry, exit to calling program with abort indication.

   g. Retry starting at step 1.

3000 SCU INTERFACE

The 3000 SCU interface acts as a data channel between CDC 3000 peripheral controllers and SCU normal channels. It simulates a standard CDC 3000 data channel. Figure 3-10 shows the relationship between the 3000 interface and other elements in the system.

The 3000 interface is limited to very basic data transfer operations. Each 12-bit data byte transferred to or from a 3000 controller requires a separate input-to-A or output-from-A instruction. Data is transferred through the interface without modification; no code conversion or assembly/disassembly takes place.

Control signals from the 3000 peripheral controllers are routed through the 3000 interface and terminate at normal channel bits. The BC controls peripheral devices by monitoring and manipulating these control signals. The 3000 interface performs only minor control functions; most of the control must be supplied by the driving software.

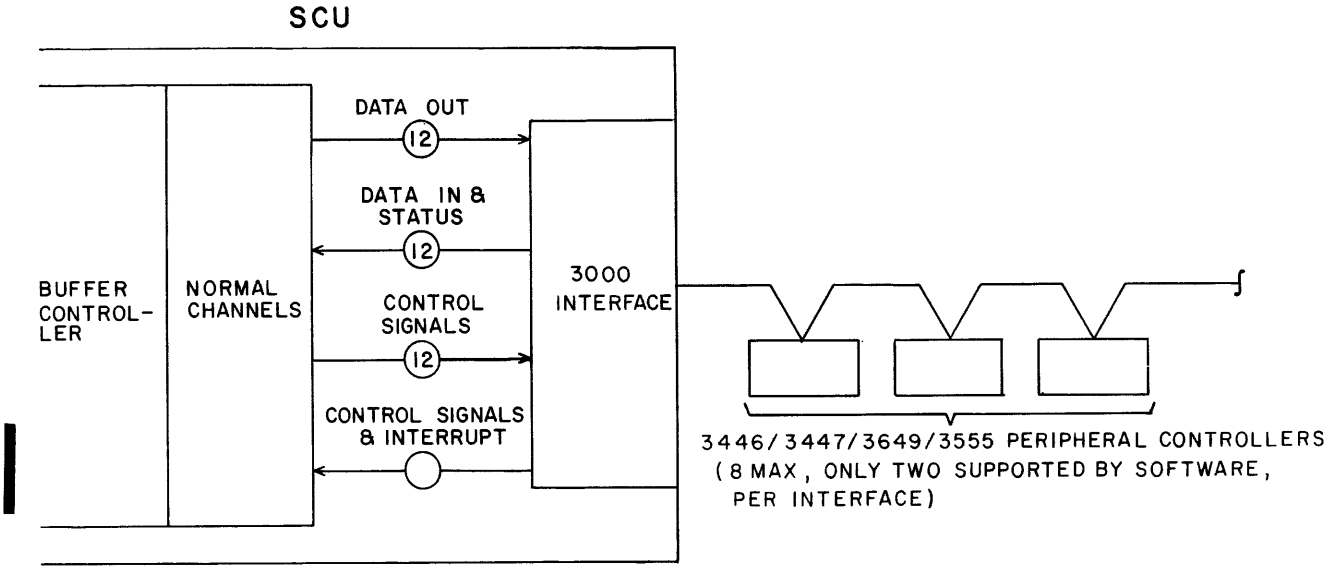


Figure 3-10. 3000 SCU Interface System Relationship

Two 3000 interfaces can be installed in the same SCU. The two interfaces are completely independent and are capable of transferring data concurrently, subject to software limitations.

This section describes the programming characteristics and function of the 3000 interface. To fully understand this material, the reader should be familiar with 3000 I/O programming conventions and the 3000 I/O channel interface specifications.†

3000 INTERFACE OPERATIONS

The 3000 interface and supporting software performs the following operations.

- Connect operation — Transmits a 12-bit connect code from a NOC to the peripheral controllers.

  This code activates one of the eight possible controllers linked to the 3000 interface. It may also select one of several peripheral units subordinate to the controller.

† Refer to the 3000 I/O specification manual.

- Function operation

  Transmits a 12-bit function code from a normal channel to the previously connected peripheral controller.

  Function codes are commands that condition a peripheral controller for I/O operations.

- Write operation

  Transmits a series of 12-bit data bytes from an NOC to the previously connected peripheral controller.

  An output-from-A instruction is required to transmit each 12-bit byte. The data byte comes from the lower 12 bits of the A register.

  An odd parity bit must accompany each output data byte. The 3000 interface does not automatically generate the required parity bit. It is the responsibility of the driver software to generate an odd parity bit and place it in the NOC along with the data byte.

- Read operation

  Transfers a series of 12-bit data bytes from a peripheral controller to the BC via a NIC.

  A separate input-to-A instruction is required to read in each 12-bit byte. The input byte appears on the lower 12 bits of the input channel.

  An odd parity bit, generated by the 3000 peripheral controller, accompanies each input data byte. It appears on the NIC along with the 12 data bits. The 3000 interface does not check parity. If a parity check is desired, the driver software must be written to analyze each data byte and associated parity bit to determine if a parity error occurred during transmission from the peripheral controller to the 3000 interface.

- Read status

  Transfers a 12-bit status word from a peripheral controller to the BC via a NIC.

  An input-to-A instruction is required to read in the status word.

3000 INTERFACE NORMAL CHANNEL SIGNALS

Figure 3-11 shows the normal channel signals used by the SCU to control the 3000 interface. Channel and bit assignments given here are for the unit record station. 3000 SCU interface bit assignments may be different in other stations. Refer to appendixes A and B of this manual for a complete list of normal channel bit assignments for all stations.

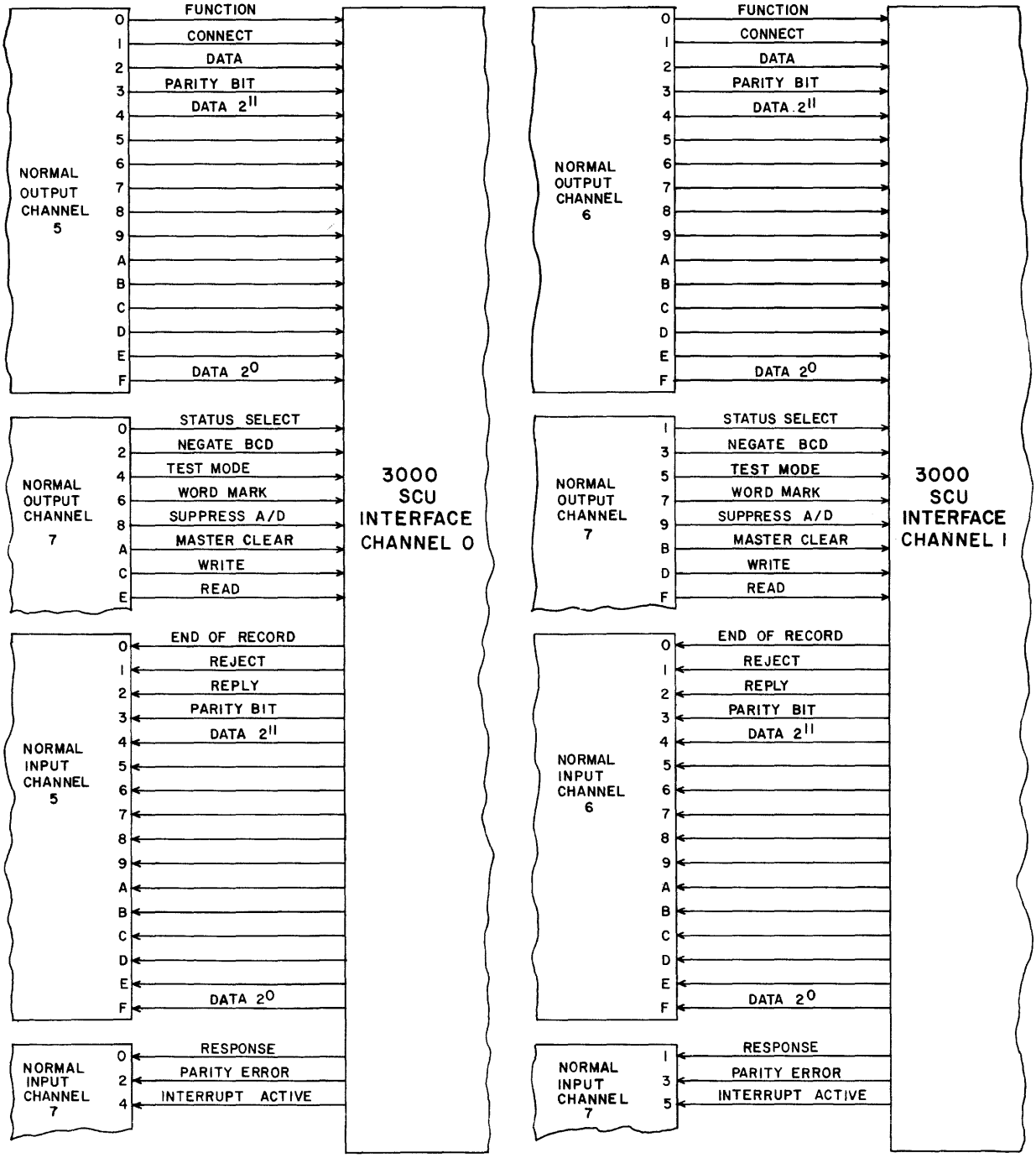


Figure 3-11. 3000 SCU Normal Channel Signals

Table 3-6 describes each of the normal channel bits used to control the 3000 interface.

TABLE 3-6. NORMAL CHANNEL BIT DEFINITIONS, 3000 SCU INTERFACE

Channel	Bit	Name	Function
NOC-5 and -6	0	Function	Setting this bit causes the 3000 interface to transmit a 12-bit function code and parity bit from the lower 13 bits of the output channel to the peripheral controller.
NOC-5 and -6	1	Connect	Setting this bit causes the 3000 interface to transmit a 12-bit connect code and parity bit from the lower 13 bits of the output channel to the peripheral controller.
NOC-5 and -6	2	Data signal	This bit acts as a data request to the peripheral controller. 1. Write operation; setting this bit causes the 3000 interface to transmit a 12-bit data byte and parity bit from the lower 13 bits of NOC-5 or -6. The data signal bit must remain set until the controller returns a reply (NIC-5 or -6, bit 2) 2. Read operation; setting this bit requests the controller to send a data byte. This bit must remain set until the controller returns a reply (NIC-5 or -6, bit 2) indicating that data byte is available for read-in.

TABLE 3-6. NORMAL CHANNEL BIT DEFINITIONS, 3000 SCU INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-5 and -6	3	Output parity bit	This bit is an odd parity bit associated with the 12-bit connect code, function code, or output data byte that occupies the lower 12 bits of the output channel. It is the responsibility of the driver software to generate the odd parity bit for each code or data byte transmitted to the peripheral controller.
NOC-5 and -6	4	Output data/code bit 2^0	These 12 bits hold a connect code, function code, or output data byte that is transmitted to the peripheral controller.
NOC-5 and -6	5 ↓	Output data/code bit 2^1 ↓	
NOC-5 and -6	F_{16}	Output data/code bit 2^{11}	
NOC-7	0	Status select (interface A)	When set, this bit selects the 12-bit status word, available from the peripheral controller, for input on NIC-5. When this bit is 0, data from the peripheral controller is available on NIC-5.
NOC-7	1	Status select (interface B)	Same as above for interface B. Selects status or data for NIC-6.
NOC-7	2	Negate BCD conversion (interface A)	When set, this bit signals the peripheral controller to suppress the internal/external BCD conversion performed by some 3000 controllers. 1. During a read operation, this bit directs the controller not to convert BCD codes from external to internal form. During a write operation, this bit directs the controller not to convert BCD codes from internal to external form.

TABLE 3-6. NORMAL CHANNEL BIT DEFINITIONS, 3000 SCU INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-7	3	Negate BCD conversion (interface B)	Same as above for interface B.
NOC-7	4	Test mode (interface A)	Setting this bit selects test mode. When test mode is in effect, data transmitted on NOC-5 is looped back to NIC-5. Also, each time the connect, function, or data signal bit (NOC-5, bit 0, 1, or 2) is set to transmit a data byte, the 3000 generates a pseudo reply that appears on NIC-5, bit 2. The reply indicates that data sent out on NOC-5 can be read back in on NIC-5.
NOC-7	5	Test mode (interface B)	Same as above for interface B. Data transmitted from NOC-6 is looped back to NIC-6.
NOC-7	A_{16}	Master clear (interface A)	When set, this bit causes the 3000 interface to send a master clear signal to the peripheral controller. The master clear signal clears out any internal operating conditions previously set up with functions and disconnects the peripheral controller.
NOC-7	B	Master clear (interface B)	Same as above for interface B.
NOC-7	C	Write (interface A)	When set, this bit conditions the 3000 interface and peripheral controller for a write (output) operation. The write bit must remain set throughout a write operation.
NOC-7	D	Write (interface B)	Same as above for interface B.
NOC-7	E	Read (interface A)	When set, this bit conditions the 3000 interface and peripheral controller for a read (input) operation. The read bit must remain set throughout a read operation.

TABLE 3-6. NORMAL CHANNEL BIT DEFINITIONS, 3000 SCU INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-7	F	Read (interface B)	Same as above for interface B.
NOC-7	6	Word mark (interface A)	When set, this bit causes the 3000 interface to transmit a word mark signal to the 3000 controller. Standard 3000 I/O channels, which assemble two or four 12-bit data bytes into a 24-bit or 48-bit memory word, send a word mark signal to designate the low-order byte in each memory word. Certain controllers are designed to transmit or accept only data bytes that correspond to the lowest byte in a memory word. That is, they respond only when a word mark signal accompanies the request to send or receive a data byte. In programming the 3000 interface, a safe practice is to always set this bit along with the data signal bit (NOC-5 or -6, bit 2) which is set to transmit or receive each data byte.
NOC-7	7	Word mark (interface B)	Same as above for interface B.
NOC-7	8	Suppress assembly/disassembly (interface A)	When set, this bit signals the peripheral controller to stop the normal 6-bit/12-bit assembly-disassembly performed by some controllers. 1. During a read operation, this bit causes the controller to stop assembling two 6-bit characters into each 12-bit byte. Each input byte contains a single 6-bit character in the lower six bit positions.

TABLE 3-6. NORMAL CHANNEL BIT DEFINITIONS, 3000 SCU INTERFACE (Cont'd)

Channel	Bit	Name	Function
			2. During a write operation, this bit causes the controller to assume that each output byte contains a single 6-bit character in the lower six bit positions. The controller does not use the upper six bits of the byte. This bit does not affect the operation of a controller that does not perform 6-bit/12-bit assembly-disassembly.
NIC-5 and -6	0	End of record	When set, this bit indicates that the peripheral controller has reached the end of a record during a read operation and will send no more data until a new read operation is initiated via the 3000 interface. When the last input data byte in a record appears on input channel 5 or 6, this status bit sets instead of the reply status bit.
NIC-5 and -6	1	Reject	When set, this bit indicates that the peripheral controller has rejected a connect code or function code. 1. A connect code can be rejected only by a dual-access peripheral controller that has been connected or reserved on the other access channel; the access channel is not linked to the 3000 interface. 2. A function code may be rejected: a. If it is not a valid code for the controller to which it is directed, or b. If the function code conflicts with an operation the peripheral controller is performing.

TABLE 3-6. NORMAL CHANNEL BIT DEFINITIONS, 3000 SCU INTERFACE (Cont'd)

Channel	Bit	Name	Function
NIC-5 and -6	2	Reply	When set, this bit indicates that: 1. The peripheral controller has accepted a connect code, a function code, or an output data byte. 2. During a read operation, this bit indicates that an input byte from the peripheral controller is available on NIC-5 or -6.
NIC-5 and -6	3	Input parity bit	An odd parity bit generated by the peripheral controller that accompanies an input data byte.
NIC-5 and -6	4	Input data/ status bit 2^0	Two types of information from the peripheral controller can be read in on these normal channel bits: 1. Input data 2. A 12-bit status word The status word is selected for input by one of the following normal channel bits. 1. Status select bit (NOC-7, bit 0 or 1) = 1 2. Connect bit (NOC-5 or -6, bit 1) = 1 3. Function bit (NOC-5 or -6, bit 0) = 1 If all three of these select bits are 0, bits 4 through F_{16} can be used to read in data.
	F_{16}	Input data/ status bit 2^{11}	
NIC-7	0	Response (interface A)	Indicates that the peripheral controller has sent a reply, reject, or end of record signal to the 3000 interface. The individual status bits for these signals (NIC-5 and -6, bits 0, 1, and 2) can be checked to determine the type of response.

TABLE 3-6. NORMAL CHANNEL BIT DEFINITIONS, 3000 SCU INTERFACE (Cont'd)

Channel	Bit	Name	Function
NIC-7	1	Response (interface B)	Same as above for interface B.
NIC-7	2	Output parity error (interface A)	When set, this bit indicates that the peripheral controller has detected a parity error in a function code or output data byte sent from the 3000 interface. This bit drops only if the 3000 interface master clears the controller.
NIC-7	3	Output parity error (interface B)	Same as above for interface B.
NIC-7	4	Interrupt active (interface A)	When set, this bit indicates that one of the eight possible peripheral controllers is sending an interrupt signal to the 3000 interface. To determine the source of the interrupt, the 3000 interface must connect each controller and check the controller status word.
NIC-7	5	Interrupt active	Same as above for interface B.

PROGRAMMING SEQUENCES

The following sequences outline the programming steps required to drive a peripheral controller through the 3000 interface.

MAJOR STEPS

The steps required in a driver program depend, to a large extent, on the type of peripheral controller being driven. The order of steps suggested is a basic sequence that could be used with a number of standard 3000 controllers.

1. Connect peripheral controller

   This activates one of the controllers linked to the 3000 interface. Connect must always be the first step in any I/O operation. A peripheral controller cannot respond to any command from the 3000 interface unless connected.

2. Issue function codes

   This step usually follows connect. Most peripheral controllers must be prepared for an I/O operation with one or more function codes before data transfer can begin.

3. Read controller status word

   This step is often used after function codes are issued to determine if the connected peripheral controller is ready for a read or write operation. A 12-bit status word is always available from the connected peripheral controller. The 3000 interface can read in the status word any time data is not being read in.

4. Read or write operation

   Transmit or read in a series of 12-bit data bytes.

5. Read controller status word

   Status can be checked at the end of an operation to determine the reason for termination.

PROCEDURES

1. Connect peripheral controller
   a. Load lower 12 bits of A register, with connect code for controller to be activated.
   b. Generate parity bit for connect code and deposit in A register, bit 3.
   c. Set connect bit (A register, bit 1).
   d. Output-from-A to NOC-5.
   e. Wait for reply bit (NIC-5, bit 1).
      1) When reply bit = 1, go to step 1f.
      2) Absence of reply indicates that no controller connected. Repeat connect operation or go to an error routine.
   f. Clear connect bit (NOC-5, bit 1). The connect code can also be dropped.

NOTE

While the connect bit (NOC-5, bit 1) is 1, a 12-bit status word from the connected controller is available on NIC-5.

2. Issue function codes

   a. Load lower 12 bits of A register with function code.

   b. Generate parity bit for function code and deposit in A register, bit 3.

   c. Set function bit (A register, bit 0).

   d. Output-from-A to NOC-5.

   e. Wait for response status bit (NIC-7, bit 0).

      1) Response bit = 1 indicates that the peripheral controller has returned either a reply or reject. Go to step 2f.

      2) No response indicates that the peripheral controller did not accept the function code. Repeat or go to error routine.

   f. Check to see which response-reply (NIC-5, bit 2) or reject (NIC-5, bit 1) was returned.

      1) If reply = 1, clear function bit (NOC-5, bit 0). Return to step 2a if another function code is required or go on to the status, read, or write sequence. (Reply = 1 indicates that the peripheral controller accepted the function code.)

      2) If reject = 1, clear function bit (NOC-5, bit 0). Return to step 2a to repeat function code or go to error routine. Reject = 1 indicates that the peripheral controller could not accept the function code.

NOTE

While the function bit (NOC-5, bit 0) is 1, a 12-bit controller status word is available on NIC-5.

3. Write operation

   a. Set write bit (NOC-7, bit C_{16}).

   b. Load data byte into lower 12 bits of A register.

   c. Set data signal bit in A register, bit 2.

   d. Generate odd parity bit for data byte and load into A register, bit 3.

e. Output-from-A to NOC-5.

(This transmits the data byte to the peripheral controller.)

f. Wait for reply bit (NIC-5, bit 1).

(Reply = 1 indicates that the controller has accepted the data byte.)

g. Clear data signal bit (NOC-5, bit 2) when reply = 1.

(Reply will drop when data signal is cleared.)

h. Return to step 3b if another output byte is required. Otherwise go to step 3i.

i. Clear write bit (NOC-7, bit C_{16}).

4. Read operation

a. Set read bit (NOC-7, bit 14).

b. Set data signal bit (NOC-5, bit 2).

(This requests that the controller send a data byte.)

c. Wait for response status bit (NIC-7, bit 0).

(Response = 1 indicates that an input byte is available on NIC-5.)

d. Input-to-A from NIC-5.

e. Check end-of-record bit (NIC-5, bit 0) to see if this is last byte in the record.

1) If end of record = 0, clear data signal bit (NOC-5, bit 2), and

a) Return to step 4b if another input word is required†, or

b) To terminate the read operation, go to step 4f.

2) If end of record = 1, clear data signal bit (NOC-5, bit 2).

Then, go to step 4f.

f. Clear read bit (NOC-7, bit E_{16}).

(This terminates the read operation.)

---

† An odd parity bit accompanies the input byte, however, the 3000 interface logic does not automatically check parity. If a parity check is desired, it is up to the driver software to analyze the input byte and parity bit (A register, bit 3) before the next byte is read in.

5. Read controller status word

   A 12-bit status word is available from the connected peripheral controller at all times.

   a. Set status select bit (NOC-7, bit 0).

   b. Input-to-A from NIC-4.

      (The 12-bit status word is in the lower 12 bits of the A register.)

   c. Clear status select bit.

854 INTERFACE

The 854 interface controls data transfers between the 854 disk units and the SCU memory. The interface controls up to three units but can transfer data to only one unit at a time. Software supports only three units per interface. The interface is used only with the maintenance control units. Figure 3-12 shows the relationship between the interface and system elements with which it communicates.

Operation of the interface is controlled by the SCU through bits on normal channels 5, 6, and 7. Principle control responsibilities of the SCU in relation to the interface are as follows:

- Selects unit, cylinder, head, and sector.
- Controls read and write operations.
- Monitors status conditions.

DISK STORAGE DRIVE DESCRIPTION

The 854 disk storage drive is a high-speed, random access, disk storage device. Data is recorded on removable disk packs. Three 854 units are used with the 854 interface in the maintenance station.

Each unit contains six disks mounted on a common vertical spindle. Recording is done on only 10 surfaces of the six disks. The top side of the upper disk and bottom side of the lower disk are not used. The access mechanism for each unit consists of 10 arms mounted on a movable carriage. A read/write head is mounted on the end of each arm.

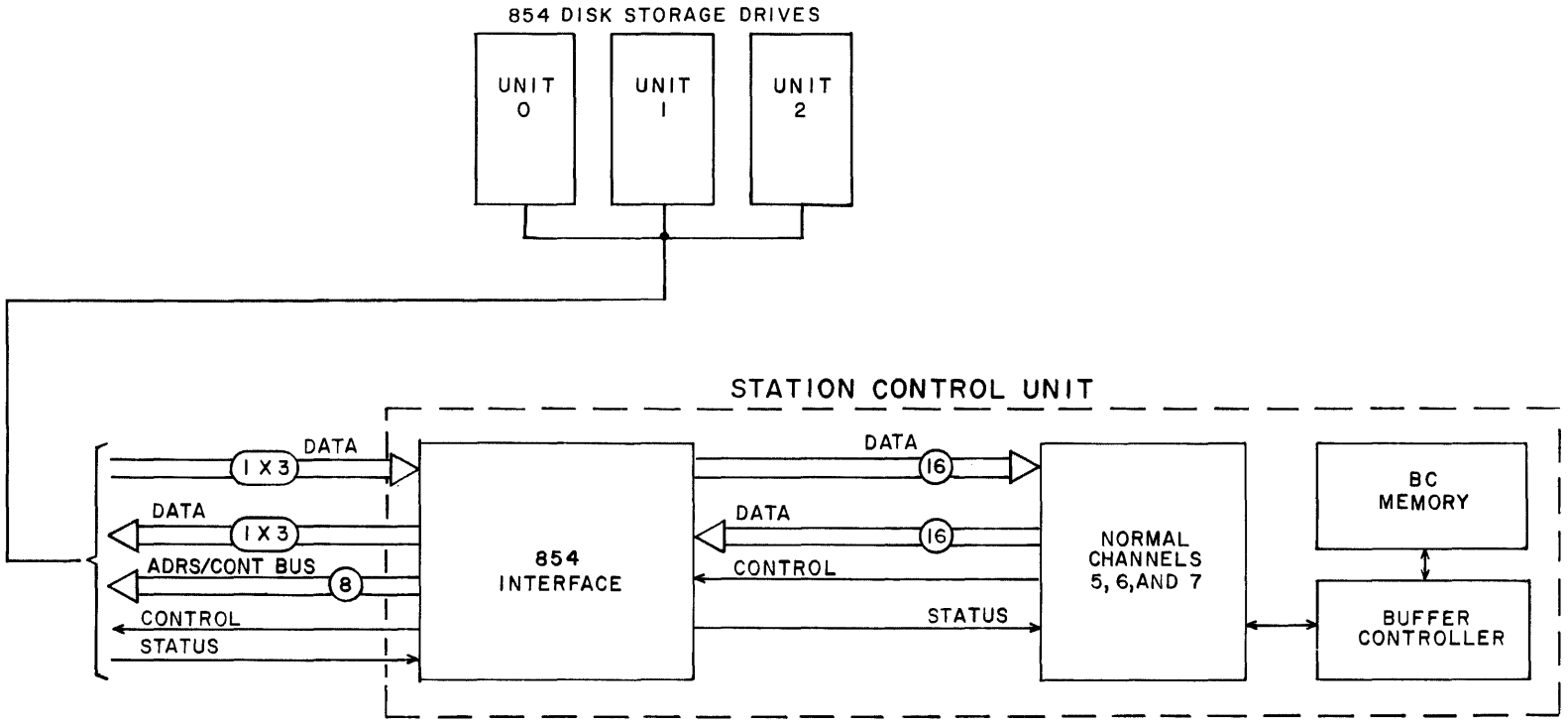


Figure 3-12. 854 Interface System Relationship

An initial seek operation is performed by loading a disk pack, closing the top cover, and pressing the START switch. The carriage then moves horizontally from a retracted position and positions the heads at the outermost track (000).

The 10 read/write heads are aligned in a vertical line to enable access to corresponding circular tracks on the 10 disk surfaces. The 10 aligned circular tracks are called a cylinder. The access mechanism can be stopped at any one of 203 track positions. This provides 203 cylinders of data or 2030 data tracks for each unit. The 10 tracks on a cylinder and associated read/write heads are numbered 0 to 9 from top to bottom. The cylinders are numbered 000 to 202 from outside ring to inside ring.

Data is read or written on consecutive even sectors from track 0 to track 9 and then on consecutive odd sectors from track 0 to track 9 for each cylinder. Electronic switching from one head to the next provides minimum selection time.

854 SECTOR RECORDING

GENERAL

Each disk surface of a unit is divided into 16 equal sized sectors by 16 sector marks numbered 0 through 15. The sector is the smallest addressable unit on the disk. All read and write operations must start at the beginning of a sector. The unit generates a sector mark signal each time a mark is sensed on the disk. The disk also has an index mark to provide a track reference point for the sector marks. Each sector contains 104 16-bit data words plus the associated address tagging information.

The unit records information serially bit by bit on the disk. The format for sector recording is shown in Figure 3-13.

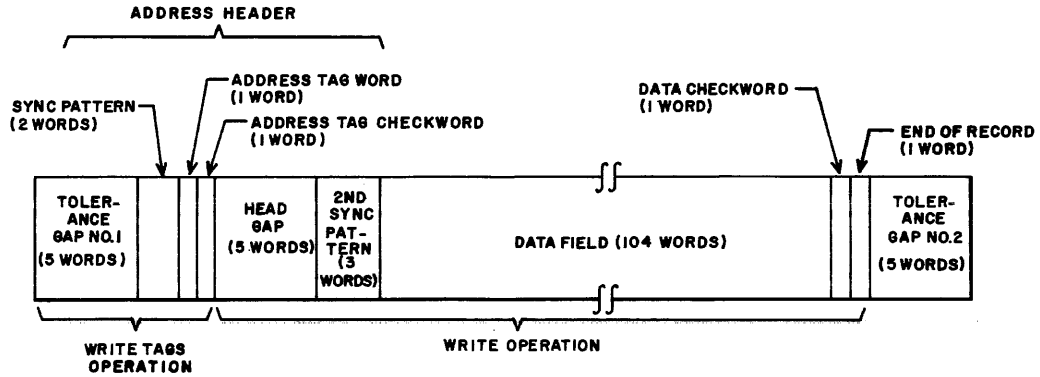


Figure 3-13. 854 Sector Recording

TOLERANCE GAP NUMBER 1

This is a block of all 1's to allow for gap between erase and write heads. A minimum of four words must be written with the erase gate set before writing the sync pattern. This ensures that the disk area is erased prior to writing the sync pattern.

SYNC PATTERN

The sync pattern contains 32 bits and consists of all 0's with a single 1-bit at the end. The 1-bit (sync bit) indicates the beginning of address tag information.

ADDRESS TAG WORD

The address tag word designates the area on the disk to be accessed. Each sector contains a unique address tag word that distinguishes it from all other sectors within a disk pack. Figure 3-14 shows a detailed breakdown of the address tag word. Bits 2^8 through 2^{15} designate one of 203 physical cylinders numbered 0 through 202. Bits 2^0 through 2^7 designate a logical sector instead of the physical location of the head and sector. The software must convert the logical sector into a physical head and sector when the address tag word is read from the disk. Logical sector addressing is necessary because the software uses a half-tracking method of addressing that enables only every other sector to be read from the disk. First, the sectors in even numbered physical locations of a cylinder track are read or written by head 0 during a revolution of the disk. After the revolution is completed, head 1 reads or writes the even numbered sectors of the next cylinder track. After each revolution, the head is incremented until all even numbered sectors on the 10 cylinder tracks have been read or written. When the last even numbered sector is read or written by head 9, head selection returns to head 0. The odd numbered sectors of the cylinder are then read or written in the same manner as the even numbered sectors.

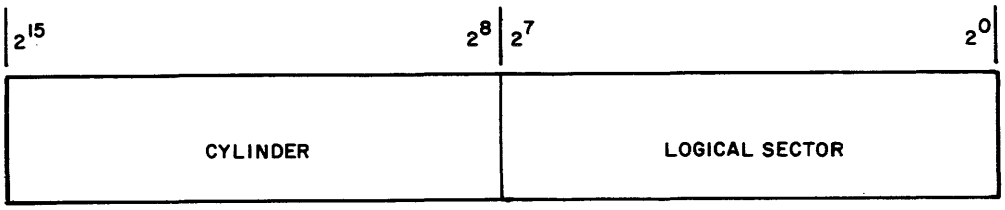


Figure 3-14. Address Tag Word

The sectors in even numbered physical locations have consecutive logical addresses from 0000_{16} through $004F_{16}$ while odd sectors have consecutive addresses from 0050_{16} through $009F_{16}$. The logical sector addresses in relation to the physical locations of the heads and sectors are as follows:

Head Number	Logical Sector Addresses For Even Physical Sectors (0, 2, 4, 6, 8, 10, 12, and 14)	Logical Sector Addresses For Odd Physical Sectors (1, 3, 5, 7, 9, 11, 13, and 15)
0	0000_{16} through 0007_{16}	0050_{16} through 0057_{16}
1	0008_{16} through $000F_{16}$	0058_{16} through $005F_{16}$
2	0010_{16} through 0017_{16}	0060_{16} through 0067_{16}
3	0018_{16} through $001F_{16}$	0068_{16} through $006F_{16}$
4	0020_{16} through 0027_{16}	0070_{16} through 0077_{16}
5	0028_{16} through $002F_{16}$	0078_{16} through $007F_{16}$
6	0030_{16} through 0037_{16}	0080_{16} through 0087_{16}
7	0038_{16} through $003F_{16}$	0088_{16} through $008F_{16}$
8	0040_{16} through 0047_{16}	0090_{16} through 0097_{16}
9	0048_{16} through $004F_{16}$	0098_{16} through $009F_{16}$

ADDRESS TAG CHECKWORD

The checkword is used for detecting errors in the transmission of the address tag. It is generated by the software and written on the disk immediately after the address tag word when address headers are initially written. The checkword is written as the complement of the address tag word and is verified by the software when it is read from the disk.

HEAD GAP

The head gap is a block of all 1's which is equivalent to the physical distance of the read/write head to erase head gap on the disk.

SECOND SYNC PATTERN

The second sync pattern contains 48 bits and consists of all 0's with a single 1-bit at the end. The 1-bit (sync bit) indicates the beginning of the data field.

DATA FIELD

The data field consists of 104 16-bit words.

DATA CHECKWORD

This checkword is generated by the software from the data field and written on the disk immediately after the data field. It is used for detecting errors in the transmission of the data field. During a read operation the software does a half add of the data read from the disk and compares the result with the checkword read from the disk. If the checkword equals all zeros, the software generates a $00FF_{16}$ code for the checkword.

END OF RECORD

The end of record contains 15 0-bits followed by a 1-bit. It is written after the data checkword before the write operation terminates. This word is not verified on a read operation.

TOLERANCE GAP NUMBER 2

This is a block of all 0's to allow for the physical distance of the read/write to erase gap and spindle speed and oscillator tolerances.

WRITE TAGS OPERATION

The write tags operation is a one-time operation used to write address tag information at the beginning of all sectors in a disk pack. (Refer to Figure 3-13 for the sector recording format.) These tags are read as addresses during read and write operations.

The cylinder portion of the address tag word corresponds to the actual cylinder number of the disk. The software must convert the logical sector portion of the address tag word into a physical head and sector number.

Initially, the software must perform the following steps.

1. Connect the 854 interface through the maintenance station normal channel switch interface.
2. Select an 854 unit.
3. Initiate a seek operation which positions the read/write head assembly to the proper cylinder.
4. Select the read/write head by converting the logical sector address into a physical head and sector number.
5. Select a write operation.

The software then waits for a sector mark signal from the unit. Following the sector mark signal, the software transfers the 80 bits of tolerance gap no. 1 through the interface to the unit. A minimum of 64 bits must be transferred with the erase gate set.

The address header information, beginning with the sync pattern, is placed on the normal channels in 16-bit data words by the software. As each 16-bit data word is written from the interface to the sector, a flag status is returned to the software. The software then has 6.2 microseconds to place another 16-bit word on the normal channels. After the sync pattern, address tag word, and address tag checkword have been written, the software waits for the next sector mark signal and them times out the associated sector. Tags are then written in the next sector (third) when the associated sector mark is detected. In this manner, tags are written consecutively in all even sectors and then in all odd sectors of a cylinder.

DATA TRANSFER OPERATIONS

GENERAL

The SCU initiates and selects all operations through the normal channels. Data transfers take place between the disk units and the BC memory in the SCU via the normal channels.

Timing for the 854 interface is provided by the software.

INITIAL PROCEDURE

Before a data transfer takes place, the software must perform the following tasks.

- Connects the 854 interface through the maintenance station normal channel switch interface.
- Selects a particular 854 unit.
- Initiates a seek operation which positions the read/write head assembly to the proper cylinder.
- Selects the proper sector on the disk.
- Selects the proper read/write head and sector on the disk by converting the logical sector address into a physical head and sector number.
- Selects a write or read operation.

The software now waits for the leading edge of the on-sector signal from the interface before reading the address tag. Following the on-sector signal, the software times out the 80 bits of tolerance gap no. 1 and then returns a zeros search signal to the interface. (The zeros search signal enables the flag status bit to set when the address tag word is read.)

The interface then reads the sync pattern from the disk in search for the sync bit. The software ignores header information until the flag status bit sets.

After the sync bit is detected, the address tag word is read from the sector and the flag status sets. The address tag word is then sent to the software over the normal channels. The software does a compare between the tag word and the cylinder, head, and sector selected by the software. If the compare fails, an address error occurs and the program should not proceed with a read or write operation.

Following the address tag word, the address check code is read from the sector. The software then checks that the code is the complement of the tag word. If the check fails, an address error again occurs.

WRITE OPERATION

A selected write operation occurs immediately following the events previously described in Initial Procedure.

The software transfers the head gap, second sync pattern, and data field to the interface in 16-bit words. This information is written in serial fashion after the address tag checkword in the sector. As each 16-bit word is transferred from the interface to the sector, a flag status is returned to the software. Each time flag status comes up, the software must place another 16-bit word on the normal channels within 6.2 microseconds.

After the last word of the data field is written in the sector, a software-generated data checkword is written after the data field. The interface then writes the end of record after the data checkword.

READ OPERATION

A selected read operation occurs immediately following the events previously described in Initial Procedure.

The software then times out the 80 bits of the head gap and sends a zeros search signal to the interface. The interface then reads the second sync pattern from the disk in search for the sync bit.

After the interface detects the sync bit, it reads the first data word from the sector. When the 16-bit word is assembled in the interface, a flag status is sent to the software. Each time a word is read and flag status comes up, the software must transfer the word on the normal channels within 6.2 microseconds.

After the data field is read, the data checkword is read from the sector. The software compares the checkword with the data field just read. If the compare fails, a read error results.

854 INTERFACE NORMAL CHANNEL SIGNALS

Tables 3-7 and 3-8 describe the normal input and output channel bits used with the 854 interface. Bit assignments in the table are for the STAR-100 maintenance control unit. Refer to appendix A of this manual for a complete list of normal channel bit assignments for all stations.

TABLE 3-7. NIC-6A, -7 BIT DESCRIPTIONS FOR 854 INTERFACE

Channel	Bit	Name	Function
NIC-6A	0(2^{15}) 7(2^{8})	Selected units 0 through 7	These status bits from the units are used to check if more than one unit is selected.
NIC-6A	8(2^{7})	Flag	During a read operation this bit indicates that a 16-bit data word, data checkword, or a header address word is available on NIC-7. The BC must read in the word within 6.2 microseconds. The flag bit status clears normally after 6.2 microseconds from the time it set but may be cleared early by setting the reset flag bit (NOC-6A, bit 8). During a write operation this bit indicates that the interface is ready to accept a 16-bit data word, checkword, or header address word, on NOC-7. The BC must supply a data word within 6.2 microseconds, after flag status comes up.
NIC-6A	9(2^{6})	Sector mark	This status bit indicates detection of a sector reference mark from the selected unit. The bit remains set for 60 microseconds.
NIC-6A	A(2^{5})	Index mark	This status bit indicates detection of an index reference mark from the selected unit. The bit remains set for 60 microseconds, occurs once per revolution of the disk, and is in sync with sector mark status for sector 15.

TABLE 3-7. NIC-6A, -7 BIT DESCRIPTIONS FOR 854 INTERFACE (Cont'd)

Channel	Bit	Name	Function
NIC-6A	B(2^4)	Ready	This status bit from the units indicates that the unit selected by unit select 2^0, 2^1, and 2^2 bits (NOC-6A) is available. Not ready conditions are: 1. Heads not loaded and motor not up to speed. 2. A disk pack not in the unit. 3. The unit ON LINE/OFF LINE switch in OFF LINE position.
NIC-6A	C(2^3)	Seek error	This status bit from the units indicates that a direct seek in the selected unit has gone beyond the cylinder limits.
NIC-6A	D(2^2)	On cylinder	This status bit from the units indicates that the selected unit has positioned the read/write heads at the addressed cylinder.
NIC-6A	E(2^1)	On sector	This status bit from the units indicates that the addressed sector of the selected unit is one sector away from the recording head. The on sector status is a 1-microsecond pulse.
NIC-6A	F(2^0)	Fault	This status bit from the units indicates that the selected unit has a fault condition. The fault condition must be manually cleared.
NIC-7	0(2^{15}) F(2^0)	Input data bits 0 through F	These bits comprise the 16-bit data word that is read serially from the disk and assembled in the interface.

TABLE 3-8. NOC-5, -6A, -7 BIT DESCRIPTIONS FOR 854 INTERFACE

Channel	Bit	Name	Function
NOC-5	$0(2^{15})$ ↓ $7(2^{8})$	Not used	
NOC-5	$8(2^{7})$	Channel switch† 2^{1}	This bit is used in conjunction with channel switch 2^{0} bit (NOC-5) to select 854 interface, printer interface, or coupler B for transfer of NIC-6 and NOC-6 bits. If both bits are clear, NIC-6A and NOC-6A (854 interface) are selected.
NOC-5	$9(2^{6})$	Channel switch† 2^{0}	Refer to channel switch 2^{1} bit (NOC-5).
NOC-5	$A(2^{5})$	Not used	
NOC-5	$B(2^{4})$	Not used	
NOC-5	$C(2^{3})$	Head select	This bit is sent to the selected unit and indicates that the address and control bits 0 through 3 (NOC-6A, bits 7 through 4) contain the head address.
NOC-5	$D(2^{2})$	Difference select	This bit is sent to the selected unit and indicates that the address and control bits 0 through 7 (NOC-6A, bits 7 through 0) contain positioning information (difference between unit's present cylinder address and new cylinder address.)
NOC-5	$E(2^{1})$	Control select	This bit is sent to the selected unit and indicates that the address and control bits 0 through 7 (NOC-6A, bits 7 through 0) contain control information.
NOC-5	$F(2^{0})$	Sector select	This bit is sent to the selected unit and indicates that the address and control bits 0 through 3 (NOC-6A, bits 7 through 4) contain the sector address.

† These two bits are inputs to the normal channel switch interface and are given here for reference.

TABLE 3-8. NOC-5, -6A, -7 BIT DESCRIPTIONS FOR 854 INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-6A	$0(2^{15})$ ↓ $7(2^{8})$	Address and control bits 7 through 0	These bits are sent to the selected unit. Information in these bits is determined by the accompaning head select, difference select, control select, or sector select bit (NOC-5). Information on the address and control bits is given in Table 3-9.
NOC-6A	$8(2^{7})$	Reset flag	This bit clears the flag status bit (NIC-6A) in the interface.
NOC-6A	$9(2^{6})$	Hold sector mark	This bit must be held at a constant 0 to enable the sector mark status bit (NIC-6A) to clear.
NOC-6A	A	Hold index mark	This bit is held at a constant 0 to enable the index mark status bit (NIC-6A) to clear.
NOC-6A	$B(2^{4})$	Write	This bit when set signals the interface to write data onto the disk. This bit when clear enables the interface to read data from the disk.
NOC-6A	$C(2^{3})$	Zeros search	This bit is set when address tag information is being read from the sector. It enables the flag status bit (NIC-6A) to set after the sync bit is read.
NOC-6A	$D(2^{2})$ ↓ $F(2^{1})$	Unit select 2^{2} through 2^{0}	These three bits provide a code that enables the interface to select one of the units. The code must be held to keep the unit selected.
NOC-7	$0(2^{15})$ ↓ $F(2^{0})$	Output data bits 0 through F	These bits comprise the 16-bit data word that is transferred serially from the interface shift register and written on the disk. The data is written in complement form. These bits must be all 1's during a read operation.

ADDRESS AND CONTROL BUS

The address and control bus in the interface consists of eight lines which transmit normal channel information to the selected unit. Information on these lines is determined by an accompaning normal channel tag line signal. Table 3-9 gives the relationship of the address and control bus and the tag line signals.

TABLE 3-9. ADDRESS AND CONTROL BUS FUNCTIONS

Address/ Control Bus	Tag Line			
	Difference Select	Sector Select	Head Select	Control Select
Bit 0 (NOC-6A, bit 7)	1	1	1	Write gate; a 1 input on this line enables the write drivers.
Bit 1 (NOC-6A, bit 6)	2	2	2	Read gate; a 1 input on this line enables the digital read data lines.
Bit 2 (NOC-6A, bit 5)	4	4	4	Seek forward; a 1 input on this line initiates forward carriage movement.
Bit 3 (NOC-6A, bit 4)	8	8	8	Not used
Bit 4 (NOC-6A, bit 3)	16	Not used	Not used	Erase gate; a 1 input on this line enables the erase driver to pass current through the head erase coil.
Bit 5 (NOC-6A, bit 2)	32	Not used	Not used	Seek reverse; a 1 input this line initiates reverse carriage movement.

TABLE 3-9. ADDRESS AND CONTROL BUS FUNCTIONS (Cont'd)

Address/ Control Bus	Difference Select	Sector Select	Head Select	Control Select
Bit 6 (NOC-6A, bit 1)	64	Not used	Not used	Return to zero; a 1 input on this line initiates carriage movement to cylinder 00.
Bit 7 (NOC-6A, bit 0)	128	Not used	Not used	Not used.

PROGRAMMING SEQUENCES

The following procedures are to be used as an aid to the programmer during the preparation of the program for conducting data transfers between the SCU BC and disk unit via the 854 Interface.

CONNECTION PROCEDURE

1. Set appropriate unit select bits (NOC-6A, bits D through F) to connect desired unit to interface.
2. Check that selected unit status bit (NIC-6A, bits 0 through 7) corresponds with the unit selected in step 1 to insure unit has power.
3. Check that ready status bit (NIC-6A, bit B) is set to insure that the read/write heads are loaded.

CYLINDER POSITIONING

1. Calculate difference between present cylinder address and new cylinder address. Set this difference in address and control bits (NOC-6A, bits 0 through 7).
2. Set and clear difference select bit (NOC-5, bit D).
3. Clear address and control bits (NOC-6A, bits 0 through 7).
4. Set seek reverse or seek forward bit (NOC-6A, bit 2 or bit 5).

5. Set and clear control select bit (NOC-5, bit E) to initiate seek.
6. Clear seek reverse or seek forward bit (NOC-6A, bit 2 or bit 5).
7. Wait for on cylinder status (NIC-6A, bit D) to set before proceeding.

SECTOR AND HEAD SELECTION

The following steps are performed prior to a read, write, or write tags operation.

1. Translate the logical sector address to determine the physical head and sector numbers.
2. Place physical head number in address and control bits (NOC-6A, bits 7 through 4).
3. Set and clear head select bit (NOC-5, bit C).
4. Clear address and control bits (NOC-6A, bits 7 through 4).
5. Place physical sector number in address and control bits (NOC-6A, bits 7 through 4).
6. Set and clear sector select bit (NOC-5, bit F).
7. Clear address and control bits (NOC-6A, bits 7 through 4).
8. Wait for leading and trailing edges of on sector status (NIC-6A, bit E).
9. Wait for leading edge of sector mark status (NIC-6A, bit 9) before proceeding.

READ AND VERIFY ADDRESS TAG

The following steps are performed prior to a read or write operation but are skipped for a write tags operation.

1. Wait 45 microseconds to allow for gap between read/write and erase heads.
2. Place all 1's in NOC-7.
3. Set and clear zeros search bit (NOC-6A, bit C).
4. Wait for leading edge of flag status (NIC-6A, bit 8).

NOTE

A timeout of each flag bit (6.2 microseconds) should be included in the program to eliminate the possibility of the program hanging up while waiting for the flag status to occur.

5. Input and verify the address tag on NIC-7, bits 0 through F.
6. Wait for the leading edge of next flag status (NIC-6A, bit 8).
7. Input and verify the address tag checkword (complement of address tag) on NIC-7, bits 0 through F.
8. If next operation is a read operation, leave all 1's in NOC-7. For a write operation, clear all bits of NOC-7.

READ OPERATION

1. Wait 45 microseconds to allow for the gap between read/write and erase heads.
2. Set and clear zeros search bit (NOC-6A, bit C).
3. Wait for leading edge of flag status (NIC-6A, bit 8).
4. Input and store 104 data words and one data checkword on NIC-7, bits 0 through F. Each word is read on the leading edge of the flag status (NIC-6A, bit 8) which occurs every 12.4 microseconds.
5. Verify that the half added sum of the data equals the data checkword shifted left around eight places.
6. If in half tracking mode, return to step 9 of Sector and Head Selection within 1.6 milliseconds and repeat the procedure.

WRITE OPERATION

1. Set write bit (NOC-6A, bit B).
2. Set erase gate bit (NOC-6A, bit 3).
3. Set write gate bit (NOC-6A, bit 7).
4. Set control select bit (NOC-5, bit E).
5. Wait for leading edge of flag status (NIC-6A, bit 8).
6. Output the following data on NOC-7, bits 0 through F.

5 words of all 1's (head gap)
3 words of all 0's, except last bit a 1 (second sync pattern)
104 words of data (data field)
1 word (data checkword)
1 word of all 0's except last bit a 1 (end of record)

Each output word must be conditioned by the leading edge of flag status (NIC-6A, bit 8).

7. Clear write bit (NOC-6A, bit B), erase gate bit (NOC-6A, bit 3). Write gate bit (NOC-6A, bit 7), and control select bit (NOC-5, bit E).

8. If in half tracking mode, return to step 9 of Sector and Head Selection within 1.6 milliseconds and repeat the procedure.

WRITE TAG OPERATION

Repeat the steps in the Write Operation, except in step 6 output the following data on NOC-7, bits 0 through F.

5 words of all 1's (tolerance gap no. 1)
2 words of all 0's, except last bit a 1 (sync pattern)
1 word (address tag)
1 word (address tag checkword, complement of address tag)

STAR-100 MAINTENANCE INTERFACE

NOTE

Since the maintenance, SAC coupler, and control interfaces of the maintenance control unit operate in such interlocking detail with the STAR-100 Computer, it is imperative that the reader be familiar with STAR-100 operation and its programming conventions to fully understand the following material.

The maintenance interface in the maintenance control unit acts as a special control area for the multiple access lines between the maintenance control unit and a STAR-100 Computer mainframe (refer to Figure 3-15). Normal input and output channels 8 through F receive status information from the computer and carry function commands to the computer. Normal input and output channels 0 through 7 carry information between the various peripheral devices and the SCU.

The BC in the SCU controls the flow of function commands to the computer and orders status checks to be made. Normal channel bit definitions for the maintenance interface are given in Tables 3-10 and 3-11.

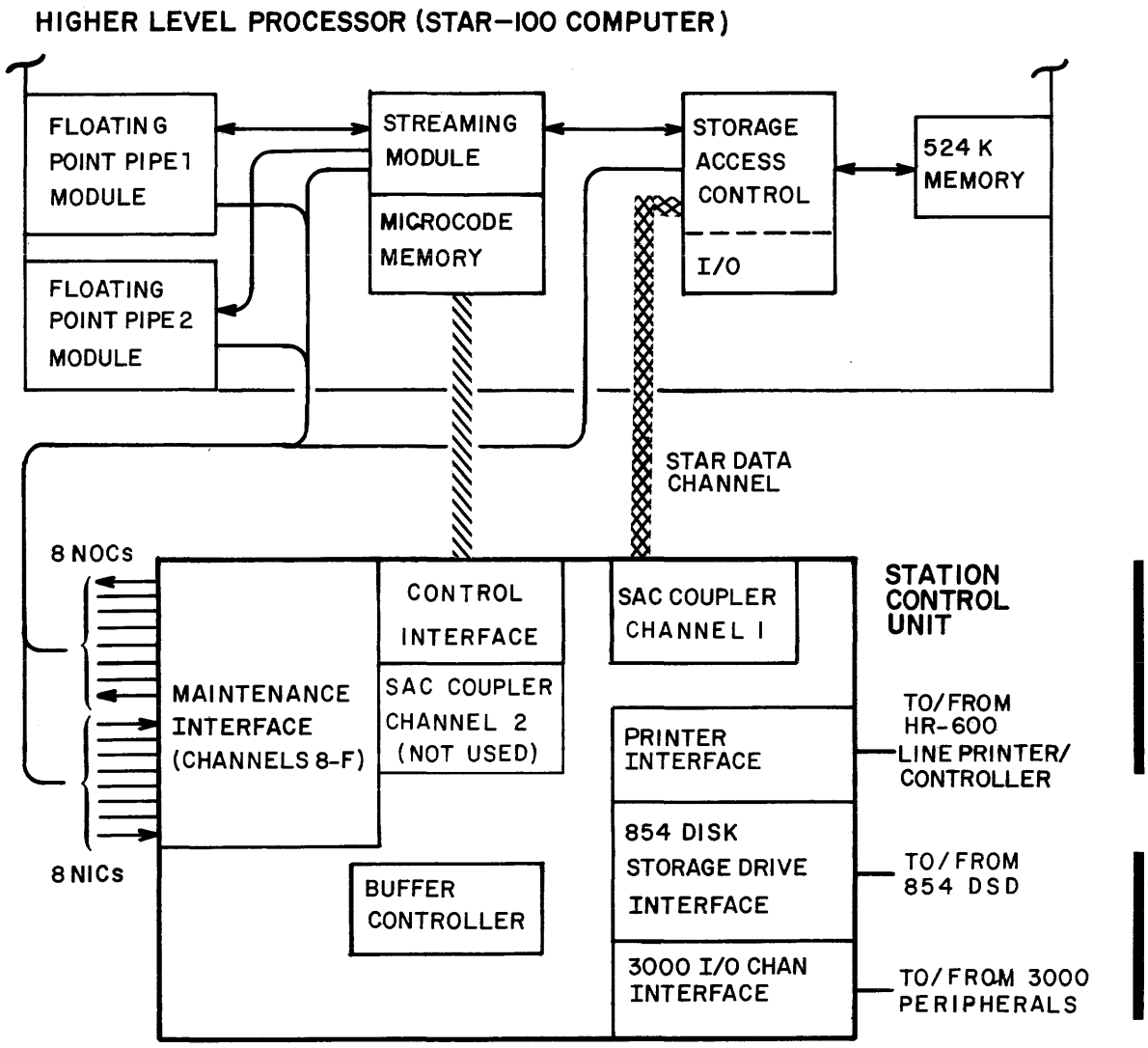


Figure 3-15. Maintenance Control Unit/STAR-100 Computer - System Relationship

TABLE 3-10. NIC-8 THROUGH NIC-F BIT DESCRIPTIONS FOR MAINTENANCE INTERFACE

Channel	Bit	Name	Function
NIC-8 (CPU chan ATB1)	0 ↓ F	CIAR bit 00 ↓ ↓ ↓ 15	Current instruction address register; these bits represent the microcode address of the instruction currently in the P address register.
NIC-9 (CPU chan ATB2)	0 ↓ F	CIAR bit 16 ↓ ↓ ↓ 31	
NIC-A (CPU chan ATB3)	0 ↓ F	CIAR bit 32 ↓ ↓ ↓ 47	
NIC-B (CPU chan ATB4)	0 ↓ F	Display register bit 00 ↓ ↓ ↓ 15	These bits represent the contents of a CPU register selected by 4-bit code generated in the maintenance interface.
NIC-C (CPU chan ATB5)	0 ↓ F	Display register bit 16 ↓ ↓ ↓ 31	
NIC-D (CPU chan ATB6)	0 ↓ F	Display register bit 32 ↓ ↓ ↓ 47	
NIC-E (CPU chan ATB7)	0 ↓ F	Display register bit 48 ↓ ↓ ↓ 63	
NIC-F (CPU chan ATB8)	0	Memory parity fault	This bit indicates a parity error occurred in central memory of the computer.
NIC-F (CPU chan ATB8)	1	MIC memory parity fault	This bit indicates a parity fault occurred in the microcode memory.

TABLE 3-10. NIC-8 THROUGH NIC-F BIT DESCRIPTIONS FOR MAINTENANCE INTERFACE (Cont'd)

Channel	Bit	Name	Function
NIC-F (CPU chan ATB8)	2	Multiple match	This bit indicates that a CPU fault in a lock-virtual page address combination occurred in the page table.
NIC-F (CPU chan ATB8)	3	Absolute sword bounds hit	This bit indicates that the memory reference the maintenance interface has made is within the limits of a specified block of memory.
NIC-F (CPU chan ATB8)	4	Event stop	This bit indicates that the CPU stopped in response to the stop counter A or B lines being set previously.
NIC-F	5	Not used	
NIC-F	6	Not used	
NIC-F (CPU chan ATB8)	7	Monitor mode	This bit indicates that CPU software is operating in the monitor mode.
NIC-F	8	Temperature/dewpoint alarm	This bit indicates that either the temperature or relative humidity in the computer have exceeded allowable limits.
NIC-F	9	MG1 power failure	This bit indicates that 60 Hz source power to the primary MG set has failed.
NIC-F	A	Section power fail	This bit indicates that a particular section, but not all, of the computer has experienced a power failure.
NIC-F	B	System power margin (+5%)	This bit indicates that system power levels have exceeded the positive tolerance figure.

TABLE 3-10. NIC-8 THROUGH NIC-F BIT DESCRIPTIONS FOR MAINTENANCE INTERFACE (Cont'd)

Channel	Bit	Name	Function
NIC-F	C	System power margin (-5%)	This bit indicates that system power levels have become more negative than allowable.
NIC-F	D	MG2 power failure	This bit indicates that 60 Hz source power to the standby MG set has failed.
NIC-F	E	CPU idle	This bit indicates that the computer's CPU is in the idle state.
NIC-F	F	CPU stopped	This bit indicates that the computer's CPU is stopped due to some abnormal condition or due to some external influence (such as the maintenance station issuing a bit to stop the CPU under specified conditions).

TABLE 3-11. NOC-8 THROUGH NOC-F BIT DESCRIPTIONS FOR MAINTENANCE INTERFACE

Channel	Bit	Name	Function
NOC-8 (CPU chan BTA1)	0	Master clear	This signal is sent to SAC and central memory modules of the computer. It clears the SAC I/O channels and the devices connected to those I/O channels.
NOC-8	1	Stop	Issuing this signal will stop the computer before the next instruction is issued.

TABLE 3-11. NOC-8 THROUGH NOC-F BIT DESCRIPTIONS FOR MAINTENANCE INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-8	2	Step†	Issuing this signal permits the execution of one instruction. (Any faults must be cleared before the computer can be stopped.)
NOC-8	3	Run†	Issuing this signal permits starting the computer from a manual stop or fault stop. (Faults must be cleared before computer can be started.)
NOC-8	4	Store associative register†	Issuing this signal permits storing the associative registers and register file in CPU central memory. Associative registers are stored starting at absolute address 4000_{16}. The register file is stored starting at absolute address 0000_{16} in monitor mode and at virtual address 0000_{16} in job mode. After this operation both must be reloaded by executing a load associative register command.
NOC-8	5	Load associative register†	Issuing this signal permits loading the associative registers and register file from CPU central memory. Associative registers are loaded starting from absolute address 4000_{16}. The register file is loaded starting at absolute address 0000_{16} in monitor mode and at virtual address 0000_{16} in job mode.

† Computer must be stopped before issuing these commands.

TABLE 3-11. NOC-8 THROUGH NOC-F BIT DESCRIPTIONS FOR MAINTENANCE INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-8 (CPU chan BTA1)	6	Stream/FP master clear	Issuing this signal sends a master clear to the streaming and floating point modules of the computer. SAC and central memory modules are not included. This signal must be set at minimum of 1 microsecond.
NOC-8	7	Clear faults	Issuing this signal clears the following conditions and allows the computer to be restarted with a run signal (NOC-8, bit 3). 1. Memory parity fault 2. Microcode memory parity fault 3. Multiple match 4. Absolute sword bounds hit 5. Parity fault address register and bounds hit address register
NOC-8	8	Not used	
NOC-8	9	Sync	This signal is used by the CPU to gate CPU data back to the maintenance station. When reading display registers, the sync signal must be set after the read signal is set.
NOC-8	A	Not used	
NOC-8	B	Read	This signal transfers the selected register and CIAR into the maintenance station's display register.

TABLE 3-11. NOC-8 THROUGH NOC-F BIT DESCRIPTIONS FOR MAINTENANCE INTERFACE (Cont'd)

<table>
<tr><th>Channel</th><th>Bit</th><th>Name</th><th>Function</th></tr>
<tr><td>NOC-8 ↓</td><td>C
D
E
F</td><td>Register select 2^3
2^2
2^1
2^0</td><td>These bits represent a 4-bit code sent by the maintenance station to the HLP to select one of several CPU registers for inputs to the maintenance interface. (Refer to Table 3-12.)</td></tr>
<tr><td>NOC-9
(CPU chan BTA2) ↓</td><td>0</td><td>Sweep mode</td><td>Issuing this bit causes all instructions to act as passes.</td></tr>
<tr><td></td><td>1</td><td>Interrupt gate</td><td>When this signal is a 1, time interrupts and external interrupts will only be processed between instruction.</td></tr>
<tr><td></td><td>2</td><td>Block instruction execution overlap</td><td>This signal will allow only one register instruction to be execution at any time.</td></tr>
<tr><td></td><td>3</td><td>Channel register select 2^1</td><td rowspan="2">These two bits form a code which permits the selection of any of three CPU registers via NOC-A. The code is:
<table>
<tr><th>bit $3(2^1)$</th><th>bit $4(2^0)$</th><th>CPU Reg. Sel.</th></tr>
<tr><td>0</td><td>0</td><td>None</td></tr>
<tr><td>0</td><td>1</td><td>Channel register A1</td></tr>
<tr><td>1</td><td>0</td><td>Channel register A2</td></tr>
<tr><td>1</td><td>1</td><td>Channel register A3</td></tr>
</table></td></tr>
<tr><td></td><td>4</td><td>Channel register select 2^0</td></tr>
<tr><td></td><td>5</td><td>Clock adjust</td><td>Setting this bit enables adjustment of the CPU clocks.</td></tr>
<tr><td></td><td>6</td><td>Decrease clock frequency</td><td>This bit is used to decrease clock frequency.</td></tr>
</table>

TABLE 3-11. NOC-8 THROUGH NOC-F BIT DESCRIPTIONS FOR MAINTENANCE INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-9 (CPU chan BTA2)	7	Increase clock frequency	This bit used to increase clock frequency.
	8	Delay trailing edge	This bit is used to delay the trailing edge of all of the clocks on the CPU panel which is specified by NOC-9, bits B through F. If bit 8 and bit 9 are set, only the odd or even clocks on a panel are moved depending on bit A.
	9	Delay leading edge	This bit is used to delay the leading edge of all of the clocks on the panel which is specified by NOC-9, bits B through F. If bits 8 and 9 are set, only the odd or even clocks on a panel are moved depending on bit A.
	A	Move clocks	0, move even clocks (refer to description for bit 8 or 9). 1, move odd clocks.
NOC-9 (CPU chan BTA2)	B(2^4) C(2^3) D(2^2) E(2^1) F(2^0)	Panel designators	These bits are used in hex codes to identify panel designators for clock margins. Bit B is the leftmost bit of the designator. They are defined as follows:

Designator Hex Code	CPU Panel(s)
00	All panels
01	All floating point panels
02	All SAC panels
03	All stream and string panels
04	Not used
05	Direct access channel

TABLE 3-11. NOC-8 THROUGH NOC-F BIT DESCRIPTIONS FOR MAINTENANCE INTERFACE (Cont'd)

Channel	Bit	Name	Function		
NOC-9 (CPU chan BTA2)			Designator Hex Code	CPU Panel(s)	
			06	Panel AA	Floating point module
			07	Panel AB	
			08	Panel BA	
			09	Panel BB	
			0A	Panel CA	
			0B	Panel CB	
			0C	Panel DA	
			0D	Panel DB	
			0E	Panel EA	
			0F	Panel EB	
			10	Panel KA	SAC module
			11	Panel KB	
			12	Panel LA	
			13	Panel LB	
			14	Panel NA	
			15	Panel NB	
			16	Panel PA	Stream, string module
			17	Panel PB	
			18	Panel FA	
			19	Panel FB	
			1A	Panel GA	
			1B	Panel GB	
			1C	Panel HA	
			1D	Panel HB	
			1E	Panel JA	
			1F	Panel JB	
NOC-A1	0	Not used			
(CPU chan BTA3-1)	1	Disable channel 1	These bits are transmitted to the SAC module in the computer. Whichever bit is set disables the corresponding channel and no central memory references will be allowed from that channel.		
	2	2			
	3	3			
	4	4			
	5	5			

TABLE 3-11. NOC-8 THROUGH NOC-F BIT DESCRIPTIONS FOR MAINTENANCE INTERFACE (Cont'd)

<table>
<tr><th>Channel</th><th>Bit</th><th>Name</th><th>Function</th></tr>
<tr><td>NOC-A1
(CPU chan BTA3-1)
↓</td><td>6
7
8
9
A
B
C
D
E
F</td><td>Disable channel 6
7
8
9
10
11
12
Not used
Not used
Not used</td><td></td></tr>
<tr><td>NOC-A2
(CPU chan BTA3-2)</td><td>0, 1
2
3</td><td>Select memory group
Delay memory strobe
Advance memory strobe</td><td>For maintenance purposes the memory strobe on CPU memory may be advanced or delayed by setting either bit 2 or 3, as appropriate. Memory is divided into four-section groups (16 banks per group) of which one group is selected for margin tests. The four-section groups correspond to a memory wing. Thus a program can be written to run in a portion of memory not experiencing strobe margin failures and test a portion of memory experiencing strobe margin failures. The code used by bits 0 and 1 to select a memory group is:
<table>
<tr><th>bit 0</th><th>bit 1</th><th></th></tr>
<tr><td>0</td><td>0</td><td>Selects sections MA-MD</td></tr>
<tr><td>0</td><td>1</td><td>Selects sections ME-MH</td></tr>
<tr><td>1</td><td>0</td><td>Selects sections MJ-MM</td></tr>
<tr><td>1</td><td>1</td><td>Selects sections MN-MR</td></tr>
</table>
</td></tr>
</table>

TABLE 3-11. NOC-8 THROUGH NOC-F BIT DESCRIPTIONS FOR MAINTENANCE INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-A2 (CPU chan BTA3-2)	4 5 6 7	Disable memory Parity check	CPU memory is divided into 262K blocks each of which may have its parity check disabled. This procedure facilitates troubleshooting a section of memory while the CPU continues normal job execution with minimum interference. Setting the appropriate bit disables the parity check on memory addresses as follows: Bit 4; addresses 0 through 0 FFF FFF Bit 5; addresses 1 000 000 through 1 FFF FFF Bit 6; addresses 2 000 000 through 2 FFF FFF Bit 7; addresses 3 000 000 through 3 FFF FFF These are absolute addresses and correspond to physical addresses only if no swaps have been made or if phase 16 is in effect.
NOC-A2 (CPU chan BTA3-2)	8	Stop on I/O parity fault	Setting this bit enables the CPU to stop whenever a central memory parity fault is found in data going to I/O logic.
	9	Not used	
	A	Not used	
	B	Not used	
	C	Swap 131K	Setting this bit causes bit 40 of central memory address to be toggled and facilitates testing that part of memory where the memory test is located. The effect is to swap alternate 131K word blocks.

TABLE 3-11. NOC-8 THROUGH NOC-F BIT DESCRIPTIONS FOR MAINTENANCE INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-A2 (CPU chan BTA3-2)	D	Phase 16	Setting this bit, one of the memory degradation lines, causes sequential memory addresses to sweep through 16 banks rather than 32. This is accomplished by interchanging specified memory address bits in computer memory. When phase 16 is set, a time penalty is unavoidable on certain instructions due to the lower order of bank interleaving.
	E	Swap 262K	Setting this bit causes memory address bit 50 to be toggled. If phase 16 bit is clear, alternate 128-word blocks are exchanged. If phase 16 bit is set, the effect is to interchange upper and lower 262K of memory.
	F	Swap 524K	Setting this bit causes memory address bit 38 to be toggled. The effect is to interchange upper and lower 524K.
NOC-A3 (CPU chan BTA3-3)	0	Absolute addressing	Setting this bit forces all central memory references to be absolute addresses.
	1	Test associative registers	Setting this bit toggles bit 30 (2K select) of central memory address when storing the associative registers. The associative registers are then stored at absolute bit address $24,000_{16}$. Therefore, the associative registers may be repeatedly loaded with a fixed pattern and stored in memory for inspection.

TABLE 3-11. NOC-8 THROUGH NOC-F BIT DESCRIPTIONS FOR MAINTENANCE INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-A3 (CPU chan BTA3-3)	2	Test data paths	Setting this bit toggles bit 30 when storing the register file and forces a register file store during an initial load. The register file is stored at absolute bit address $20,000_{16}$ and loaded from address 0. Testing of the data paths to and from central memory is facilitated using this procedure and absolute addressing.
	3	Test invisible package	Setting this bit toggles bit 30 when storing the invisible package. The invisible package is stored 2048_{10} words ahead or behind where it is loaded. Therefore, the test mode can be used starting with a given initial invisible package to see the result of CPU activity on the invisible package.
	4 ↓ F	Not used	
NOC-B (CPU chan BTA4) ↓	0	Not used	These bits are sent to the computer which in turn sends them on to the appropriate device via an I/O channel. They are used to initiate an autoloading sequence in the BC of the maintenance station.
	1	External flag chan 1	
	2	2	
	3	3	
	4	4	
	5	5	
	6	6	
	7	7	
	8	8	
	9	9	
	A	10	
	B	11	
	C	12	

TABLE 3-11. NOC-8 THROUGH NOC-F BIT DESCRIPTION FOR MAINTENANCE INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-B (CPU chan BTA4)	D	Not used	
	E	1 upper bounds	Setting this bit identifies the bits on NOC-C as the upper bounds sword address.
		0 lower bounds	Clearing this bit identifies the bits on NOC-C as the lower bounds sword address.
	F	1 bounds in upper 524K	When bit F is set, this bit identifies the address on NOC-C as being in the upper 524K of memory.
		0 bounds in lower 524K	When bit F is clear, this bit identifies the address on NOC-C as being in the lower 524K of memory.
NOC-C (CPU chan BTA5)	0	Bounds sword Address bit 00	These bits represent either the upper or lower bounds sword address as determined by the state of bit E of NOC-B. Bit 0 of this word is the leftmost bit of the sword address in 524K memory. (Refer to bit F of NOC-B.)
	F	bit 15	
NOC-D (CPU chan BTA6)	0	Check bounds on memory reads	When set, bits 0 through 4 provide the means for the maintenance station to selectively test various categories of requests for inbounds conditions of memory references. Any combination of classes may be selected. If the CPU has been stopped by a bounds hit (bit 4), the maintenance interface must set bit 7 (clear fault) of NOC-8 before the CPU can be restarted. The CPU can then be restarted by
	1	Check bounds on memory writes	
	2	Check bounds on CPU references	
	3	Check bounds on channel references	
	4	Stop CPU on bounds hit	

TABLE 3-11. NOC-8 THROUGH NOC-F BIT DESCRIPTIONS FOR MAINTENANCE INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-D (CPU chan BTA6)			setting bit 3 (run) of NOC-8, and it will execute the next instruction in sequence. To identify a second bounds hit, the maintenance interface must set bit 7 (clear fault) of NOC-8 so that another bounds hit can be detected on NIC-F, bit 3.
	5	Enable bounds check	Setting this bit enables the particular bounds checking bit(s) (bits 0 through 4) to perform their intended function(s). The conditions desired in bits 0 through 4 must be set before setting this bit.
	6	Count A	Setting this bit initiates operation of counter A. The proper counter specification (bit F) and bits 8 through E of this channel must be set up before this count line is enabled.
	7	Count B	Same explanation as counter A but applied to counter B.
	8	Clear counter overflow bits	Setting this bit clears only the upper bit of each 32- or 16-bit counter. If the bit is cleared, both A & B counters are cleared after the read signal (NOC-8, bit B) is received and after both counters are transferred into the maintenance interface's display registers.
	9 A	Stop counter A Stop counter B	Setting either of these bits causes the event incrementing either of the 16-bit counter pairs to stop the CPU. Clear fault (NOC-8, bit 7) clears the condition.

TABLE 3-11. NOC-8 THROUGH NOC-F BIT DESCRIPTIONS FOR MAINTENANCE INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-D (CPU chan BTA6)	B	Carry into A1	These bits are used for reconfiguring CPU counter word size during monitoring functions. Setting bit B enables the carry into counter A1 from counter A2. Clearing both A1 and A2 allows counters to operate as two 16-bit counters. To obtain a 32-bit count, clear A1 and set A2. If both lines of a counter pair are set and NOC-D, bit 8 is set, undefined results occur when the counters are cleared.
	C	Carry into A2	
	D	Carry into B1	
	E	Carry into B2	
NOC-D	F	0 Counter A specifications 1 Counter B specifications	Setting this bit selects counting specifications on NOC-E, bits 0 through F for counter B. Clearing this bit selects counting specifications on NOC-E, bits 0 through F for counter A.
NOC-E (CPU chan BTA7)	0	Counter A1/B1 event Select codes	Setting the desired bit(s) in this group determines which counter pair and event will be selected by the maintenance interface for monitoring purposes. Bits 0 through 4 apply to counter pair A1/B1, and bits 5 through 9 apply to A2/B2. (Refer to Table 3-13).
	9	Counter A2/B2 event Select codes	
	A	Not used	
	B	All jobs enable	Setting this mask bit enables the counters to be incremented for any selected event.

TABLE 3-11. NOC-8 THROUGH NOC-F BIT DESCRIPTIONS FOR MAINTENANCE INTERFACE (Cont'd)

Channel	Bit	Name	Function
NOC-E (CPU chan BTA7)	C	Monitor mode mask	Setting this bit enables counters to be incremented only when the CPU is in monitor mode.
	D	Job mode mask	Setting this masking bit enables a count to be made during CPU selected jobs only.
	E	Data flag bit 56 mask	Setting bit E enables counters to be incremented for sword address on NOC-C and CPU selected functions on NOC-D, bits 0 through F.
	F	Data flag bit 57 mask	Setting bit F enables counters to be incremented for sword address on NOC-C and CPU selected events and masks on NOC-E, bits 0 through F.
NOC-F (CPU chan BTA8)	0	Function select bit 2^7	These 8 bits constitute a function select code which determines the count condition sent to the CPU. If the select code and the corresponding hex code (refer to hex code 12, Table 3-13) bits are equal, wherever there is a 1 in the mask, the counter will be incremented. If the mask contains all zeros, all instructions will be counted. Bit 0 is the leftmost bit of the code.
	1	2^6	
	2	2^5	
	3	2^4	
	4	2^3	
	5	2^2	
	6	2^1	
	7	2^0	
	8	Mask bit 2^7	These 8 bits constitute a mask for the function code used with the function select bits (bits 0 through 7) to define the masking to be used. Bit 8 is the leftmost bit of the code.
	9	2^6	
	A	2^5	
	B	2^4	
	C	2^3	
	D	2^2	
	E	2^1	
	F	2^0	

MAINTENANCE INTERFACE MONITORING ACTIVITIES

The maintenance interface of the maintenance station performs all the system monitoring functions required for computer system operation. The maintenance interface monitors the outputs of its two display registers as its main medium of evaluating system activity, monitors four 16-bit counters in the CPU, and can request microcode memory status and selected CPU status.

DISPLAY REGISTER MONITORING

Two maintenance interface display registers are used for this function; one always contains the output of the current instruction address register (CIAR) in the CPU and the other contains the output of a register selected by a 4-bit code sent to the CPU by the maintenance interface. The code used for register selection is transmitted to the CPU via NOC-8, bits C through F. The maintenance interface must also set NOC-8, bit 8 (read) to enable the CPU to transfer data from its CIAR and data from the selected register to the maintenance interface display registers. Note that the maintenance interface must set up the register select code before issuing the read signal to the CPU. The CIAR is received at the maintenance interface via the bits of NIC-8, NIC-9, and NIC-A. Data from the code-selected register is received via the bits of NIC-B through NIC-E. Select codes and their corresponding registers are listed in Table 3-12.

TABLE 3-12. MAINTENANCE INTERFACE SELECT CODES

Hex Code	Register(s) Selected	Bits
0	Current instruction address register (CIAR)	0-63
1	Data flag register	0-63
2	Invisible package address (absolute bit address)	0-31
	Page zero address (absolute bit address)	32-63
3	External interrupt register	17-31
	Channel 1	17
	2	18
	3	19
	4	20
	5	21
	6	22
	7	23

TABLE 3-12. MAINTENANCE INTERFACE SELECT CODES (Cont'd)

Hex Code	Register(s) Selected	Bits
3	Channel 8	24
	9	25
	10	26
	11	27
	12	28
	Not used	29
	Direct access channel	30
	Monitor interval timer	31
	Channel read active-write active	32-55
	Channel 1	32-33
	2	34-35
	3	36-37
	4	38-39
	5	40-41
	6	42-43
	7	44-45
	8	46-47
	9	48-49
	10	50-51
	11	52-53
	12	54-55
4	Parity fault type	0-7
	Access instruction parity fault	0
	Stream instruction parity fault	1
	CPU parity fault	2
	Search parity fault	3
	Exchange parity fault	4
	I/O parity fault	5
	NIC memory 0 parity fault	6
	NIC memory 1 parity fault	7
	[These signals may be cleared by setting Clear Fault bit (NOC-8, bit 7) in maintenance interface]	
	Floating point counts	26-31

TABLE 3-12. MAINTENANCE INTERFACE SELECT CODES (Cont'd)

Hex Code	Register(s) Selected	Bits
	When counting floating point items with monitoring counters, lowest order bit is 2^2-bit. More accuracy can be obtained by reading bits 2^0 and 2^1.	
		2^1 2^0
	Number of alignments > K	26 27
	Number of normalizations > K	28 29
	Number of right shift normalizations	30 31
	If 2^0 and 2^1 are to be used, computer must be stopped before selecting the event and reading the counter. These bits are cleared by the not select code.	
	Parity fault address Address of first parity error retained in this register. This register cleared by clear fault (NOC-8, bit 7).	32-63
5	Bounds hit address (absolute bit address, right justified). Address of first bounds hit retained in this register. Bounds checking performed on sword boundaries only. This register cleared by clear fault (NOC-8, bit 7).	0-31
6	Counter A1	0-15
	Counter A2	16-31
	Counter B1	32-47
	Counter B2	48-63
	If NOC-D, bit 8 = 0, both counters clear after read bit (NOC-8, bit B) is sent and after both counters are transferred into display register. If NOC-D, bit 8 = 1, only upper bit of each 32- or 16-bit counter will be cleared.	

COUNTER MONITORING

A second method of evaluating computer system performance is that of monitoring the two 32-bit counters in the CPU. To be compatible with 16-bit word format, the two counters are functionally divided into two 16-bit counters each and carry the designations A1/B1 and A2/B2. On command from the BC, the maintenance interface may be connected by an event line to any of the four 16-bit counters. The counters can be read by BC selection as inputs to the display registers in the maintenance interface. They also can be combined in various ways to form one or two 32-bit counters, if desired. This reconfiguration is accomplished using the carry lines (bits B, C, D, and E of NOC-D). The counters are enabled by normal channel bits (NOC-D, bits 6 and 7) selected by a mask from the maintenance interface. The maintenance interface has the option of stopping the CPU on any count condition and accomplishes this function using the stop bits (NOC-D, bits 9 and A). Table 3-13 lists the events which can be counted and their corresponding select codes required to read the count. NOC-E, bits 0 through 9 are used to obtain the proper counter-event combination.

TABLE 3-13. MAINTENANCE INTERFACE EVENT SELECT CODES

Hex Codes		Event
Counter A1/B1	Counter A2/B2	
01_{16}		Number of branches out of instruction stack.
	01_{16}	Number of branches in instruction stack.
02		Number of instruction sword references.
	02	Total number of minor cycles execution was stopped waiting for an instruction from central memory. This does not include waits for jumps.
03		Number of times an issue of a register instruction has been blocked because of an operand-result conflict.
	03	Total number of minor cycles that register instructions have had to wait before issue because of an operand-result conflict.
	04	Number of shortstop path usages.
05		Number of space table searches.
	05	Number of quarter swords searched in space table searches.

TABLE 3-13. MAINTENANCE INTERFACE EVENT SELECT CODES (Cont'd)

Hex Codes		Event
Counter A1/B1	Counter A2/B2	
06		Number of exchanges caused by program force interrupts.
	06	Number of exchanges caused by external interrupts.
07		Number of exchanges caused by access interrupts.
	07	Total number of exchanges.
08		Number of direct access channel memory requests.
	08	Number of direct access channel requests accepted.
09		Number of normal channel memory requests.
	09	Number of normal channel memory requests accepted.
0A		Number of CPU memory requests.
	0A	Number of CPU memory requests accepted.
0B		Total number of memory requests.
	0B	Total number of memory requests accepted.
	0C	Number of vector instructions with result fields less than 64 words.
0C		Number of vector instructions with result fields between 65 and 8K words.
	0D	Number of vector instructions with result fields of more than 8K words.
	0E	Number of string instructions with result fields less than 8 bytes.
0E		Number of string instructions with result fields between 8 and 32 bytes.
	0F	Number of string instructions with result fields of more than 32 bytes.

TABLE 3-13. MAINTENANCE INTERFACE EVENT SELECT CODES (Cont'd)

Hex Codes		Event
Counter A1/B1	Counter A2/B2	
	10	†Number of alignments greater than a certain constant. The constant has not been selected. It will be between 0 and 128.
11		†Number of normalizations greater than a certain constant. The constant has not been selected. It will be between 0 and 128.
	11	†Number of right shift normalizes.
12		Number of times a particular function code or a particular category of function codes is executed. The count condition is determined by an 8-bit select code (NOC-F, bits 0 through 7) and an 8-bit mask sent to the CPU via NOC-F, bits 8 through F. If the select code bits and the corresponding instruction function code bits are equal wherever there is a 1 in the mask, the counter will be incremented. If the mask contains all zeros, all instructions will be counted.
	12	Time - 1 MHz.
13		Number of vector instructions.
	13	Number of cycles where data is not available at the output of a functional unit (string or floating point) once data has been requested for all input streams. This time does not include the time required for initial setup (preceding requests for memory) or shutdown (following the input of the last operands to a functional unit) of vector or string instructions. This count thus permits the programmer to analyze the amount of time required for start-up memory accesses, pipeline/functional unit length, space table searches, and memory conflicts for a specific instruction.

† Counted in groups of four; therefore, the lowest bit in the counter is the 2^2-bit.

TABLE 3-13. MAINTENANCE INTERFACE EVENT SELECT CODES (Cont'd)

Hex Code		Event
Counter A1/B1	Counter A2/B2	
14		Time enabled from microcode; number of minor cycles microcode MON = 1 is selected.

COUNT GATES AND MASKING

Counters are incremented when the selected event occurs, the proper count bit (NOC-D, bit 6 or 7) is up, and one or more of the following gate-mask conditions is satisfied:

- Data flag 56/57 is set in the CPU and data flag 56/57 mask (NOC-E, bits E and F) in the maintenance interface are set.
- The CPU is in monitor mode and the monitor mode mask bit (NOC-E, bit C) in the maintenance interface is set.
- The CPU is in job mode and the job mode mask (NOC-E, bit D) and the all jobs enable mask bits (NOC-E, bit B) are set.
- The CPU is in job mode, the job mode mask is set and the job enable of monitoring counters is set in the invisible package of the job being executed. This feature allows a count to be made during CPU selected jobs only.

CARRY OPERATIONS

There is one enable carry line associated with each 16-bit counter. Enable carry bit A1 enables the carry operation into counter A1 from counter A2. Enable carry bit A2 (NOC-D, bit C) enables the carry operation into counter A2 from counter A1. There are also equivalent lines for B counter carry operations. The four bits used in controlling the carry operations are NOC-D, bits B through E.

A zero on carry lines A1 and A2 allows the counters to operate as two 16-bit counters. Only half of the total number of events are available at the selection network for counter A1 or A2; therefore, if a 32-bit count is desired, the logical state of the carry lines must be manipulated accordingly. For example, if an event is enabled to use counter A1 and 32-bit count is desired, enable carry line A1 (NOC-D, bit B) must be 0 or clear and carry line A2 must be a 1 or set. In this example counter A1 will have the least significant bits.

If both carry lines for a counter pair are set, and it NOC-D, bit 8 (clear counter overflow bits) is set, undefined results will occur when the counters are cleared.

STOP BITS

One stop bit is available for each counter pair. When a stop bit is set, the event incrementing either 16-bit counter in the pair will stop the computer. The two bits used for controlling the stop lines are bits 9 and A of NOC-D. In response to setting a stop bit, the computer responds to the maintenance interface by setting the event stop bit (NIC-F, bit 4). The maintenance interface may restart the computer after setting the clear fault bit (NOC-8, bit 7).

COUNTER PROGRAMMING SEQUENCES

The following procedures are typical steps that could be used to set up the CPU counters from the maintenance interface.

NOTE

To insure proper initialization of the counters, the count bits (NOC-D, bits 6 and 7) must be cleared prior to making each new count selection.

1. Set the following bits, as required:
   a. Stop; counter A (NOC-D, bit 6)
   b. Stop; counter B (NOC-D, bit 7)
   c. Enable carry (into) A1 (NOC-D, bit B)
   d. Enable carry (into) A2 (NOC-D, bit C)
   e. Enable carry (into) B1 (NOC-D, bit D)
   f. Enable carry (into) B2 (NOC-D, bit E)

2. Clear NOC-D, bit F (0 Counter A specifications) and set desired event select code (NOC-E, bits 0 through 4) and mask selection bits (NOC-E, bits B through F) for counter A into NOC-E.
3. Set NOC-D, bit F (1 counter A specifications) and set desired event select code (NOC-E, bits 5 through 9) and mask selection bits (NOC-E, bits B through F) for counter B into NOC-E.
4. If A1/B1 event hex code 12 for function counting was selected, set the appropriate counter/event select and mask bits in NOC-E and NOC-F, respectively.
5. Set count bit A or B (NOC-D, bit 6 or 7) as desired to activate the counters.

The counters will then count the selected events and will continue until their respective count lines are dropped.

LOGIC FAULT MONITORING

There are three types of logic faults detected in the computer. They are:

- Memory parity (CPU)
- Microcode memory parity
- Multiple match

Refer to STAR-100 Computer System Hardware Reference Manual listed in the preface.

The maintenance interface may detect any of these types of faults on NIC-F, bits 0, 1, and 2. After sensing the fault, the maintenance interface must clear the fault by setting NOC-8, bit 7 and must determine the appropriate response to the fault. It may restart the computer by setting NOC-8, bit 3 (run).

STAR-100 CONTROL INTERFACE

The MCU control/microcode memory interface can be used to perform the following operations:

- Load computer microcode memory from MCU memory
- Read from computer microcode memory into MCU memory
- Read status information from computer microcode memory
- Issue control bits to govern operation of computer microcode memory

The control interface in the MCU is connected to the microcode memory module of an HLP such as the STAR-100 or STAR-65 via an I/O channel similar to a STAR data channel. The control interface connects to the block transfer channel of the BC and is programmed as the SAC coupler except that different normal channels are used for control. The microcode memory (or HLP) end of the channel is like a standard A coupler with channel control and fanin/fanout logic between the channel and the 224-bit microcode words. The A coupler in the HLP does not use the following lines normally used in a STAR data channel:

1. Parity error from A (PEFA)

   The A coupler does not check parity on any function sent by the control interface or does not send a parity error to the control interface on a microcode parity error. The MCU checks for microcode parity errors via the normal channel lines between the MCU and the CPU just as it would for any other CPU parity error.

2. Illegal from A (IFA)
3. Interrupt from B (IFB)
4. System control [such as control strobe from A (CSFA) or control from A (CFA)]
5. No parity is sent with data on a read operation

Refer to appendix C in the back of this manual for a complete description of a STAR data channel.

The BC in the MCU uses the block transfer channel for data transfers through the control interface and normal channel 6C for function codes and control of the control interface.

CONTROL INTERFACE TO A COUPLER FUNCTION CODES

The control interface in the MCU sends function codes via the I/O channel as directed by the BC. These codes are sent to the A coupler (or microcode memory module) via NOC-6C, bits 1, 2, and 3 and are defined in Table 3-14.

TABLE 3-14. FUNCTION CODES FOR CONTROL INTERFACE/MICROCODE MEMORY INTERFACE

Bit $3(2^0)$	Bit $2(2^1)$	Bit $1(2^2)$	Name	Function
0	0	0	Null	Automatically sent by control interface as the second half of any function.
0	0	1	Read memory	Read a block of microcode memory from the current microcode address.
0	1	0	Write memory	Write a block of microcode memory from the current microcode address.
0	1	1		Not normally used, but performs the same as an EOP in computer.
1	0	0	Data	Automatically sent with data during a write microcode memory operation.
1	0	1	Read status	Read the current microcode status (explained in detail later in this section).
1	1	0	Write switch	Switches that provide control of microcode execution (explained in detail later in this section).
1	1	1	EOP	End of operation clears microcode memory interface of all previous functions and also clears counter that controls data fanin/fanout to and from the I/O channel.

MICROCODE WRITE SWITCHES

Microcode switches are 1-bit control terms used to control the microcode memory. Each switch is one bit of the write switch control word (Figure 3-16). When the BC in the MCU issues the write switch (110) function code, the control interface receives the code from the block transfer channel and sends it on to the microcode memory interface (A interface). The write switch function code causes the microcode memory to store the write switch control word in a register.

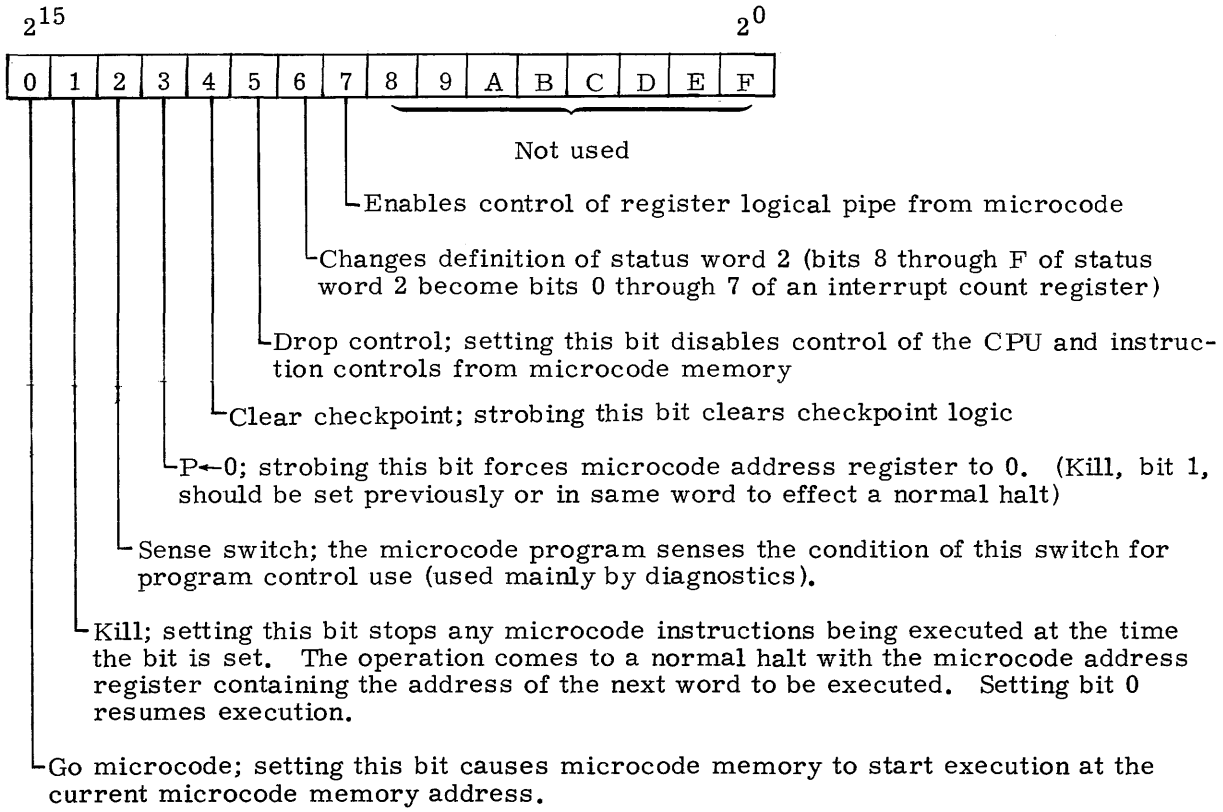


Figure 3-16. Microcode Write Switch Control Word

Write switch functions represented by bits 0, 3, and 4 are one-shot functions. This is accomplished by setting the required bit in the even 16-bit half word of a transfer and clearing the bit in the odd 16-bit half word of the transfer. If the bit is set in both halves of the 32-bit transfer, the function would be performed during the first transfer but ignored in the second transfer and on succeeding 32-bit transfers if the switch remains set.

Switch functions represented by bits 0 and 3 are delayed by one memory cycle so that other functions in the same data word have time to propagate. For example, bit 1 (kill) and bit 3 (P 0, forcing address register to 0) are a legal combination as are bits 0 and 2 (go microcode and sense switch). Note that legal combinations are not limited to the two examples just given.

Switch functions represented by bits 1, 2, 5, 6, and 7 are latching in that they do not change from their present state until another function is sent. Since a single function consists of two 16-bit data transfers, each switch function must be repeated in both halves of a 32-bit data transfer.

MICROCODE STATUS

When the BC in the MCU requests a status report (using read status, 101_2) of microcode memory operations via the control interface, the A interface in the HLP puts two 16-bit status words on the I/O channel. They are shown in Figure 3-17. The MCU may receive any number of input status words from the HLP. However, if the MCU requests more than two status words the microcode memory continues to repeat status word 2. The requesting and receiving of status does not have any effect on microcode memory or controls.

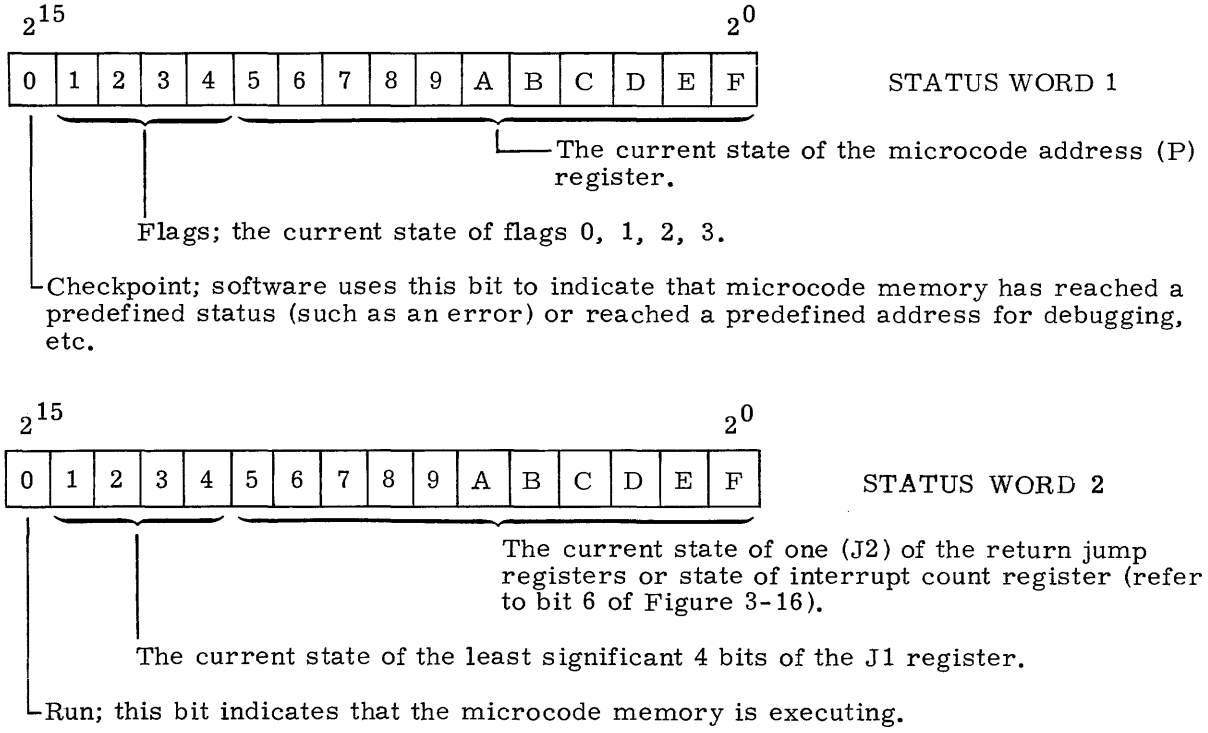


Figure 3-17. Microcode Memory Status Words

PROGRAMMING SEQUENCES

The following paragraphs outline basic programming sequences that can be used to exercise the microcode memory through the control interface in the STAR-100 MCU.

CONTROL INTERFACE SELECTION

Prior to attempting any of the following procedures such as reading from or writing into microcode memory, select the control interface as follows:

1. Select the control interface by clearing bit 9 and setting NOC-2, bit A.
2. Connect NOC-6C to the control interface by setting NOC-5, bit 8 (channel switch set 2^1) and clearing NOC-5, bit 9.

WRITING INTO MICROCODE MEMORY

1. Issue EOP function code to clear the microcode memory interface.
   a. Clear initiate functions bit (NOC-6C, bit 0).
   b. Place EOP function code (111_2) in NOC-6C, bits 1 through 3.
   c. Set, then clear the initiate functions bit (NOC-6C, bit 0).

   NOTE

   The EOP code does not clear the microcode memory address counter.

2. Issue Write Memory function code.
   a. Place write memory function code (010_2) in NOC-6C, bits 1 through 3.
   b. Set initiate functions bit (NOC-6C, bit 0).
   c. Execute output-block transfer instruction to issue a 32-bit (2-word) dummy address.

   NOTE

   The microcode memory ignores this address; data transfer will proceed from the current microcode memory address. However, a 2-word dummy address must be issued to satisfy STAR data channel procedures. Content of the dummy address does not matter.

   d. Clear initiate functions bit (NOC-6C, bit 0).
3. Start output data transfer.
   a. Place data function code (100_2) in NOC-6C, bits 1 through 3.
   b. Execute output-block transfer instruction to initiate data transfer.

   NOTE

   The block transfer instruction word count determines the number of 16-bit words transmitted from the MCU. The block transfer instruction word count must have a remainder of 0, modulus 14.

   c. Clear data function code in NOC-6C, bits 1 through 3.
4. Issue EOP function code (refer to step 1).
5. If more data must be transferred, repeat the sequence in step 2 after reloading BC memory.

NOTE

Because BC memory is smaller than the microcode memory, several repetitions of the write sequence may be required to write a long block in microcode memory.

READING FROM MICROCODE MEMORY

1. Issue EOP function code to clear the microcode memory interface.
   a. Clear initiate functions bit (NOC-6C, bit 0).
   b. Place EOP function code (111_2) in NOC-6C, bits 1 through 3.
   c. Set, then clear the initiate functions bit (NOC-6C, bit 0).
2. Issue Read Memory function code.
   a. Place read memory function code (001_2) in NOC-6C, bits 1 through 3.
   b. Set initiate functions bit (NOC-6C, bit 0)
   c. Execute output-block transfer instruction to issue a 32-bit dummy address.

NOTE

Data transfer proceeds from the current microcode memory address. The dummy address is ignored.

   d. Clear initiate functions bit (NOC-6C, bit 0).
3. Execute input-block transfer instruction to start data transfer.

NOTE

The block transfer word count, placed in the A register, determines the number of 16-bit words read in.

4. Issue EOP function code (refer to step 1)
5. If more data is to be transferred, repeat sequence from step 2.

NOTE

Before reading in another data block, the data block, the data just read in must be moved from BC memory to some other storage medium (a disk, for example).

ISSUING WRITE SWITCH BITS TO MICROCODE MEMORY

1. Issue EOP function code.
   a. Clear initiate functions bit (NOC-6C, bit 0).
   b. Place EOP function code (111_2) in NOC-C, bits 1 through 3.
   c. Set, the clear the initiate functions bit (NOC-6C, bit 0).
2. Issue write switch function code.
   a. Place write switch function code (110_2) in NOC-6C, bits 1 through 3.
   b. Set initiate functions bit (NOC-6C, bit 0).
   c. Execute output-block transfer instruction to issue 32-bit dummy address.

   NOTE

   The microcode memory ignores this address.

   d. Clear initiate functions bit (NOC-6C, bit 0).
3. Output write switch bits.
   a. Place data function code (100_2) in NOC-6C, bits 1 through 3.
   b. Execute output-block transfer instruction to issue two 16-bit write switch words.

   NOTE

   Normally this operation is limited to two 16-bit output words, however, there is no restriction on data length. In some cases, it may be useful to issue a series of write switch words, to make repeated starts of a recurring operation for example.

4. Issue EOP function code (refer to step 1).

READING STATUS FROM MICROCODE MEMORY

1. Issue EOP function code.
   a. Clear initiate functions bit (NOC-6C, bit 0).
   b. Place EOP function code (111_2) in NOC-6C, bits 1 through 3.
   c. Set, then clear the initiate functions bit (NOC-6C, bit 0).

2. Issue read status function code.

   a. Place read status function code (101_2) in NOC-6C, bits 1 through 3.

   b. Set initiate functions bit (NOC-6C, bit 0).

   c. Execute output-block transfer instruction to issue a 32-bit dummy address.

   NOTE

   The microcode memory ignores the dummy address.

   d. Clear initiate functions bit (NOC-6C, bit 0).

3. Execute input-block transfer instruction to read in 16-bit status words.

4. Issue EOP function code (refer to step 1).

SDU COUPLER INTERFACE

This interface is present in the SCU and is used in the display/edit station. Its function is to drive the Station Display Unit (SDU). Because the SDU coupler and SDU are so closely related, both items are treated together in section 4 of this manual.

HR600 PRINTER INTERFACE

GENERAL

The printer interface in the maintenance control unit acts as a data handling channel between the HR600 Line Printer and the SCU normal channels. Function commands, status words, and data words are transferred to and from the printer via this interface. Bit descriptions of the various normal channel bits used in this interface are given in Table 3-15.

The printer interface is multiplexed with the 854 Disk Storage Drive and SAC coupler (coupler B) interfaces on channel 6 by the normal channel switch logic. Although the printer interface uses normal channel 6B on both input and output operations, the SCU may sense status bits on normal input channel 3 (refer to Table 3-15) as well.

NORMAL CHANNEL SWITCH

Although the normal channel switch is not an interface, it is connected to specified bits of normal channels 5 and 6. Since three interfaces must share normal channel 6 between the peripheral devices and the BC's normal channel interface, the normal channel switch must be programmed to make the proper selection. Normal input and output channels 6A service the 854 Disk Storage Drive interface, channel 6B services the HR600 printer interface, and channel 6C services the second SAC coupler (or coupler B).† Figure 3-18 illustrates the relationship of the normal channel switch to normal channels 5 and 6 and to interfaces connected to channel 6. To select the desired channel (6A, 6B, or 6C), the BC must set or clear bits 8 and 9 (channel switch set) of NOC-5 as follows:

	Bit 8	Bit 9
Channel 6A (disk)	0	0
Channel 6B (printer)	0	1
Channel 6C (coupler B)	1	0
Channel 6D (not used)	1	1

Selecting the desired channel with the appropriate code of NOC-5 selects both the input and output channels for the device interface. For a more detailed explanation of the normal channel bits used in channels 6A, 6B, and 6C, refer to the appropriate interface description in this manual.

†A fourth selectable channel (6D) is not used.

TABLE 3-15. BIT DESCRIPTIONS FOR PRINTER/NORMAL CHANNEL INTERFACE

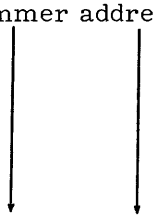
Channel	Bit	Name	Function
NOC-6B	0	Start indicator	Setting this bit lights the START indicator on the printer in response to a start signal, provided all status conditions are correct.
NOC-6B	1	Select 8 lines/inch	Setting this bit causes the printer controller to select a printing density of eight lines per inch printing format. Clearing this bit selects a printing density of six lines per inch.
NOC-6B	2	Adv ribbon	Setting this bit causes the printer to advance the print ribbon during printing operations.
NOC-6B	3	Adv paper	Setting this bit causes the printer to advance the paper forms in its carriage. Paper motion continues only as long as this bit is set.
NOC-6B	4	Hammer enable	Setting this bit causes all hammers addressed for a specific drum row pulse count to fire.
NOC-6B	5	Compare	Setting this bit strobes the address on bits 8 through F to be sent to the printer.
NOC-6B	6 7		(Not used) (Not used)
NOC-6B	8 9 A B C D E F	Hammer address 	These eight bits constitute the address sent to the printer to select the desired print hammer. 7 6 5 4 3 2 1 0
NIC-6B	D	Out of paper	This status bit indicates that the printer has run out of paper.
NIC-6B	E	Drum latch switch	This status bit indicates that the printer gate is not closed and that the printer is therefore not ready.
NIC-6B	F	29V alarm	This status bit indicates that a malfunction occurred in the 29-volt power source.

TABLE 3-15. BIT DESCRIPTIONS FOR PRINTER/NORMAL CHANNEL INTERFACE (Cont'd)

Channel	Bit	Name	Function
NIC-3	4	Index pulse	This status bit indicates that the print drum is at its starting point. It is also used as a resync signal for counting drum row pulses.
NIC-3	5	Row pulse	This status bit indicates that a new drum row is coming under the printer hammers.
NIC-3	6	Paper strobe	This status bit from the printer indicates that paper motion has stopped.
NIC-3	7	Start	This status bit indicates that the printer was given the command to begin operation.
NIC-3	8	Page eject	This status bit indicates that the paper in the printer was advanced until a punched hole was detected in the format tape (top of form).

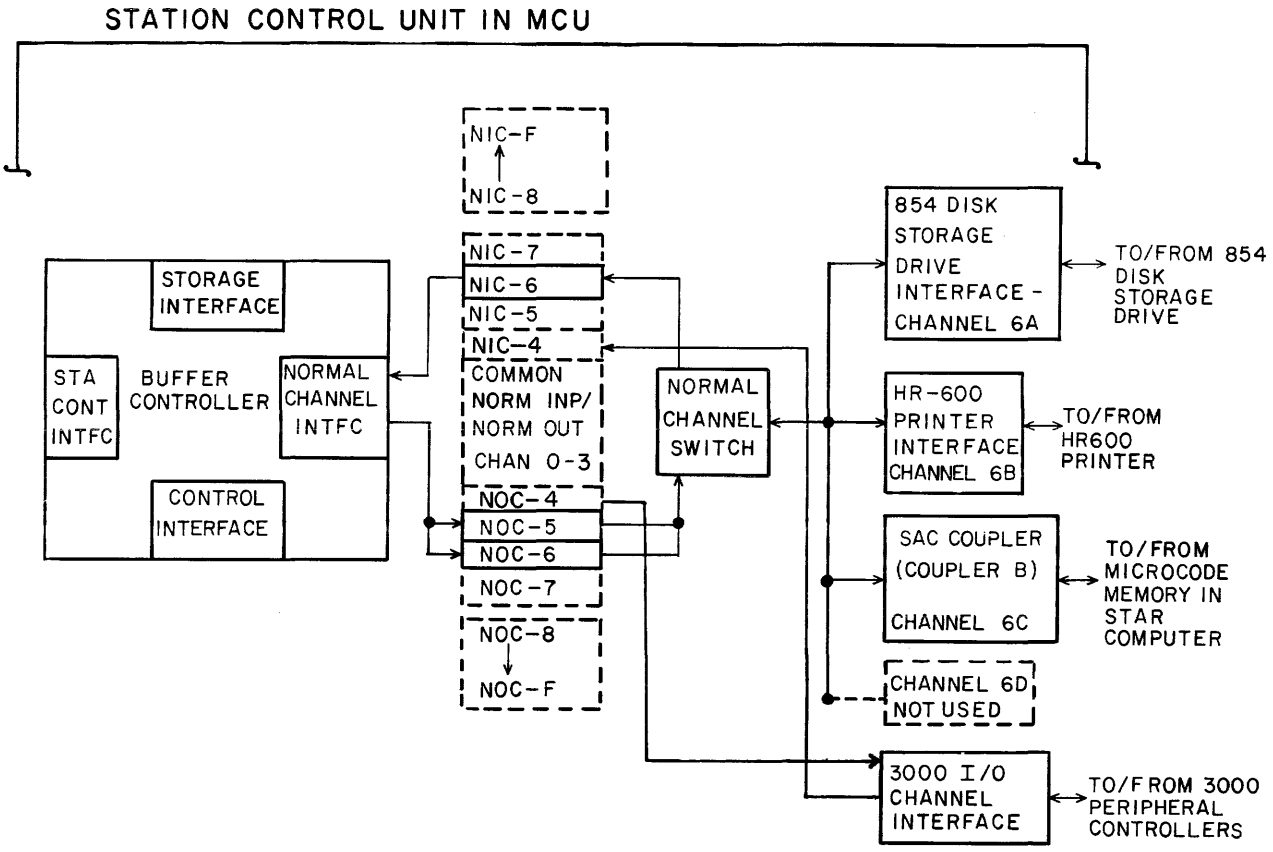


Figure 3-18. Normal Channel Switch/Normal Channel 6 Relationship

PROGRAMMING SEQUENCES

Since the HR600 printer is completely controlled by software, only a general sequence of events is included in the following paragraphs.

1. Initially, the software initiates the following bits.

   Sets advance ribbon bit (NOC-6B, bit 2)
   Selects 6/8 lines per inch printing density (NOC-6B, bit 1)
   Tests status for the start bit (NIC-3, bit 7)
   Tests the start indicator (NOC-6B, bit 0)
   Sets status for index pulse (NIC-3, bit 4) to establish drum synchronism at a drum speed of 800 rpm. At this speed an index pulse occurs every 75 milliseconds.

2. Following the index pulse, software checks for a drum row pulse (NIC-3, bit 5). There are 63 drum row pulses between each index pulse. Since a drum row pulse corresponds to a character row, the software must match the character to be printed with the correct drum row pulse.

3. When the correct drum row pulse is determined, the hammer address(es) (NOC-6B, bits 8-F) is sent to the printer to specify which of the 136 hammers are to be fired.

4. The compare bit (NOC-6B, bit 5) strobes the hammer address to the printer.

5. The hammer enable bit (NOC-6B, bit 4) causes all hammers previously addressed for the specified drum row pulse to fire. From the initial sensing of the drum row pulse, the software has 1.17 milliseconds to print the character.

6. Steps 2 through 5 are repeated until all characters on that line are printed.

7. After all characters for one line are printed, the software advances the paper (NOC-6B, bit 3) to print the next line and continues printing subsequent lines.

STAR-65 MAINTENANCE INTERFACE

To be supplied.

STATION DISPLAY UNIT AND SDU COUPLER 4

GENERAL DESCRIPTION

The SDU coupler, the SDU, and keyboard/display units form a display/edit station subsystem (Figure 4-1) which enables the SCU to transmit and receive data to or from the keyboard/display units. The SDU coupler serves as an interface that allows blocks of data to be transferred in either direction between SCU memory and SDU memory. Data transfers are initiated by the BC in the SCU.

The SDU is a memory that is used to hold messages and data displayed on the display units. SDU memory is partitioned into a number of buffer areas that each serve one CRT display unit. A message written into one of these buffers is automatically displayed continuously on the corresponding display as long as the message remains in the buffer. Messages to be displayed come from the SCU, which can read or write any area in SDU memory, through the SCU coupler. All display data must be in the form of ASCII characters.

The basic SDU contains 4096 16-bit words, enough to support a group of seven displays. Up to three additional 4K memory modules can be added to serve a maximum of 28 displays. For addressing purposes, the four 4K-memory modules and associated groups of seven keyboard/display units are labeled groups 0, 1, 2, and 3.

A second purpose of the SDU is to provide a path from the keyboards to the SCU normal input channels. The SDU simply provides an electrical path from the keyboards to the SCU. Keyboard data does not flow directly into SDU memory for display. To display typed-in data, the SCU must first read from a keyboard and then transmit the keyboard data to SDU memory.

Figure 4-2 shows the relationship between SDU memory and the displays. Note that there are actually two buffer areas for each display, a 512-word buffer for the 16-line A field and a 64-word buffer for the two-line B field. Each 16-bit word holds two 7-bit, ASCII-coded display characters. Aside from the difference in buffer area locations, there is no difference between the two fields. A memory map (Figure 4-11), located near the end of this section, shows the buffer areas assigned to each display.

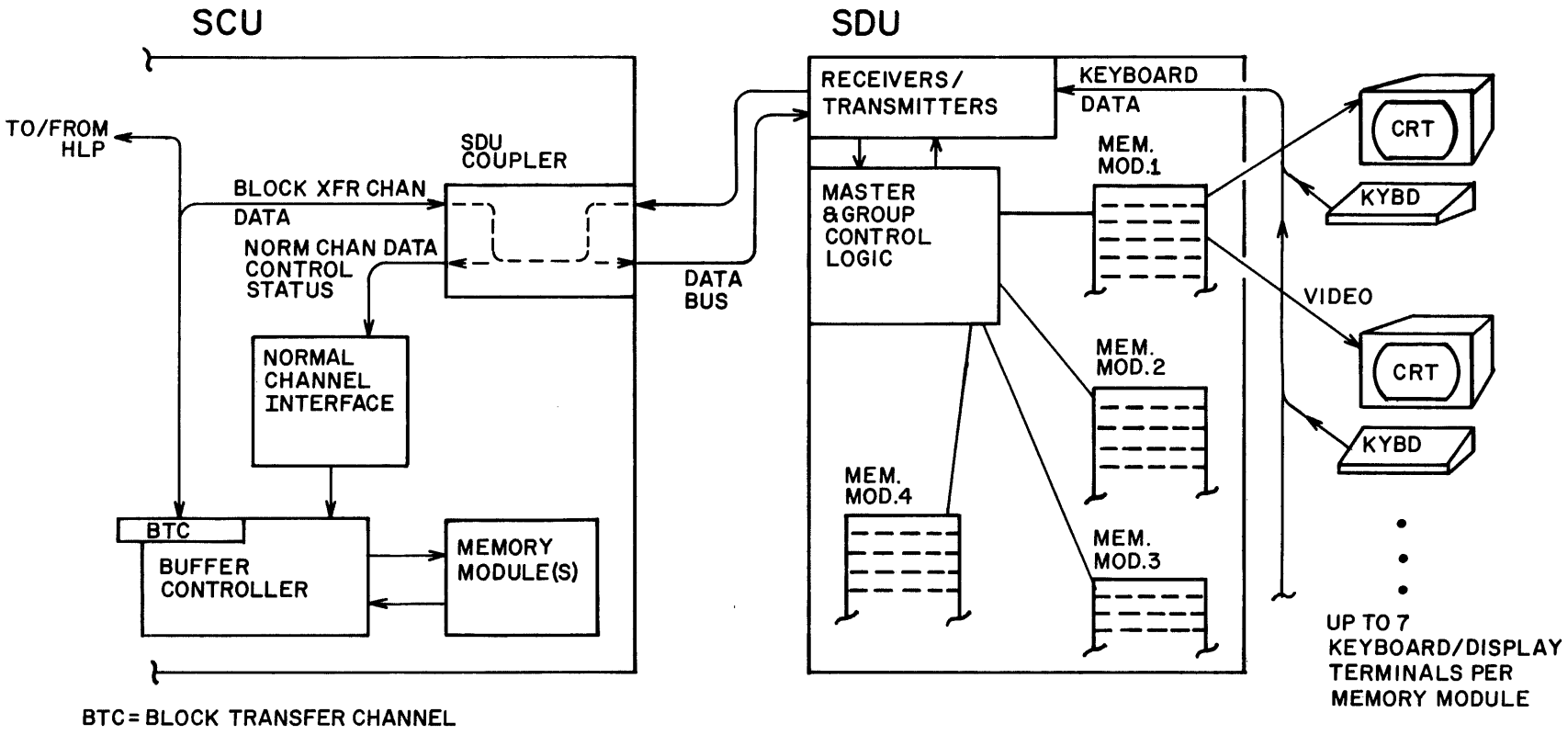


Figure 4-1. Typical SCU/SDU System Communication Paths

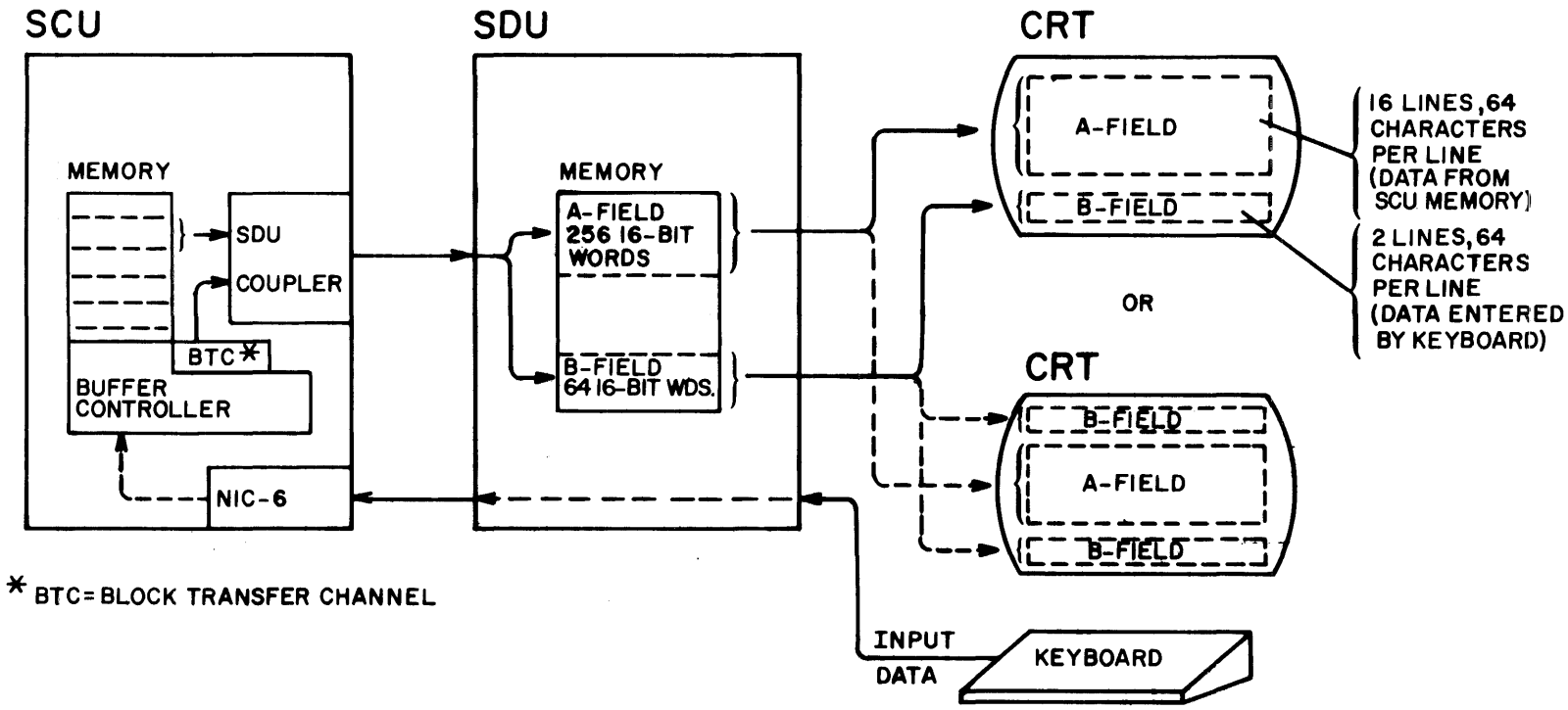


Figure 4-2. CRT Display Formats

As Figure 4-2 indicates, there are two possible B-field formats. The desired format must be selected at installation time by setting switches located on the SDU.

SDU COUPLER

The SDU coupler is an SCU interface that permits block transfers of data between the memories in the SCU and the SDU. Its functions are summarized as follows:

- Block transfers of data to and from the SDU.
- Transfers control, address, and data bits to SDU via normal output channels 5 and 6.
- Receives status and data bits from SDU via normal input channels 5, 6, and 7.

Figure 4-3 illustrates the relationship between the SDU coupler and the system with which it communicates.

Note that the SDU coupler shares the buffer controller block transfer channel with a STAR data channel coupler. Before executing a block transfer operation, the SCU program must issue a select code on one of the normal channels to select either the SDU coupler or STAR data channel coupler.

OUTPUT-BLOCK TRANSFER OPERATIONS

The BC can write into SDU memory using an output-block transfer operation through the SDU coupler.

An output-block transfer instruction initiates data transfer to the SDU. Before executing this instruction, the BC must inform the SDU of the starting address where the block of data will be stored. This information is transferred to the SDU via a normal output channel. Also, the twos complement of the number of words in the block to be transferred must be placed in the A register in the BC. To initiate an output, the BC executes an output-block transfer instruction. Normally, the block transfer stops when the number of words specified in the A register have been transferred. However, the SDU coupler can be programmed to terminate the block transfer operation earlier on one or all of the three following conditions.

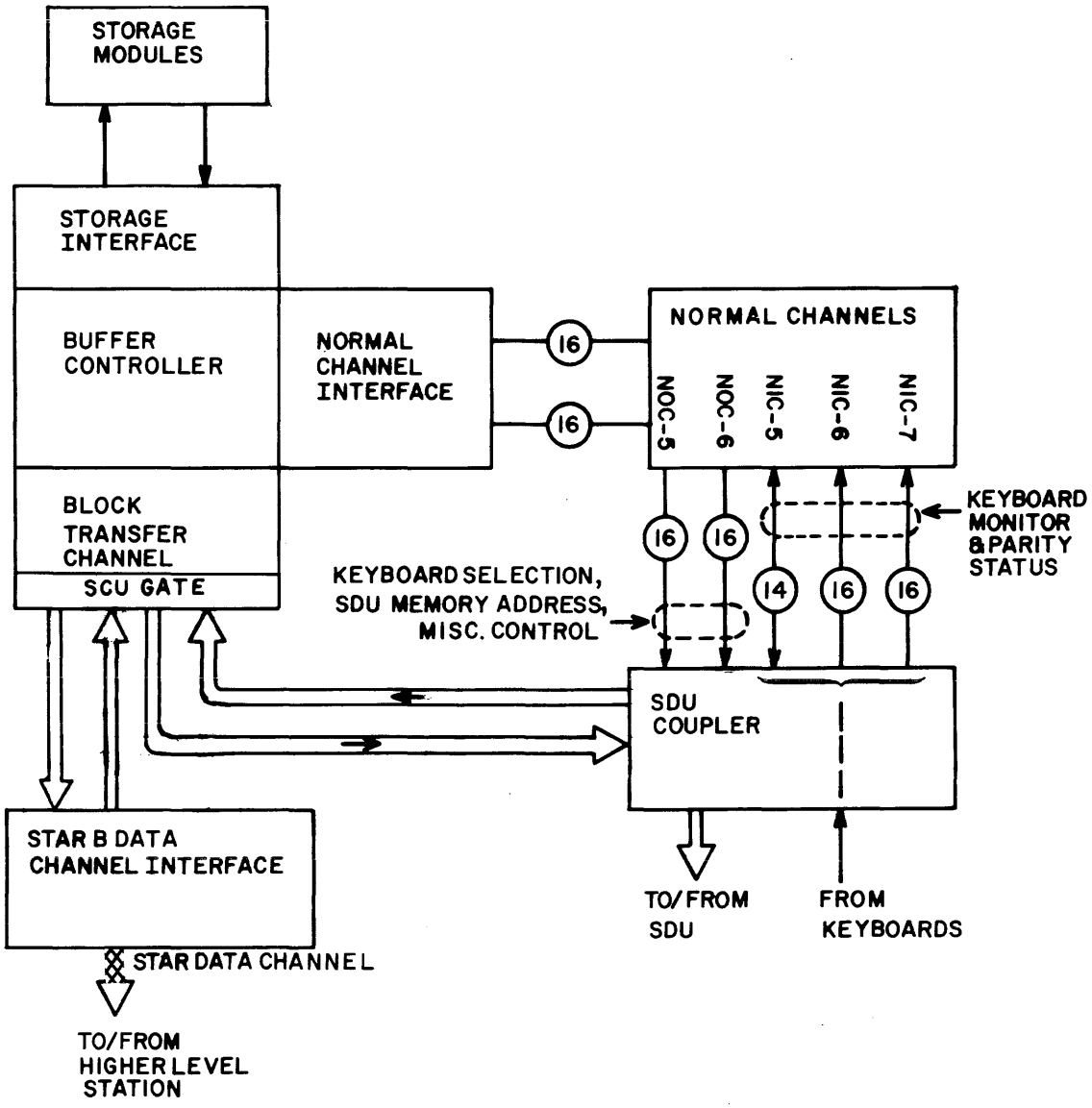


Figure 4-3. SDU Coupler System Relationship

Bit compare termination	The block transfer operation terminates when a character in which the terminating bit (upper bit) is 1 appears in the data stream.
Character compare termination	The block transfer operation terminates when a character in the data stream matches a compare character. The compare character must be placed in NOC-5.
Double compare	The block transfer operation terminates when either an end-of-record character (001F) or end-of-file character (001C) appears in the data stream. If this terminating condition is selected, either the end-of-record character or end-of-file character† must be held in NOC-5 throughout the data transfer operation.

These terminating conditions must be selected by normal channel bits before an output-block transfer is initiated.

Two types of information can be written into SDU memory, ASCII-coded display characters and cursor addresses. Cursor addresses are explained later in this section. Figure 4-4 shows the ASCII data format.

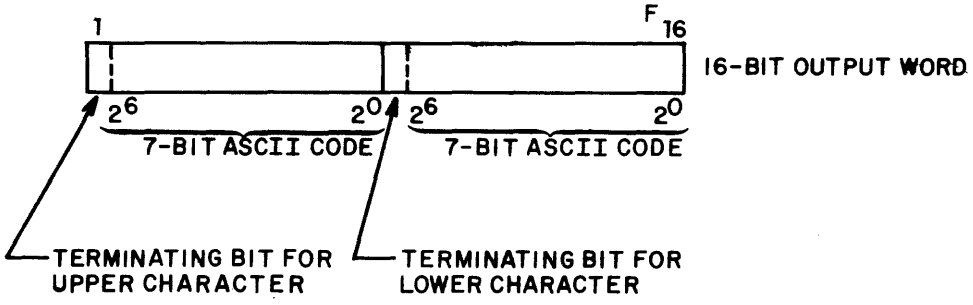


Figure 4-4. ASCII Data Format

† It makes no difference which of these two codes is placed on NOC-5. If one of them is present in NOC-5, the output operation terminates if either code appears in the data stream.

The upper character is displayed to the left of the lower character on the CRT screen.

PATTERN-WRITE OPERATION

A pattern-write operation is a special form of output-block transfer that writes a single 16-bit word repetitively throughout a block of SDU memory. It is useful for clearing SDU memory and for maintenance purposes.

Like a normal output-block transfer operation, a pattern-write operation is initiated by an output-block transfer instruction. Several preparatory steps must precede the block transfer instruction.

1. Issue starting address to SDU.
2. Set a normal channel bit that forces the SDU coupler into pattern-write mode.
3. Place the 16-bit pattern word in NOC-5.
4. Place two odd parity bits for the pattern word NOC-6, bits 4 and 5. One parity bit is required for the upper character and another for the lower character. The SCU program must generate the correct odd parity bits based on the contents of the pattern word.
5. Place a twos complement word count in the buffer controller A register.

When the output operation is initiated, the SDU coupler repetitively transfers the pattern word to the SDU until the number of words specified in the A register has been transferred.

The pattern word and associated parity bits must be held in the normal channels throughout the pattern-write operation.

Since no data is transferred from SCU memory during a pattern-write operation, the SCU memory starting address included in the output-block transfer instruction is of no significance.

CURSOR CONTROL

A cursor mark can be displayed on each of the 18 lines on a CRT screen. The cursor is a short, dash-like character that can be placed under any one of the 64-character positions on a line. Usually the cursor is used when keyboard data is being displayed

to indicate the position where the next character typed appears. Cursor position is controlled by a 7-bit cursor address sent from the SCU and held in SDU memory. A group of nine words for each display unit is reserved in SDU memory to hold cursor addresses. Each 16-bit memory word holds two 7-bit cursor addresses; thus, a group of nine words holds addresses for all 18 lines on a CRT display. Figure 4-5 shows the format of the cursor address.

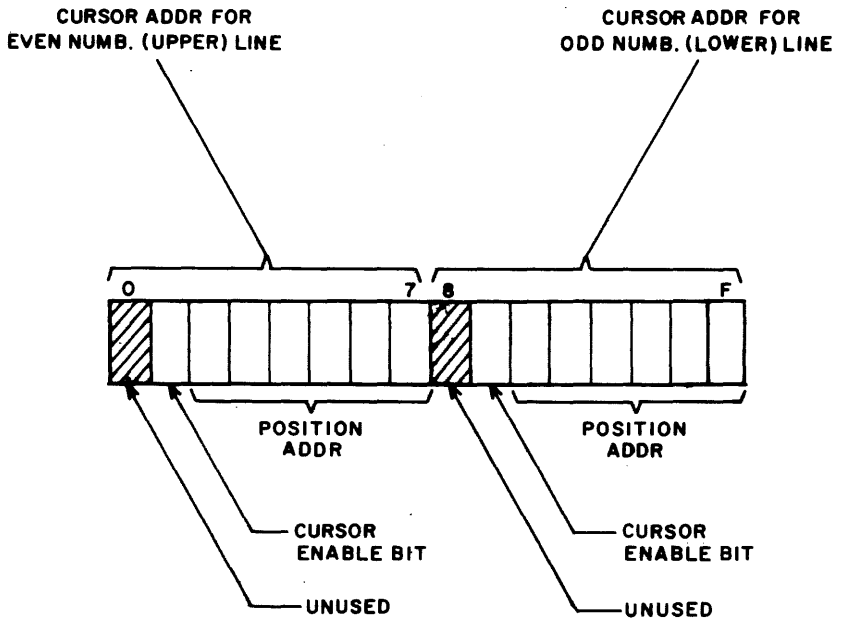


Figure 4-5. Cursor Address Format

If the cursor enable bit is 0, no cursor appears. If the enable bit is 1, the lower six bits of the address determine the position of the cursor mark. For example:

Cursor address = 40_{16}	Cursor mark under first (leftmost) character position of line.
Cursor address = 41_{16}	Cursor mark under second character position of line.
Cursor address = $7F_{16}$	Cursor mark under 64th (rightmost) character position of line.

Once positioned, a cursor mark does not move until the SCU loads a new cursor address. The SCU writes cursor addresses into SDU memory via the SDU coupler by an output-block transfer operation.

The memory areas reserved for cursor addresses are shown on the SDU memory map (Figure 4-11) near the end of this section.

INPUT-BLOCK TRANSFER

The SDU coupler allows the SCU to read a block of data words from any area in SDU memory. To prepare for a block-input operation, the SCU must:

1. Issue a starting address to the SDU on one of the normal output channels.
2. Place the complement word count (the number of words to be read in) in the buffer controller A register.

An input-block transfer instruction initiates data transfer.

Data transfer stops when the number of words specified by the word count has been transferred. A parity error in a data word read from SDU memory terminates the input operation early. Three optional terminating conditions (bit compare, character compare, and double compare) may be used for input-block transfer operations as well as for output operations. Refer to the previous description of the output-block transfer operation for an explanation of these terminating conditions.

OFFSET MODE

Offset mode modifies both input- and output-block transfer operations. It causes all characters in a data block to shift left or right one character position to allow for insertion or deletion of characters in a line of display data. A bit, appended to the SDU starting address sent from the SCU at the beginning of a block transfer operation, selects offset mode.

Figure 4-6 illustrates a write-offset operation.

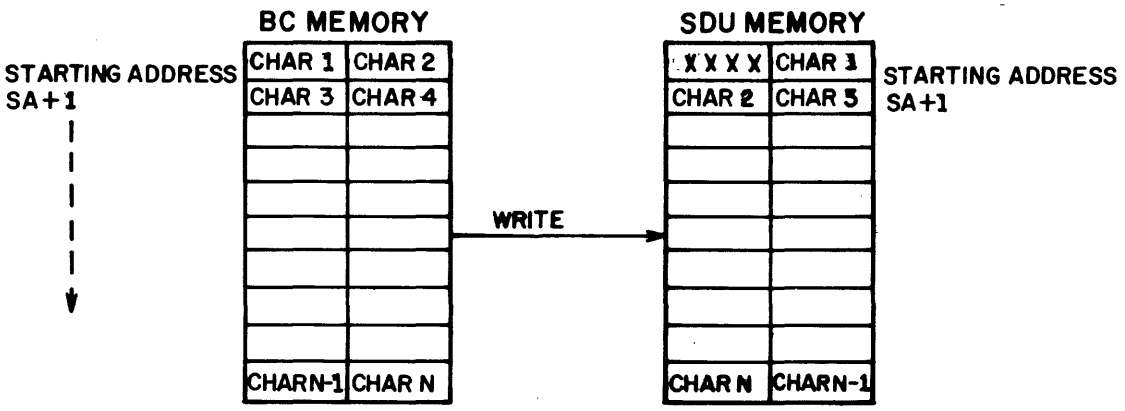


Figure 4-6. Write-Offset

Data is shifted one position left during a write-offset operation; the first character in the output block is lost. The first character position in the SDU memory block is equal to the contents of the offset register from the previous operation.

A read-offset operation (Figure 4-7) shifts data one character position to the right. The first character in the data block (character 1) is lost.

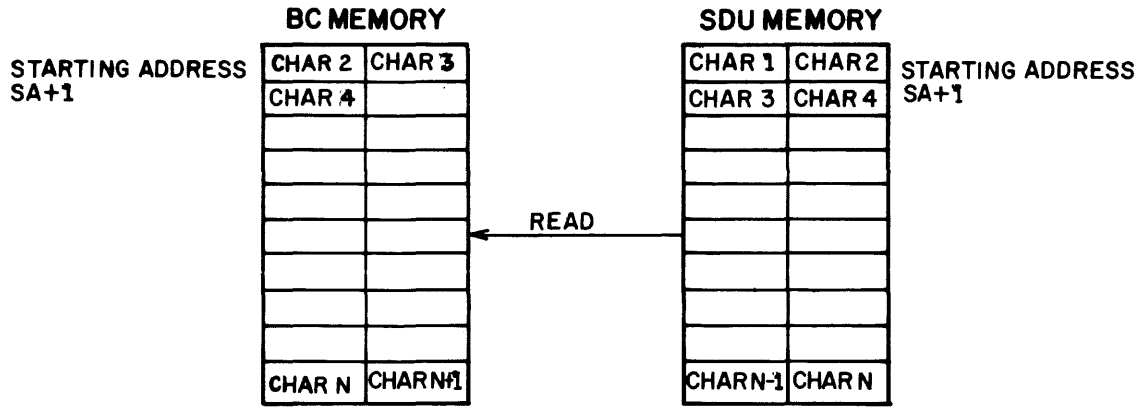


Figure 4-7. Read-Offset

KEYBOARD

The keyboards (Figure 4-8) allow operators to enter ASCII-coded data into SCU memory. A normal input channel acts as the input path to the BC in the SCU. In addition to 96 ASCII characters, the keyboards provide several function and mode keys that are used for control purposes.

As shown in Figure 4-2, input from the keyboards is routed through the SDU to SCU normal input channel 5. Keyboard data does not flow through SDU memory nor does the SDU provide any hardware link between the keyboards and display units. Software, resident in the SCU, must act as the link between the keyboards and displays.

To display keyboard data, the SCU program must individually read in each character on normal channel 5 as an operator types. After a character is read in, the program must initiate an output operation to write the character in the appropriate SDU memory buffer area for display.

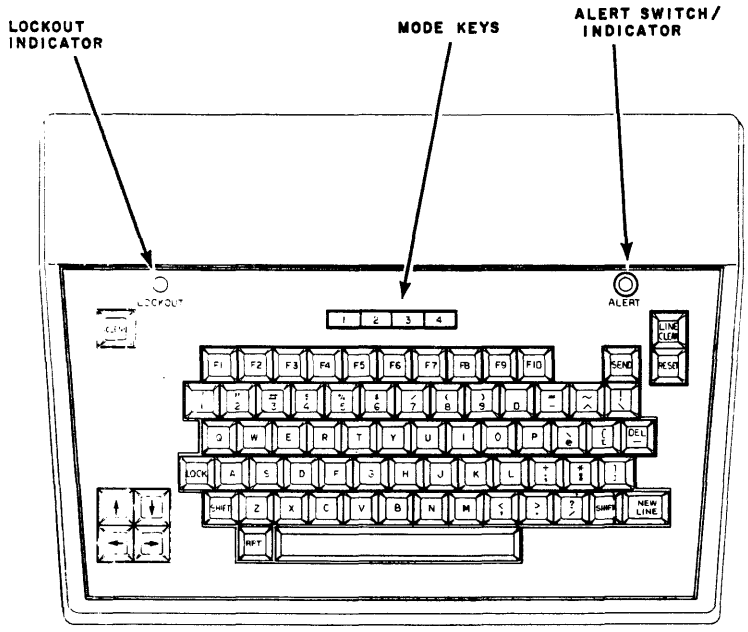
The SCU can read in data from only one keyboard at a time. Before reading from a keyboard, the SCU must issue a select code on normal channel 6 to designate the keyboard to be read.

NORMAL CHANNEL CONTROL OF KEYBOARD INPUT

Figure 4-9 shows the keyboard signals that are monitored or controlled by normal channel bits.

KEYBOARD ACTIVE SIGNALS

A keyboard active signal is available from each keyboard. This signal goes to 1 when the operator presses one of the function or ASCII data keys. The keyboard active signals from all keyboards remain connected to the normal input channels at all times regardless of which keyboard is selected for input. The SCU program must continuously scan the keyboard active bits to determine which keyboards have a data or function code available for input. When a keyboard goes active, the program must select the keyboard and read in the character before he releases the key he has pressed.



FUNCTION KEYS:

F1 THRU F10	NEW LINE
SEND	CLEAR
LINE CLEAR	ENTRY MARK KEYS (↑, ↓, ←, →)
RESET	

Figure 4-8. Keyboard Layout

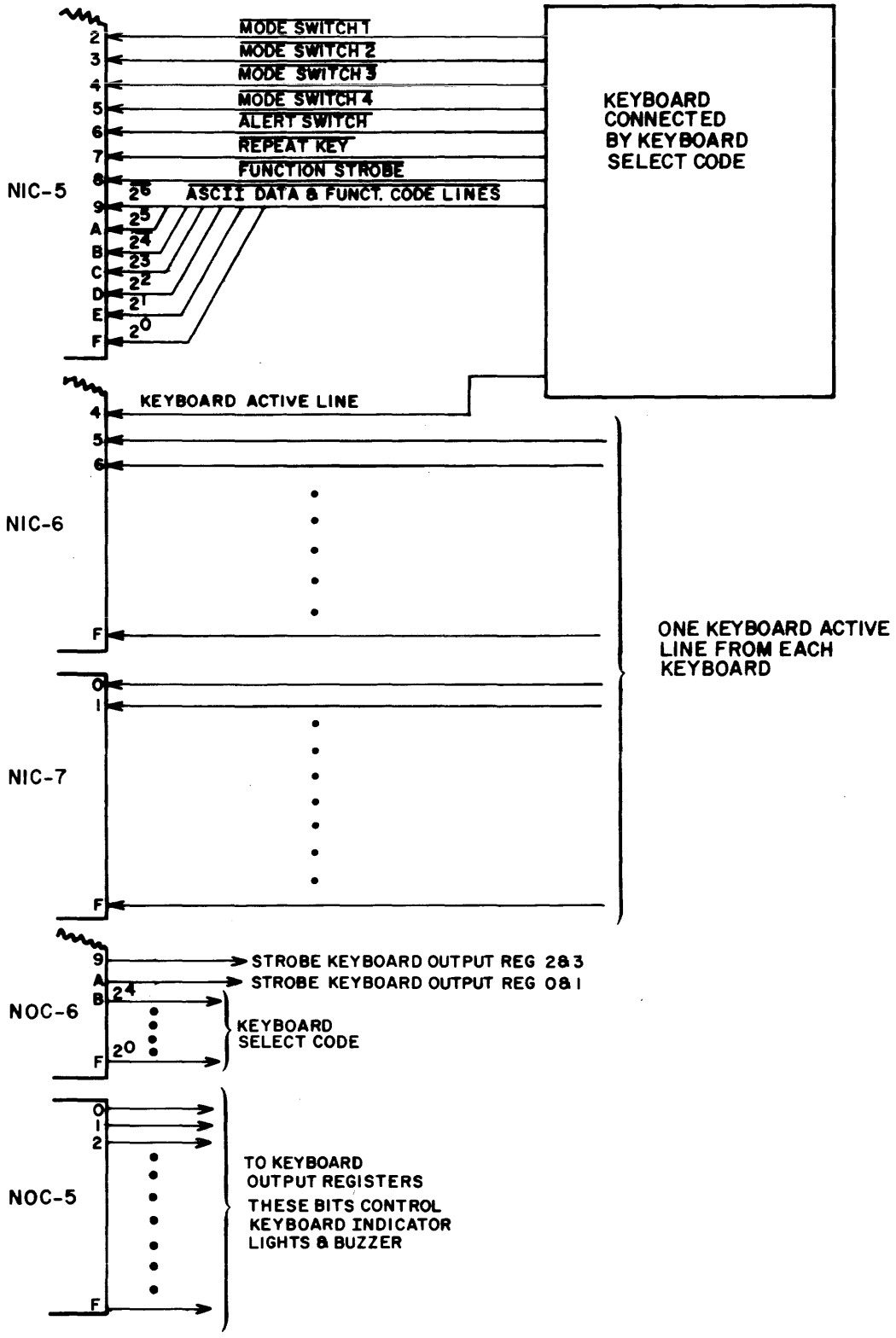


Figure 4-9. Keyboard/Normal Channel Signals

KEYBOARD SELECT CODE

The 5-bit select code is used to connect one of the 28 possible keyboards to NIC-6. A keyboard remains connected as long as the corresponding select code is held in NOC-6.

NORMAL INPUT CHANNEL 5 SIGNALS

These signals come from the keyboard designated by the select code issued on NOC-6. All other keyboards are disconnected from NIC-5.

FUNCTION/ASCII DATA LINES

These lines carry two types of information.

1. If the operator presses one of the ASCII keys, a 7-bit ASCII code appears on the input lines.
2. If the operator presses one of the function keys, a 7-bit binary function code appears on the input lines.

The function strobe line indicates which of these two types of data is present on the function/ASCII data lines. A 1 indicates a function code; 0 indicates an ASCII code.

Tables 4-1 and 4-2 list the keyboard ASCII and function codes.

Even though some of the function keys carry functional names (for example, LINE CLEAR), none of the function keys initiate any function or operation directly. The activity initiated by each function key is software-assigned. The SCU program must interpret function codes and perform the steps necessary to carry out the assigned activity for a given code.

REPEAT, ALERT, AND MODE LINES

Each of these lines is activated by the key with the same name. When the operator presses one of these keys, the corresponding signal remains active as long as the key is held down. Like the function keys, these keys do not directly initiate or control any function or operation. The SCU program must sense these signals and then initiate the assigned function or set up the assigned operating mode.

TABLE 4-1. ASCII CODING SET FOR SDU CRT DISPLAY

Bit 7 →					0	0	0	0	1	1	1	1
Bit 6 →					0	0	1	1	0	0	1	1
Bit 5 →					0	1	0	1	0	1	0	1
Bit 4	Bit 3	Bit 2	Bit 1	Column	0	1	2	3	4	5	6	7
0	0	0	0	0			SP	0	@	P		p
0	0	0	1	1			!	1	A	Q	a	q
0	0	1	0	2			"	2	B	R	b	r
0	0	1	1	3			#	3	C	S	c	s
0	1	0	0	4			$	4	D	T	d	t
0	1	0	1	5			%	5	E	U	e	u
0	1	1	0	6			$	6	F	V	f	v
0	1	1	1	7			'	7	G	W	g	w
1	0	0	0	8			(	8	H	X	h	x
1	0	0	1	9			)	9	I	Y	i	y
1	0	1	0	10			*	:	J	Z	j	z
1	0	1	1	11			+	;	K	[	k	{
1	1	0	0	12			,	<	L	\	l	¦
1	1	0	1	13			-	=	M	]	m	}
1	1	1	0	14			.	>	N	^	n	~
1	1	1	1	15			/	?	O	_	o	DEL

Not Used

END of Rec $1F_{16}$

END of File 1C

* Space or blank

TABLE 4-2. KEYBOARD FUNCTION CODES

Function	Binary Code $2^6 \longleftarrow 2^0$	Hexadecimal Equiv
SEND	000 0001	01
CLEAR	000 0000	00
LINE CLR	000 0010	02
RESET	000 1000	08
↑	000 0101	09
→	000 0110	0A
←	000 0011	03
↓	000 0111	07
NEW LINE	000 0100	04
F10	000 1001	09
F1	000 1010	0A
F2	000 1011	0B
F3	000 1100	0C
F4	000 1101	0D
F5	000 1110	0E
F6	000 1111	0F
F7	001 0000	10
F8	001 0001	11
F9	001 0010	12

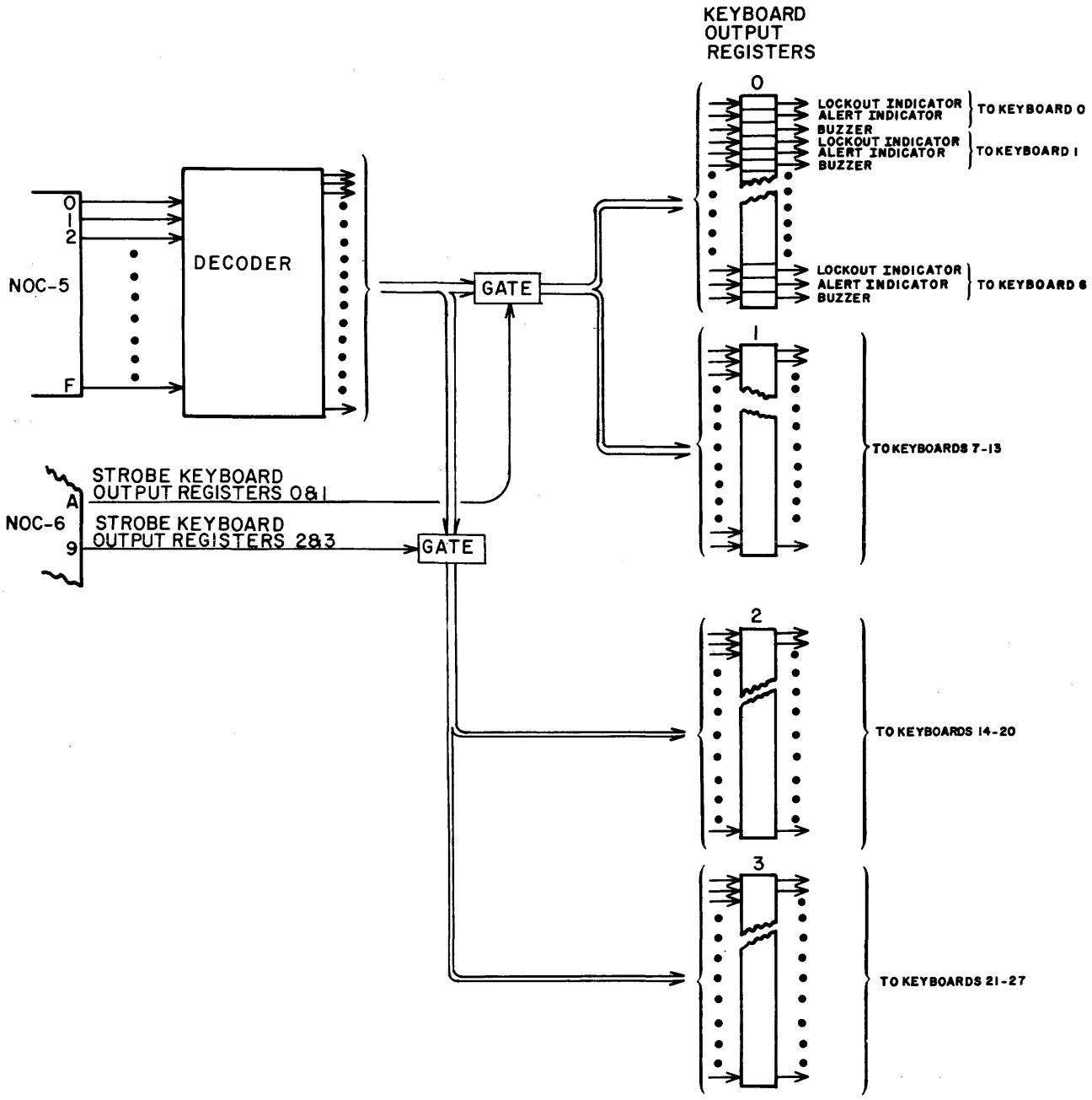


Figure 4-10. Keyboard Indicator Control

KEYBOARD INDICATOR LIGHTS AND ALARM BUZZER

Each keyboard has an ALERT buzzer and two indicators, the LOCKOUT light and the ALERT light, that are turned ON or OFF by normal channel bits. Figure 4-10 shows the normal channel control arrangement.

The two keyboard output registers contain three bits for each of the 28 possible keyboards. Each bit controls one indicator light or buzzer. To turn ON a light or buzzer, the SCU program must set the corresponding bit in the keyboard output register. The light or buzzer then remains ON until the SCU program clears this output register bit.

To set bits in one of the keyboard output registers, the proper bit-select pattern must first be set up in NOC-5. Then, one of the two strobe bits, NOC-6, bit 9 or NOC-6, bit 10, must be set and then cleared.

Table 4-3 shows the NOC-5 select bits. To select a specific light or buzzer, two NOC-5 bits must be set. For example, bits 0 and bit A must be set to select the LOCKOUT light for keyboard number 2.

The strobe bit for keyboard groups 1 and 0 (NOC-6, bit A) must then be pulsed to set the selected bit in the keyboard output register. This action turns ON the LOCKOUT light for keyboard 2. At the same time all unselected bits in the keyboard output register for groups 1 and 2 are cleared. If the other strobe bit (NOC-5, bit 9) were pulsed, this same combination of bits (NOC-5, bit 0 and bit A set) would turn ON the LOCKOUT light for keyboard 16. Two or more indicators or alarms can be turned ON at the same time if the correct pattern is set up on NOC-5.

Turning ON the LOCKOUT light does not in any way physically lock the keyboard or prevent the keyboard from sending data. It is up to the SCU software to lockout a keyboard by simply not reading in data from it.

TABLE 4-3. KEYBOARD INDICATOR SELECT BITS

Bit 0 - Selects LOCKOUT light group 0	Selects LOCKOUT light group 2
1 - Selects LOCKOUT light group 1	Selects LOCKOUT light group 3
2 - Selects ALERT light group 0	Selects ALERT light group 2
3 - Selects ALERT light group 1	Selects ALERT light group 3
4 - Selects ALERT alarm group 0	Selects ALERT alarm group 2
5 - Selects ALERT alarm group 1	Selects ALERT alarm group 3
6 - Not used	Not used
7 - Not used	Not used
8 - Not used	Not used
9 - Selects keyboard 1 or 8	Selects keyboard 15 or 22
A - Selects keyboard 2 or 9	Selects keyboard 16 or 23
B - Selects keyboard 3 or 10	Selects keyboard 17 or 24
C - Selects keyboard 4 or 11	Selects keyboard 18 or 25
D - Selects keyboard 5 or 12	Selects keyboard 19 or 26
E - Selects keyboard 6 or 13	Selects keyboard 20 or 27
F - Selects keyboard 7 or 14	Selects keyboard 21 or 28
Bit A of NOC-6 used to strobe selects bits to SDU.	Bit 9 of NOC-6 used to strobe selects bits to SDU.

NOTE

Group 0 is keyboards 1-7
Group 1 is keyboards 8-14
Group 2 is keyboards 15-21
Group 3 is keyboards 22-28

MEMORY MAP

Figure 4-11 shows the assignment of buffer areas and cursor address areas within a 4K segment of SDU memory. The SDU can contain up to four 4K segments. Each one serves a group of seven display units. Memory allocation is the same within each of the four segments.

SDU Memory Address				
X000-X1FF	512 words	monitor unit 1	A Field	Buffer Areas
X200-X3FF	512 words	monitor unit 2		
X400-X5FF	512 words	monitor unit 3		
X600-X7FF	512 words	monitor unit 4		
X800-X9FF	512 words	monitor unit 5		
XA00-XBFF	512 words	monitor unit 6		
XC00-XDFF	512 words	monitor unit 7		
XE00-XE3F	64 words	monitor unit 1	B Field	
XE40-XE7F	64 words	monitor unit 2		
XE80-XEBF	64 words	monitor unit 3		
XECO-XEFF	64 words	monitor unit 4		
XF00-XF3F	64 words	monitor unit 5		
XF40-XF7F	64 words	monitor unit 6		
XF80-XFBF	64 words	monitor unit 7		
XFC0-XFC7	8 words	monitor unit 1	A Field	Cursor Address Area
XFC8-XFCF	8 words	monitor unit 2		
XFD-XFD7	8 words	monitor unit 3		
XFD8-XFDF	8 words	monitor unit 4		
XFE0-XFE7	8 words	monitor unit 5		
XFE8-XFEF	8 words	monitor unit 6		
XFF0-XFF7	8 words	monitor unit 7		
XFF8	1 word	monitor unit 1	B Field	
XFF9	1 word	monitor unit 2		
XFFA	1 word	monitor unit 3		
XFFB	1 word	monitor unit 4		
XFFC	1 word	monitor unit 5		
XFFD	1 word	monitor unit 6		
XFFE	1 word	monitor unit 7		
XFFF	1 word	unused		

Figure 4-11. SDU Memory Map

NORMAL CHANNEL CONTROL BITS

Figure 4-12 shows the normal channel bits associated with the SDU coupler, SDU, and keyboards. Each of these bits is described on the following pages in Tables 4-4 and 4-5.

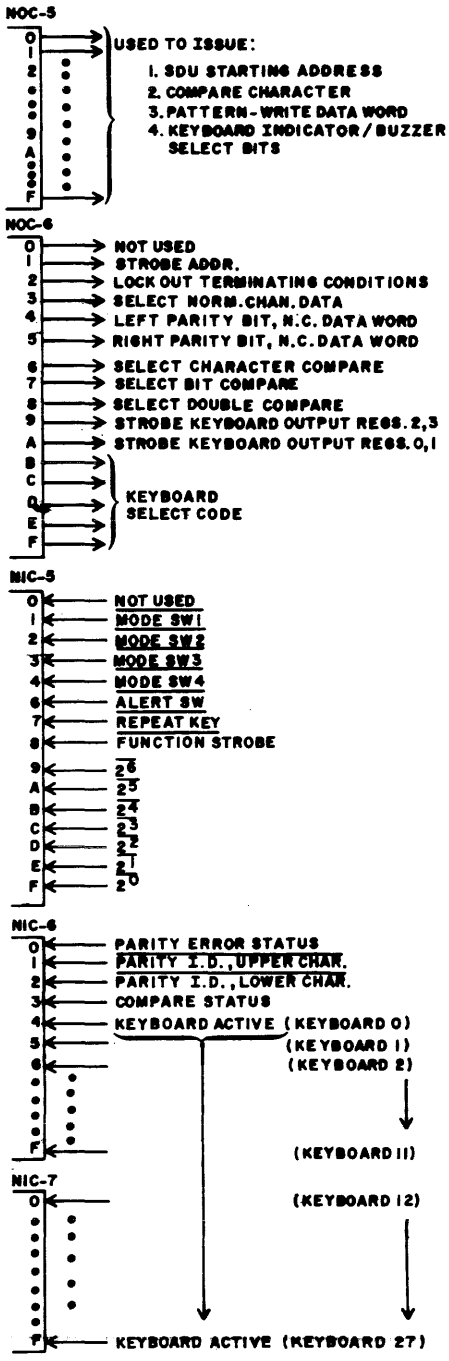


Figure 4-12. SDU and SDU Coupler Normal Channel Bits

USES OF NORMAL OUTPUT CHANNEL 5

Normal output channel 5 is used to issue four different types of information. Bits on normal output channel 6 are used to identify the type of information being transmitted on NOC-5.

1. SDU starting address

   A 15 bit starting address (Figure 4-13) must be issued to the SDU via NOC-5 before the start of a block-input, block-output, or pattern-write operation. NOC-6, bit 3 (select normal channel data) must also be set to prepare the coupler to issue an address. Setting, then immediately clearing NOC-6, bit 1 (strobe address) transfers the address to the SDU.

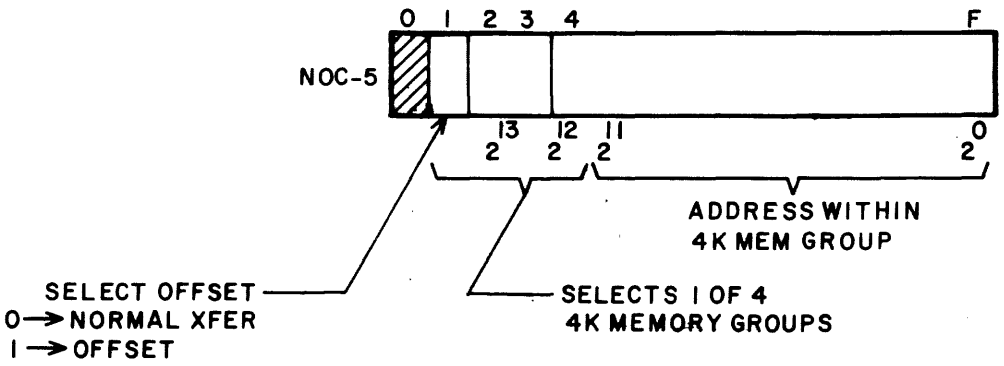


Figure 4-13. Address Format

2. Compare character

   When the character compare or double compare terminating conditions are selected, a compare character must be held in NOC-5 throughout a block-input or block-output operation. NOC-6, bit 6 (select character compare) or NOC-6, bit 8 (select double compare) must also be set. As shown in Figure 4-14, the same compare character must be repeated in the upper and lower halves of NOC-5.

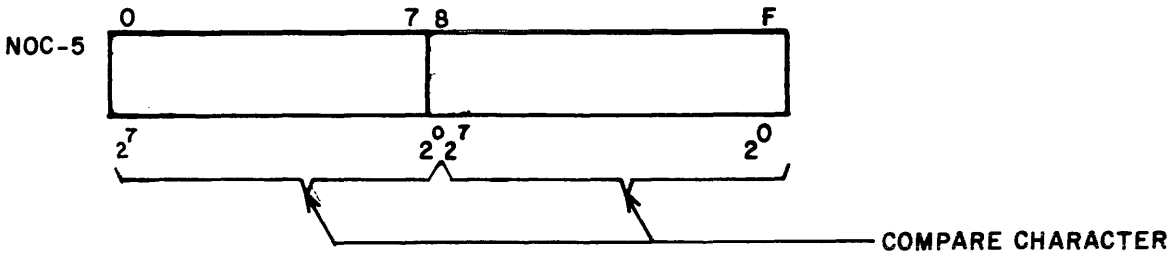


Figure 4-14. Compare Character Format

If character compare is selected, any 7-bit character can be issued on NOC-5. If double compare is selected, either the end-of-file code ($1C_{16}$) or end-of-record code ($1F_{16}$) must be placed in NOC-5. It makes no difference which of these codes is issued; if one of them is present in NOC-5, the coupler terminates a block-input or block-output operation if either code appears in the data stream.

3. Pattern - write data word

   During a pattern-write operation, the pattern word to be written repetitively in SDU memory must be held in NOC-5 throughout the operation. Also, NOC-6, bit 3 (select normal channel data) must be set throughout the operation to gate the pattern word from NOC-5 to the SDU data lines.

4. Keyboard indicator/buzzer select bits

   The bits listed in Table 4-3 are used to turn on keyboard indicator lights and alarm buzzers. One of two keyboard output register strobe bits (NOC-6,bit 9 or bit A) must be set, then cleared to transfer the select bits from NOC-5 to the keyboard output registers. Also, NOC-6,bit 3 (select normal channel data) must be set.

TABLE 4-4. NOC-6 BIT DESCRIPTIONS FOR SDU COUPLER INTERFACE

Channel	Bit	Name	Function
NOC-6	$1(2^{14})$	Strobe address	This set/clear† bit strobes the starting address on NOC-5 into SDU memory prior to a block transfer operation.
NOC-6	$2(2^{13})$	Lockout terminating conditions	Normally, a compare hit or parity error terminates a block transfer operation. When set, this bit causes the coupler to overlook these terminating conditions. This bit is used primarily as a maintenance aid.
NOC-6	$3(2^{12})$	Select NC data	When set, this bit connects NOC-5, bits 0-F, to the SDU data lines. The normal path from the BC block transfer channel to the SDU data lines is disconnected. This bit must be set (1) when a starting address is issued to the SDU via NOC-5, (2) during a pattern-write operation, or (3) during an output operation of the keyboard output register. It must be clear during a normal output-block transfer operation.
NOC-6	$4(2^{11})$	NC data PBL (parity bit, left character)	This bit is used to add the proper parity to the left half of a 16-bit word when NOC-5 is used for data in a pattern-write operation.
NOC-6	$5(2^{10})$	NC data PBR (parity bit, right character)	This bit is used similar to that described in bit 4, above, except it is used with the right half of the 16-bit pattern word.

† A set/clear bit must be set and then immediately cleared to initiate its function.

TABLE 4-4. NOC-6 BIT DESCRIPTIONS FOR SDU COUPLER INTERFACE (Cont'd)

<table>
<tr><th>Channel</th><th>Bit</th><th>Name</th><th>Function</th></tr>
<tr><td>NOC-6</td><td>6(2^9)</td><td>Select character compare</td><td>While set, this bit selects the character compare terminating condition.</td></tr>
<tr><td>NOC-6</td><td>7(2^8)</td><td>Select bit compare</td><td>While set, this bit selects the bit compare terminating condition.</td></tr>
<tr><td>NOC-6</td><td>8(2^7)</td><td>Select double compare</td><td>While set, this bit selects the double compare terminating condition.</td></tr>
<tr><td>NOC-6</td><td>9(2^6)</td><td>Strobe keyboard output register 2, 3</td><td>Setting this bit gates NOC-5 data into the group 2 or 3 keyboard output register. Each bit in the keyboard output register controls an indicator light or buzzer on one of the keyboards.</td></tr>
<tr><td></td><td>A(2^5)</td><td>Strobe keyboard output register 0, 1</td><td>Setting this bit gates NOC-5 data into the group 0 or 1 keyboard output register.</td></tr>
<tr><td>NOC-6</td><td>B (2^4)
↓
F (2^0)</td><td>Select keyboard input</td><td>(2^4)
↓
(2^0)
Bits 2^3 and 2^4 are used to select one of four groups of keyboards, and bits 2^0, 2^1, and 2^2 are used to select one of seven keyboards within a group.
<table>
<tr><th>Bit 2^4</th><th>Bit 2^3</th><th>Group selected</th><th>Bit 2^0</th><th>Bit 2^1</th><th>Bit 2^2</th><th>Keyboard selected</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>2</td></tr>
<tr><td>1</td><td>0</td><td>2</td><td>0</td><td>0</td><td>1</td><td>3</td></tr>
<tr><td>1</td><td>1</td><td>3</td><td>1</td><td>0</td><td>1</td><td>4</td></tr>
<tr><td></td><td></td><td></td><td>0</td><td>1</td><td>1</td><td>5</td></tr>
<tr><td></td><td></td><td></td><td>1</td><td>1</td><td>0</td><td>6</td></tr>
<tr><td></td><td></td><td></td><td>1</td><td>1</td><td>1</td><td>7</td></tr>
</table></td></tr>
</table>

TABLE 4-5. NIC -5, -6, AND -7 BIT DESCRIPTIONS FOR SDU COUPLER INTERFACE

Channel	Bit	Name	Function
NIC-5	2 (2^{13}) down to 5 (2^{10})	Not mode switch 1, 2, 3, 4	These four bits, sent over four lines to the SCU, indicate one of the four mode switches on the selected keyboard that was pressed. A 0 indicates the switch is ON.
NIC-5	6 (2^9)	Not ALERT switch	This status bit indicates the state of the ALERT switch on the selected keyboard. A 0 indicates the switch is ON.
NIC-5	7 (2^8)	Not RPT key	When 0, this bit indicates that the operator is pressing the repeat key on the selected keyboard. No keyboard data accompanies the repeat bit; it is up to the software to repeat the previously typed character in the next character position in SCU memory.
NIC-5	8 (2^7)	Not function strobe	When a 0, this status bit indicates that bits 9 through F of NIC-5 contain a function code from the function keys of a selected keyboard rather than an ASCII-coded data character. A 1 indicates ASCII data.
NIC-5	9 (2^6) down to F (2^0)	Not keyboard data	These input bits carry one of two types of information from the selected keyboard. 1. An ASCII-coded data character each time one of the ASCII character keys are pressed. 2. A hexadecimal function code each time one of the function keys is pressed.

TABLE 4-5. NIC-5, -6, AND -7 BIT DESCRIPTIONS FOR SDU COUPLER INTERFACE (Cont'd)

Channel	Bit	Name	Function
			Information on these bits is inverted (in ones complement form).
NIC-6	0	Parity error status	This status bit indicates that a parity error occurred in a 16-bit word read from SDU memory during an input-block transfer operation.
NIC-6	1, 2	Not parity I.D.	These two status bits indicate that a parity error exists in bits received with the 16 bits of data from the SDU. Bit 1 indicates the parity received with the left byte. Bit 2 indicates the parity received with the right byte. These bits can be sampled after a parity error occurs.
NIC-6	3	Compare status	This status bit indicates that a compare hit has occurred for one of the previously selected terminating conditions; that is, bit compare, character compare, or double compare.
NIC-6	4 ↓ F	Keyboard active (keyboard 1) ↓ Keyboard active (keyboard 12)	There is one of these bits for each keyboard. A keyboard active bit goes to 1 when one of the following actions occurs at the corresponding keyboard.
NIC-7	0 ↓ F	Keyboard active (keyboard 13) ↓ Keyboard active (keyboard 28)	1. A data key is pressed 2. A function key is pressed 3. The ALERT key is pressed The bit returns to 0 as soon as the depressed key is released.

PROGRAMMING SEQUENCES

The following paragraphs illustrate the programming steps necessary to control the SDU through the SDU coupler.

BASIC STEPS

A. Connect SDU coupler

1. Place SDU coupler connect code (10_2) in NOC-2, bits 9 and A.

   The connect code must be held in NOC-2 throughout a block-input or block-output transfer with the SDU.

B. Issue SDU starting address

1. Place 15-bit address in NOC-5, bits 1 through F.
   (The offset bit, bit 1, must be 0 unless an offset operation is desired.)
2. Set the select normal channel data bit (NOC-6, bit 3).
3. Set, then immediately clear the strobe address bit (NOC-6, bit 1).
4. Clear the select normal channel data bit (NOC-6, bit 3).

C. Select character compare terminating condition

1. Place terminating character in NOC-5.
   (The same 8-bit character must be repeated in the upper and lower halves of NOC-5.)
2. Set the select-character-compare bit (NOC-6, bit 6).

D. Select double compare terminating condition

1. Place end-of-record ($1F_{16}$) or end-of-file ($1C_{16}$) character in the upper and lower halves of NOC-5.
2. Set the select-double-compare bit (NOC-6, bit 8).

E. Select bit compare

1. Set the select-bit-compare bit (NOC-6, bit 7).

F. Output-block transfer to SDU memory.

1. Connect (refer to B)
2. Issue SDU starting address (refer to B).
3. Select optional terminating conditions (refer to C, D, or E).
4. Place word count (in ones complement form) in buffer controller A register.
5. Execute output-block transfer operation. Step 5 starts data transfer and will continue until one of the following occurs.
   - The word count specified in the block transfer instruction is reached.
   - A terminating character appears in the data stream (if one of the optional terminating conditions was selected).

G. Input block transfer from SDU memory

1. Connect (refer to B)
2. Issue SDU starting address (refer to B).
3. Select optional terminating conditions (refer to C, D, or E).
4. Place word count (in ones complement form) in buffer controller A register.
5. Execute input-block transfer instruction. This starts data transfer. Terminating conditions are one of the following.
   - Word count satisfied
   - Parity error in data from SDU
   - Terminating character in data stream (optional)

H. Pattern-write operation

1. Connect (refer to B)
2. Issue SDU starting address (refer to B).

   Omit step B4 in the addressing sequence. The select normal channel data bit (NOC-6, bit 3) must remain set during a pattern write.
3. Place 16-bit pattern word in NOC-5.
4. Generate parity bits for upper and lower characters of pattern word. Set or clear the two pattern-write parity bits:

   Left parity bit, NOC-6, bit 4
   Right parity bit, NOC-6, bit 5

5. Place word count (complement form) in buffer controller A register.

6. Execute output-block transfer instruction to start repetitive transfer of the pattern word. The operation continues until the word count is reached.

NOTE

After an input or output block transfer operation terminates, the BC condition bit can be tested to determine what caused termination.

- If condition bit = 1, the word count was reached.
- If condition bit = 0, one of the early terminating conditions occurred.

KEYBOARD

I. Input from keyboard

1. Scan keyboard active bits (NIC-6 and -7).

2. When active (1) bit is found, connect corresponding keyboard by placing connect code in NOC-6, bits B through F.

3. Input keyboard character on NIC-5, bits 9 through F and function strobe (NIC-5, bit 8).

4. Test function strobe bit to determine if keyboard character is an ASCII data character or a function code: 1 = data, 0 = function code.

J. Activate keyboard lights, buzzer

1. Set select normal channel data bit (NOC-6,bit 3).

2. Set up appropriate select bits in NOC-5. Refer to Table 4-3.

3. Set, then clear keyboard strobe bit (NOC-6,bit 9 or bit A).

OPERATION 5

GENERAL

This section contains operating procedures, functions of the switches, controls, and indicators used during operation and illustrations, as appropriate, of various operator panels used in the peripheral stations. For more detail concerning the operation of a particular peripheral unit refer to the appropriate manual listed in the preface of this manual.

STATION CONTROL UNIT SWITCHES AND INDICATORS

POWER SEQUENCE

The POWER ON/OFF switches and TEMPERATURE WARNING indicator are located to the right side of the drop-down station control unit panel.

The POWER ON switch is the operator's power control switch. Additional power control includes the power supply circuit breaker and the power disconnect switch. Pressing the POWER ON switch brings the SCU to a STANDBY condition with the BC stopped and master cleared.

The next step is to autoload the station; this is done either remotely or from the control panel of the SCU. Autoloading is described later in this section.

The normal sequence to turn the power off in the SCU is to press the POWER OFF switch.

TEMPERATURE WARNING INDICATOR

The high temperature warning indicator (TEMP WARN) is thermostat controlled and lights if the input-air supply reaches 100°F. If the input-air supply temperature reaches 110°F, a second thermostat closes and removes power from the SCU.

SCU CONTROL PANEL

The SCU control panel (Figure 5-1) is primarily used for maintenance and start-up purposes. It is not intended to be an operator's panel; the keyboard/display is the SCU operator's console.

The control panel consists of 15 switches and two indicators. Eleven switches interface the BC either directly or indirectly and are disabled by the PANEL LOCKOUT switch. Four switches are lockout switches for certain devices and interfaces and are not disabled by the PANEL LOCKOUT switch (a key-operated switch).

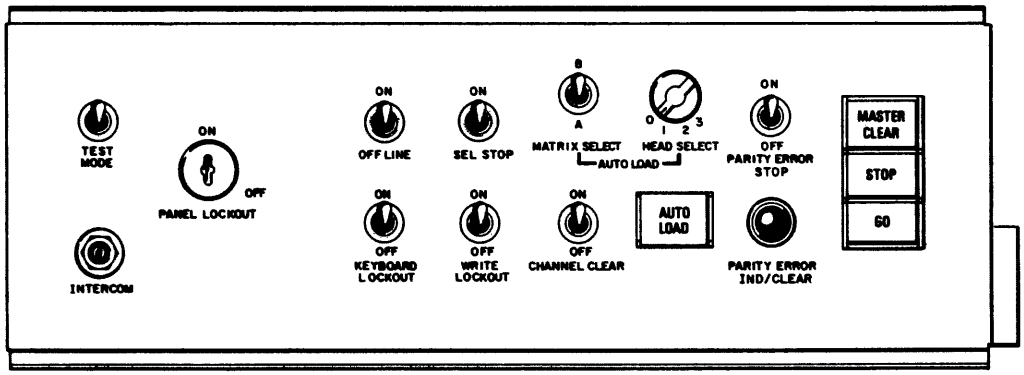


Figure 5-1. Station Control Unit Control Panel

BUFFER CONTROLLER INTERFACE SWITCHES

AUTOLOAD OPERATION

The SCU autoload sequence can be initiated remotely or from the SCU control panel. For remote operation, the STAR data channel provides a line called the external flag line. When the external flag signal is received by the SCU coupler interface, an autoload sequence is initiated in the drum.

To autoload from the SCU control panel, enable the panel by turning the key-operated PANEL LOCKOUT switch to the OFF position. The microdrum track to be used for loading is then selected using the MATRIX SELECT and HEAD SELECT switches (these switches apply to remote autoloading as well as the procedure using the SCU control panel). After the track selection is made, an autoload sequence is initiated by pressing the AUTOLOAD switch.

The MATRIX SELECT switch selects one of two head groups in the microdrum. In position A, this switch selects a head group which is protected by the WRITE LOCKOUT switch. In position B, a head group is selected which is not write-protected.

The HEAD SELECT switch is a 4-position rotary switch which selects one of four heads in the head group selected by the MATRIX SELECT switch.

AUTOLOAD SWITCH

The AUTOLOAD switch initiates the autoload sequence. When this switch is pressed, the microdrum hardware reads one track from the drum and transfers the information to the BC, starting at address 0000_{16}, via the SCU coupler hardware. At the completion of the autoload sequence, the BC jumps to address 0001_{16} and begins execution of the program just loaded.

TEST MODE SWITCH

The TEST MODE switch is used for maintenance purposes only (writing the clock, checking out the drum display hardware, etc.). This switch, in the ON position, initiates an autoload sequence every third drum revolution. The sequence is identical to that performed by pressing the AUTOLOAD switch.

BUFFER CONTROLLER/STATION CONTROL INTERFACE SWITCHES

Seven switches interface the BC's station control interface, four through the pluggable station control interface and three through the nonpluggable interface. All seven switches are disabled when the PANEL LOCKOUT switch is in the ON position.

PLUGGABLE STATION CONTROL INTERFACE SWITCHES

The SEL STOP, CHANNEL CLEAR, PARITY ERROR STOP, PARITY ERROR CLEAR switches, and the STOP and PARITY ERROR indicators are connected to the pluggable station control interface. A cable is provided to plug into the C connector of the BC (connector is located under the sliding table top). This cable must be disconnected when operating with the customer engineering maintenance panel. These switches and indicators are duplicated on the customer engineering maintenance panel.

SEL STOP SWITCH

When the PANEL LOCKOUT switch is ON, the BC does not stop regardless of the position of the SEL STOP switch. When the PANEL LOCKOUT switch is OFF, the BC stops on selective stop instructions if the STOP switch is ON.

PARITY ERROR STOP SWITCH

With the PANEL LOCKOUT switch OFF, the BC stops on parity errors only if the PARITY ERROR STOP switch is in the ON position. With the PANEL LOCKOUT switch in the ON position, the BC stops on all parity errors regardless of the position of the PARITY ERROR STOP switch.

PARITY ERROR IND/CLEAR SWITCH AND INDICATOR

The PARITY ERROR indicator displays parity errors regardless of the setting of the PANEL LOCKOUT switch, but the PARITY ERROR CLEAR switch is only operable when the PANEL LOCKOUT switch is OFF. The PARITY ERROR CLEAR switch and indicator are in the same physical assembly.

CHANNEL CLEAR SWITCH

The CHANNEL CLEAR switch is operable only when the PANEL LOCKOUT switch is OFF. In the ON position, the CHANNEL CLEAR switch holds a clear on all BC normal channels.

STOP INDICATOR

The STOP indicator is part of the STOP switch assembly. The indicator lights whenever the BC is not running, regardless of the position of the PANEL LOCKOUT switch.

NON-PLUGGABLE STATION CONTROL INTERFACE SWITCHES

Three switches interface the nonpluggable station control interface of the BC: GO, STOP, and MASTER CLEAR. These switches are operable only when the PANEL LOCKOUT switch is OFF.

GO SWITCH

The GO switch is used to start the BC. The switch lights when the BC is operating.

STOP SWITCH

The STOP switch stops the BC at the end of an instruction cycle.

MASTER CLEAR SWITCH

The MASTER CLEAR switch, when pressed, stops and then clears the BC.

LOCKOUT SWITCHES

The LOCKOUT switches are designed to protect certain on-line devices or interfaces from accidental misuse.

PANEL LOCKOUT

The key-operated PANEL LOCKOUT switch in the ON position disables the following control panel switches: MASTER CLEAR, STOP, GO, PARITY ERROR STOP, PARITY ERROR IND/CLEAR, AUTOLOAD, CHANNEL CLEAR, SEL STOP, AND TEST MODE.

WRITE LOCKOUT SWITCH

When the WRITE LOCKOUT switch is ON, certain head groups in the microdrum cannot be used for writing. The head addresses that are protected include 00_{16} to $0B_{16}$ and 10_{16} to $1B_{16}$. The head addresses not affected by the WRITE LOCKOUT switch are 20_{16} to $2B_{16}$.

KEYBOARD LOCKOUT SWITCH

The KEYBOARD LOCKOUT switch sets a bit in the BC normal channel which can be sensed by the station software. The software will then lock out that portion of the display keyboard which is to be protected on that station. The LOCKOUT light on the display keyboard will be lighted when this switch is ON.

OFF LINE SWITCH

The OFF LINE switch disables the SCU coupler. Control signals into the coupler are disabled including external flag, which is the remote autoload signal. However, the control panel autoload operation via the coupler is still operable.

PORTABLE MAINTENANCE CONSOLE CONTROL PANELS

Table 5-1 lists the controls, switches, and indicators and the function of each on the control panel of the paper tape input type portable maintenance console. Table 5-2 provides similar information for the cassette input portable maintenance console. For further details concerning operation of either console, refer to sections 2 and 3 of the appropriate CE manual listed in the preface of this manual. Figure 5-2 illustrates the paper tape input console's control panel, and Figure 5-3 illustrates the cassette input console's control panel.

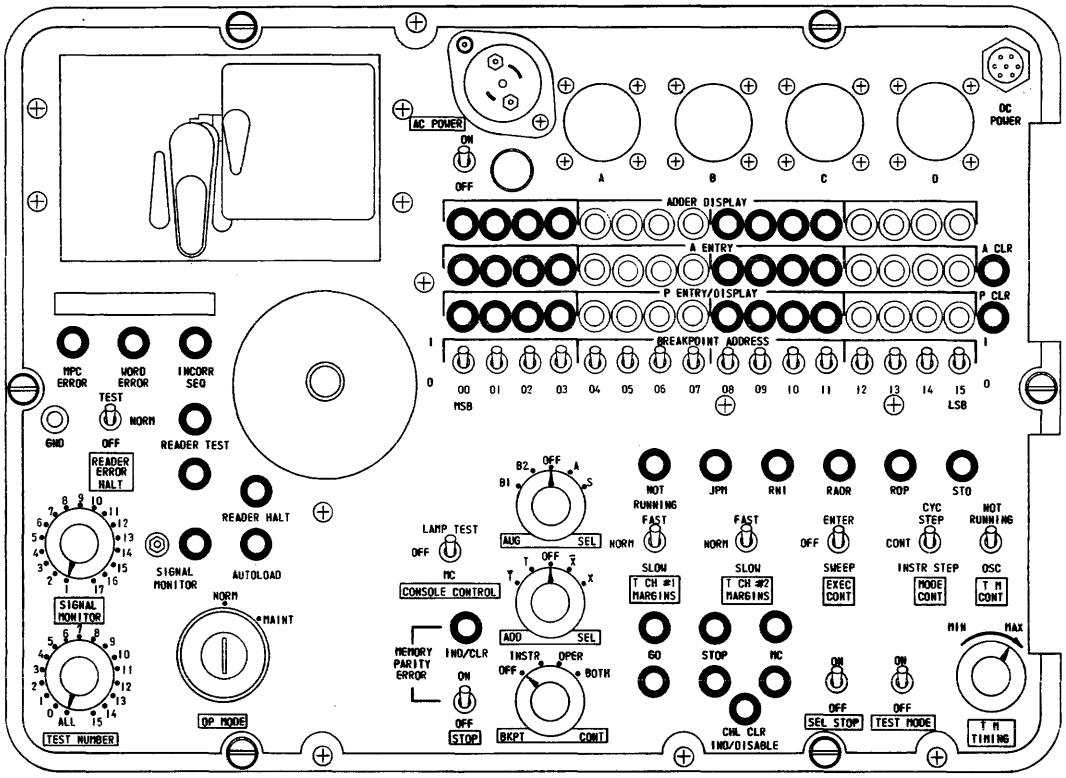


Figure 5-2. Portable Maintenance Console Control Panel (TF201)

TABLE 5-1. PAPER TAPE CONSOLE CONTROL PANEL CONTROLS

Component Name	Function
AC POWER switch	
ON	Applies both live and neutral lines (120-volt operation) or two live lines (220-volt operation) to the console's AC-operated devices.
OFF	Breaks both AC lines, provided the OP MODE switch is in the MAINT position.
ADDER DISPLAY BIT 00 through BIT 15	Sixteen indicators, any of which, when illuminated, indicates the presence of the corresponding bit being transmitted from the BC adder.
A ENTRY BIT 00 through BIT 15 (momentary)	Sixteen pushbuttons, any of which, when pressed, causes the corresponding bit of the BC A register to set (the bit becomes logical 1), provided the console is not in the autoload data entry mode or reader test, and the OP MODE switch is not in the NORM position.
A CLR momentary switch	Pressing this pushbutton switch clears the BC A register (all bits become logical 0), provided the console is not in the autoload data entry mode or reader test, and the OP MODE switch is not in the NORM position.
P ENTRY/DISPLAY BIT 00 through BIT 15 momentary switch/indicator	Sixteen pushbutton/switch indicators, any of which, when pressed, causes the corresponding bit of the BC P register to set (the bit becomes logical 1), provided the OP MODE switch is not in the NORM position. The associated indicator illuminates for as long as the bit is set.
P CLR (momentary)	Pressing this pushbutton/switch clears the BC P register (all bits become logical 0), provided the OP MODE switch is not in the NORM position.
MPC ERROR indicator	Indicates detection of a vertical tape parity error or the lack of an MPC sequence preceding a halt.
WORD ERROR indicator	Indicates detection of a horizontal tape parity error.
INCORR SEQ indicator	Indicates detection of a tape byte which is out of sequence.
BREAKPOINT ADDRESS BIT 00 through BIT 15 switches	
1	Signals presence of the corresponding breakpoint bit selection (one of 16) to the BC.
0	Signals absence of the corresponding breakpoint bit selection (one of 16) to the BC.

TABLE 5-1. PAPER TAPE CONSOLE CONTROL PANEL CONTROLS (Cont'd)

Component Name	Function
READER ERROR HALT switch	
OFF	Tape motion does not halt upon detection of errors but does halt upon detection of a reader halt control code (if preceded by an MPC sequence) or the out of tape condition. Errors are indicated but clear upon detection of a reader halt control code or the out of tape condition.
NORM	Tape motion halts upon console detection of byte parity error (word error), illegal halt (MPC error), message parity error (MPC error), or incorrect sequence. The associated error indicator will remain illuminated when the reader halts.
TEST	Tape motion, once initiated, is continuous and can be halted only upon opening the reader READY switch, placing the CONSOLE CONTROL switch in the MC position, depressing the READER HALT pushbutton, or moving the OP MODE switch to the NORM position.
READER TEST indicator	Indicates that the paper tape reader is being tested and all outputs to the BC are locked out.
READER TEST switch	Pressing this pushbutton initiates a test of the paper tape reader. All the normal conditions and lockouts of the autoload data entry mode are present during READER TEST, with the additional feature of locking out all outputs to the BC. AUTOLOAD, TEST MODE, STOP, GO, and MC are not locked out.
READER HALT momentary switch	Pressing this pushbutton unconditionally halts the paper tape reader and clears the associated control and switch lockouts. Clears reader error conditions.
SIGNAL MONITOR switch	Routes one of 17 internal console signals to the SIGNAL MONITOR indicator and SIGNAL MONITOR test point. The signals available at the 17 positions are listed below.
1	Tape 2^0
2	Tape 2^1
3	Tape 2^2
4	Tape 2^3
5	Tape 2^4
6	Tape 2^5
7	Tape 2^6
8	Tape 2^7
9	Ready
10	Resume II
11	Tape drive
12	Autoload
13	Ø1
14	MPC sequence

TABLE 5-1. PAPER TAPE CONSOLE CONTROL PANEL CONTROLS (Cont'd)

Component Name	Function
SIGNAL MONITOR switch	
15 16 17	Timing generator Start data Go
SIGNAL MONITOR test point	The signal selected by the SIGNAL MONITOR switch is routed to this test point. The signal is neither attenuated nor delayed.
SIGNAL MONITOR indicator	Indicates that the signal selected by the SIGNAL MONITOR switch is active and its time, if less than 20 milliseconds, has been extended to a minimum of 25 milliseconds.
AUTOLOAD momentary switch	Pressing this pushbutton initiates the autoload data entry mode and sets the channel clear (CLRCHL) to logical 0 (active). Also sets the logical lockout of the following switches and pushbuttons: STOP, GO, MC, TEST MODE, A ENTRY, A CLR, READER TEST and subsequent setting of autoload. Outputs to the BC are enabled during autoload.
TEST NUMBER switch	
0 through 15	Allows selection of a specific program segment on a paper tape by signaling the console to detect a specific test number in start data/test number bytes.
ALL	The test number is not required on a tape and all valid program segments will be read.
OP MODE switch	A locking switch. A key, which is required to operate this switch, can be inserted or removed only when the switch is in the NORM position.
NORM	When moved to this position, generates an internal console master clear, disables all console switches and pushbuttons, with the following exceptions: READER HALT, SIGNAL MONITOR, CONSOLE CONTROL. The BC is not under console control.
MAINT	Operation of the BC is subject to console control.
CONSOLE CONTROL switch	
MC (momentary)	The console's internal logic is returned to initial conditions, without affecting the BC.
OFF	When released, the switch returns to this center, inactive position.
LAMP TEST momentary	A test of all console indicators is performed.

TABLE 5-1. PAPER TAPE CONSOLE CONTROL PANEL CONTROLS (Cont'd)

Component Name	Function
MEMORY PARITY ERROR IND/CLR momentary switch/ indicator	When illuminated, indicates that a memory parity error has occurred in the BC; depressing the pushbutton causes a clear of the memory parity error.
MEMORY PARITY ERROR STOP switch	
ON	Enables the BC to halt upon detection of a memory parity error.
OFF	No effect on the BC's operation.
AUG SEL switch	Locked out when the BC is in the RUN status.
A S B1 B2	These four positions correspond to registers in the BC. When in any of these positions, the content of the corresponding register is gated to the BC's adder, where that value is treated as an augend.
OFF	The value used by the BC's adder as an augend is not affected by the console.
ADD SEL switch	Locked out when the BC is in the RUN status.
$\frac{T}{\overline{T}}$ $\frac{X}{\overline{X}}$	These four positions correspond to the true and complement values of the T translation and the X register. When in any of these positions, the corresponding value is gated to the BC's adder, where this value is treated as an addend.
OFF	The value used by the BC's adder as an addend is not affected by the console.
BKPT CONT switch	
INSTR	Enables comparison by the BC of instruction addresses with the address selected by the BREAKPOINT ADDRESS switches.
OPER	Enables comparison by the BC of operand addresses with the address selected by the BREAKPOINT ADDRESS switches.
BOTH	Enables comparison by the BC of both operand and instruction addresses with the address selected by the BREAKPOINT ADDRESS switches.
OFF	The BC is not affected by the settings of the BREAKPOINT ADDRESS switches.

TABLE 5-1. PAPER TAPE CONSOLE CONTROL PANEL CONTROLS (Cont'd)

Component Name	Function
NOT RUNNING indicator	Indicates that the BC is not executing a program.
JPM indicator	Indicates that a jump condition has been met in the BC.
RNI indicator	Indicates that the BC is operating in the read next instruction mode.
RADR indicator	Indicates that the BC is operating in the read address mode.
ROP indicator	Indicates that the BC is operating in the read operand mode.
STO indicator	Indicates that the BC is operating in the store mode.
T CH #1 MARGINS switch	
FAST	Enables the BC to subtract 1.5 nanoseconds from each step of its timing chain #1.
NORM	Enables the BC's timing chain #1 to operate a 20 nanoseconds per step.
SLOW	Enables the BC to add 1.5 nanoseconds to each step of its timing chain #1.
T CH #2 MARGINS switch	
FAST	Enables the BC to subtract 1.5 nanoseconds from each step of its timing chain #2.
NORM	Enables the BC's timing chain #2 to operate at 20 nanoseconds per step.
SLOW	Enables the BC to add 1.5 nanoseconds to each step of its timing chain #2.
EXEC CONT	
ENTER	Enables storage of data in the BC's core memory.
OFF	No effect on the BC's operation.
SWEEP	Enables information in the BC's core memory to be read from sequential locations, without execution of instructions which may be contained in those locations.

TABLE 5-1. PAPER TAPE CONSOLE CONTROL PANEL CONTROLS (Cont'd)

Component Name	Function
MODE CONT switch	
CYC STEP	Enables the BC to perform one pass through its main timing chain and halt, regardless of the number of passes which may be required to complete an instruction.
CONT	Enables the normal read and execution by the BC of instructions which may be contained within its core memory.
INSTR STEP	Enables the BC to read and execute one complete instruction and halt.
T M CONT switch	
NOT RUNNING	Repetition of test mode cycles is controlled by return of the RUN-ID signal as a logical 1 from the BC, following each cycle.
OSC	Duration of test mode cycles is determined by the setting of the T M TIMING control.
GO	Indicates that the GO command is active.
GO momentary switch	Pressing this pushbutton causes generation of a GO command (which is 8 microseconds in duration) to the BC. The associated indicator illuminates for 20 milliseconds. Locked out when the TEST MODE switch is in the ON position, the console is in autoload or the OP MODE switch is in the NORM position.
STOP indicator	Indicates that the STOP command is active.
STOP momentary switch	Pressing this pushbutton causes generation of a STOP command (which is 8 microseconds in duration) to the BC and sets the channel clear line (CLRCHL) to logical 0 (active). The associated indicator illuminates for 20 milliseconds. Locked out when the TEST MODE switch is in the ON position, the console is in autoload, or the OP MODE switch is in the NORM position.
MC indicator	Indicates that the MC command is active.
MC momentary switch	Pressing this pushbutton causes generation of an MC command (which is 8 microseconds in duration) to the BC, and sets the channel clear line (CLRCHL) to logical 0 (active). The associated indicator is illuminated for 20 milliseconds. Locked out when the TEST MODE switch is in the ON position, the console is in autoload or the OP MODE switch is in the NORM position.

TABLE 5-1. PAPER TAPE CONSOLE CONTROL PANEL CONTROLS (Cont'd)

Component Name	Function
CHL CLR IND/DISABLE momentary switch/ indicator	The indicator portion of this pushbutton/indicator is illuminated whenever the CLRCHL signal is active (logical 0). This condition is generated by: being in autoload; pressing the STOP or MC pushbuttons (provided test mode is inactive); exiting from test mode. Pressing the pushbutton or the OP MODE switch in the NORM position causes the CLRCHL signal to become logical 1 and extinguishes the associated indicator.
SEL STOP switch	
ON	Causes the BC to halt upon detection of any instruction word, read from its memory, which contains 00XX.
OFF	No effect on BC operation.
READY switch	
OPEN	The reader starwheels are up and tape motion is disabled.
CLOSED	The reader starwheels are down and make contact with the paper tape or the read head spindle (whichever is exposed).
TEST MODE switch	
ON	Causes a sequence of STOP, MC, and GO commands, in that order, to be sent to the BC. The sequence is repeated at a rate determined by either the setting of the T M TIMING control or the rate at which the BC returns the RUN-ID signal; one of these two parameters is selected by the T M CONT switch. GO, STOP, and MC pushbuttons are logically locked out.
OFF	Test mode sequences are disabled after the next MC is generated. When switched from ON to OFF, this switch causes the channel clear line (CLRCHL) to be set to logical 0 (active).
T M TIMING control	
MAX to MIN	If the T M CONT switch is in the OSC position, this control allows variation of the test mode cycle repetition rate between the nominal limits of 17 KHz to 450 Hz.

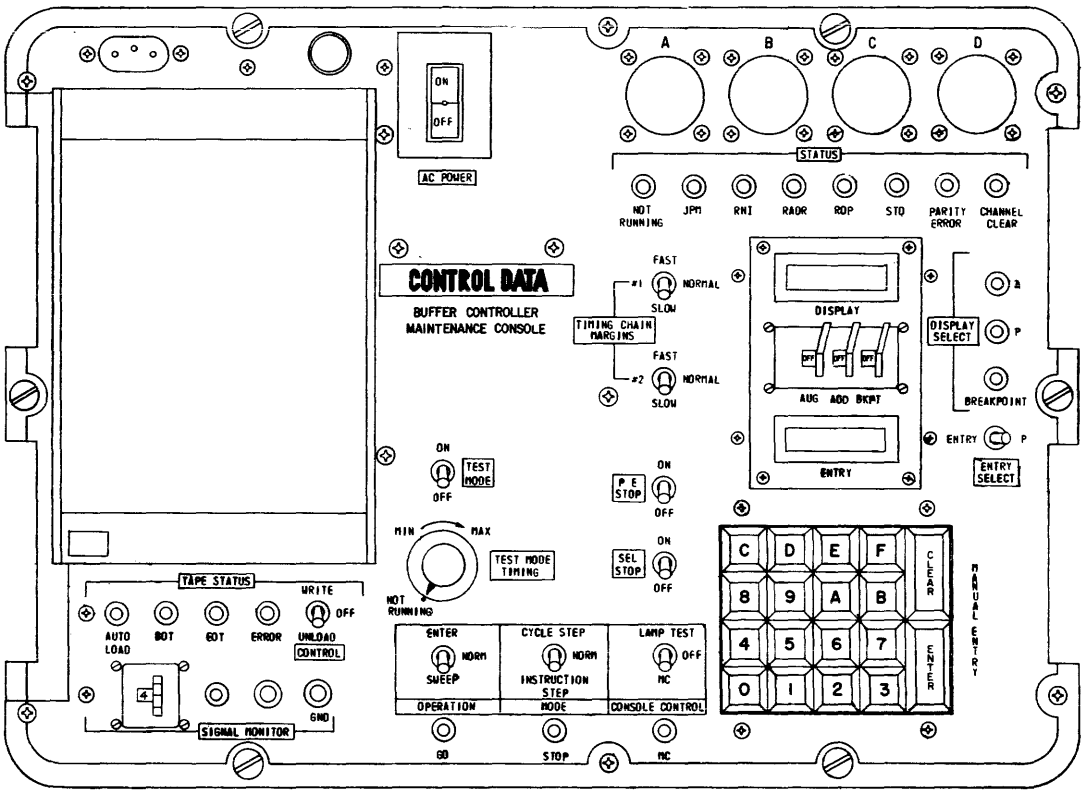


Figure 5-3. Portable Maintenance Console II Control Panel (TF204)

TABLE 5-2. CASSETTE CONSOLE CONTROL PANEL CONTROLS

Component Name	Function
TIMER	A 0 to 2000 hour elasped time indicator that measures the length of time AC POWER has been applied to the console.
AC POWER circuit breaker	
ON	Applies both live and neutral lines (120-VAC operation) or two live lines (220-VAC operation) to AC operated components.
OFF	In this position both AC lines are broken. The circuit breaker can be manually operated to this position or it will go to this position automatically if the line current rises above 3 amps.
STATUS NOT RUNNING indicator	Indicates that the BC is not executing a program.
JPM indicator	Indicates that a jump condition has been met in the BC.
RNI indicator	Indicates that the BC is operating in the read next instruction mode.
RADR indicator	Indicates that the BC is operating in the read address mode.
ROP indicator	Indicates that the BC is operating in the read operand mode.
STO indicator	Indicates that the BC is operating in the store operand mode.
PARITY ERROR indicator	Indicates that a memory parity error has been detected in the BC. The condition is cleared by pressing the pushbutton.
CHANNEL CLEAR indicator	Indicates that the clear channel signal is active. This condition is generated by, 1) being in autoload, 2) pressing the STOP or MC pushbuttons (provided the TEST MODE switch is off), and 3) exiting from the test mode. Pressing the pushbutton or setting the OPERATION switch to its NORM position will reenable the channel and extinguish the indicator.
DISPLAY readout	A four-section hexadecimal readout that displays the information selected by the DISPLAY SELECT pushbutton-indicators.
DISPLAY SELECT	
A	When this pushbutton indicator is pressed the output of the adder will be shown in the DISPLAY readout. The indicator lights to show that A has been selected.

TABLE 5-2. CASSETTE CONSOLE CONTROL PANEL CONTROLS (Cont'd)

Component Name	Function
DISPLAY SELECT	
P	When this pushbutton indicator is pressed, the contents of the P register in the BC will be shown in the DISPLAY readout and a new address can be entered. The indicator lights to show that P has been selected.
BREAKPOINT indicator	When this pushbutton is pressed, the breakpoint address is shown in the DISPLAY readout and a new address can be entered. The indicator lights to show that BREAKPOINT has been selected.
ENTRY SELECT switch	
ENTRY	In this position, the contents of the entry register are displayed in the ENTRY readout.
P	In this position, the current address in the P register of the BC is displayed in the ENTRY readout.
BKPT switch	
BTH	Enables the BC to make comparisions between operand and instruction addresses and the address stored in the breakpoint address register. There are two positions on this selector that performs this function.
INS	Enables the BC to make comparisons between instruction addresses and the address stored in the breakpoint address register.
OFF	The BC is not affected by the contents of the breakpoint address register.
OPR	Enables the BC to make comparisons between operand addresses and the address stored in the breakpoint address register.
ADD switch	Locked out when the BC is in the run mode. These four positions correspond to the true and complement values of the T translation and the X register. In any of these positions the corresponding value is gated to the BC's adder, where this value is treated as an addend. In the OFF position the value used by the BC's adder as an addend is not affected by the console.
$\overline{X}$	
X	
OFF	
T	
$\overline{T}$	

TABLE 5-2. CASSETTE CONSOLE CONTROL PANEL CONTROLS (Cont'd)

Component Name	Function
AUG switch	Locked out when the BC is in the run mode.
S A OFF B1 B2	These four positions correspond to registers in the BC. In any of these positions the contents of the selected register are gated to the BC's adder where the value is treated as an augend. In the OFF position the value used by the BC's adder as an augend is not affected by the console.
ENTRY readout	A four-section hexadecimal LED readout that displays manual entries from the keyboard and the contents of the P register.
MANUAL ENTRY keyboard	A 16 key hexadecimal keyboard with enter and clear keys, used to manually enter instructions, data, and addresses into the BC via the console.
TIMING CHAIN MARGIN #1 switch	
FAST	Enables the BC to subtract 1.5 nanoseconds from each step of its timing chain #1.
NORMAL	Enables the BC's timing chain #1 to operate at the normal rate of 20 nanoseconds per step.
SLOW	Enables the BC to add 1.5 nanoseconds to each step of its timing chain #1.
TIMING CHAIN MARGIN #2 switch	
FAST	Enables the BC to subtract 1.5 nanoseconds from each step of its timing chain #2.
NORMAL	Enables the BC's timing chain #2 to operate at its normal rate of 20 nanoseconds per step.
SLOW	Enables the BC to add 1.5 nanoseconds to each step of its timing chain #2.
PE STOP switch	
ON	Enables the BC to halt upon detection of a memory parity error.
OFF	No effect on BC operation.

TABLE 5-2. CASSETTE CONSOLE CONTROL PANEL CONTROLS (Cont'd)

Component Name	Function
SEL STOP switch	
ON	Enables the BC to halt upon detection of any instruction word read from its memory which contains 00XX.
OFF	No effect or BC operation.
CONSOLE CONTROL switch	
LAMP TEST momentary	A test of all the console's indicators is performed. The DISPLAY and ENTRY readouts are not checked. The switch in this position has no effect on the operation of either the console or the BC.
OFF	When released, the switch returns to this center inactive position.
MC momentary	The consoles internal logic is returned to initial condition without affecting the BC.
MC switch	Pressing this pushbutton causes a master clear command to be transmitted to the BC.
MODE switch	
CYCLE STEP	Enables the BC to execute an instruction one storage cycle at a time.
NORM	Enables the BC to execute a program of instructions without interruption.
INSTRUCTION STEP	Enables the BC to execute a program one instruction at a time.
STOP switch	Pressing this pushbutton causes a stop command to be transmitted to the BC.
OPERATION switch	
SWEEP	Enables the storage of instruction and data read from tape or entered via the keyboard into the BC's core memory.
NORM	No effect on the BC's operation.
ENTER	Enables information in the BC's core memory to be read from sequential locations shown in the DISPLAY readout and written on tape. Instructions that may be contained in these locations will not be executed.

TABLE 5-2. CASSETTE CONSOLE CONTROL PANEL CONTROLS (Cont'd)

Component Name	Function
GO switch	Pressing this pushbutton causes a go command to be transmitted to the BC.
TEST MODE TIMING control	
NOT RUNNING	When the control is in this detent position repetition of the test mode cycles is controlled by the return of the RUN-ID signal as an active Hi from the BC following each cycle.
MIN/MAX	Out of the NOT RUNNING position, the test mode cycle repetition rate is infinitely variable between the nominal limits of 20 KHz to 200 Hz. (50 nsec to 5 ms)
TEST MODE switch	
ON	Causes a sequence of STOP, MC, and GO commands, in that order, to be sent to the BC. The sequence is repeated at a rate determined by either the setting of the TEST MODE TIMING control or the rate at which the BC returns the RUN-ID signal. GO, MC, and STOP pushbuttons are logically locked out.
OFF	Test mode sequences are disabled after the next MC is generated. When switched from ON to OFF, the channel clear line is set to active LO.
GND test point	This test point is a common logic ground and is connected to chassis ground.
SIGNAL MONITOR test point	The signal selected by the SIGNAL MONITOR switch is routed to this test point. The signal is neither attenuated or delayed.
SIGNAL MONITOR indicator	Indicates that the signal selected by the SIGNAL MONITOR switch is active and its time, if less than 20 milliseconds, has been extended to a minimum of 20 milliseconds.
SIGNAL MONITOR selector	Routes one of 16 internal console signals to the SIGNAL MONITOR test point and SIGNAL MONITOR indicator.
	The signals available are as follows:
0	GO
1	STOP
2	MASTER CLEAR
3	NOT RUNNING
4	ENABLE PLAYBACK
5	$\overline{\text{WRITE ENABLE}}$
6	$\overline{\text{WRITE DATA}}$
7	$\overline{\text{READ DATA}}$
8	P = BREAKPOINT

TABLE 5-2. CASSETTE CONSOLE CONTROL PANEL CONTROLS (Cont'd)

Component Name	Function
SIGNAL MONITOR selector	
9	TAPE WORD =
10	GAP # <
11	GAP # >
12	GAP # =
13	BCC ERROR
14	TAPE PARITY ERROR
15	SEARCH ERROR
AUTOLOAD switch/ indicator	Pressing this pushbutton initiates the autoload data entry mode and sets the channel clear signal to active LO.
BOT switch/indicator	Pressing this pushbutton causes the tape cassette unit to search forward or reverse to the beginning of tape point. When this point is reached, tape motion stops and the indicator lights.
EOT indicator	Indicates that the end-of-tape slot has been detected.
ERROR indicator	This indicator will light when either a parity error, a BCC error, or a search error has been detected while reading or writing a tape. Setting the SIGNAL MONITOR switch to positions 13, 14, or 15 observing the SIGNAL MONITOR indicator will show which type of error has been detected.
TAPE STATUS CONTROL switch	
WRITE momentary	Enables programs and data stored in the BC's memory to be written on the cassette tape. The OPERATION switch must be in the SWEEP position.
OFF	This is the normal position for this switch and it has no affect on the operation of the console or the BC.
UNLOAD momentary	Causes the cassette tape to be fully re-wound. BOT is ignored.
EJECT switch/indicator (unmarked on panel)	Pressing this pushbutton will raise the front edge of the magnetic tape cassette so that it can be manually removed.

STATION NORMAL CHANNEL BIT ASSIGNMENTS A

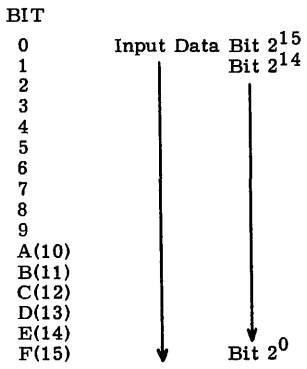
This appendix identifies the normal channel bit assignments for each STAR peripheral station. Bit assignments for channels 0 through 4 are common to all stations except for the maintenance station. Bit assignments for normal channels 5, 6, and 7 and for all channels of the maintenance station follow the listings for channel 4. In those instances where functions are not given for particular bits, the bits are unassigned. Use the index below to find the beginning page for the desired station.

Refer to appendix B for a listing of bit assignments arranged by interface.

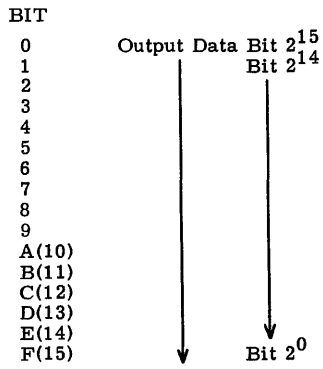
COMMON NORMAL CHANNEL BIT ASSIGNMENTS

The following bit assignments for channels 0 through 4 are common to all stations except for the maintenance control units.

INPUT CHANNEL 0
Microdrum



OUTPUT CHANNEL 0
Microdrum



COMMON NORMAL CHANNEL BIT ASSIGNMENTS

INPUT CHANNEL 1
Keyboard

BIT	
0	Control Panel Lockout
1	Character Code 2^6
2	↓
3	↓
4	↓
5	↓
6	↓
7	2^0
8	Alarm Disable (alert)
9	Repeat Key
A(10)	Keyboard Lockout Switch
B(11)	Data Strobe
C(12)	Mode 1
D(13)	Mode 2
E(14)	Mode 3
F(15)	Mode 4

OUTPUT CHANNEL 1
Microdrum

BIT	
0	Data Mode
1	Write Request
2	Read Request
3	Sector Address 2^4
4	↓ 2^3
5	↓ 2^2
6	↓ 2^1
7	↓ 2^0
8	Character Mark
9	Word Address 2^6
A(10)	↓ 2^5
B(11)	↓ 2^4
C(12)	↓ 2^3
D(13)	↓ 2^2
E(14)	↓ 2^1
F(15)	↓ 2^0

INPUT CHANNEL 2
Coupler A/Microdrum

BIT	
0	Coupler Time-out Status
1	Parity Error Fault Status
2	Read Parity Error Status
3	Illegal Status
4	Suspend Status
5	Invalid Status
6	Fault Status
7	Channel Busy Status
8	Coupler Ready Status
9	
A(10)	
B(11)	Sector Address 2^4
C(12)	↓ 2^3
D(13)	↓ 2^2
E(14)	↓ 2^1
F(15)	↓ 2^0

OUTPUT CHANNEL 2
Coupler A

BIT	
0	Initiate Functions
1	Function 2^2 B0
2	Function 2^1 B1
3	Function 2^0 B2
4	Interrupt
5	Clear Fault
6	Disable Parity Check
7	
8	
9	Coupler Select 2^1
A(10)	Coupler Select 2^0
B(11)	
C(12)	
D(13)	
E(14)	
F(15)	

COMMON NORMAL CHANNEL BIT ASSIGNMENTS

INPUT CHANNEL 3
Scanner High Priority

BIT	
0	Microdrum Busy
1	Real-Time Strobe
2	Flag Coupler A
3	Flag Coupler B
4	
5	
6	
7	
8	
9	
A(10)	
B(11)	
C(12)	
D(13)	
E(14)	
F(15)	

OUTPUT CHANNEL 3
Microdrum/Keyboard

BIT	
0	Checksum Mode
1	
2	
3	
4	
5	Deadman Strobe
6	Alert Indicator
7	Buzzer
8	Lockout Keyboard
9	Compare Lockout (Drum)
A(10)	Head Address 2^5
B(11)	2^4
C(12)	2^3
D(13)	2^2
E(14)	2^1
F(15)	2^0

INPUT CHANNEL 4
Coupler B

BIT	
0	Coupler Time-out Status
1	Parity Error Fault Status
2	Read Parity Error Status
3	Illegal Status
4	Suspend Status
5	Invalid Status
6	Fault Status
7	Channel Busy Status
8	Coupler Ready Status
9	Coupler Software Flag 1
A(10)	Coupler Software Flag 2
B(11)	
C(12)	
D(13)	
E(14)	
F(15)	

OUTPUT CHANNEL 4
Coupler B

BIT	
0	Initiate Functions
1	Function 2^2
2	Function 2^1
3	Function 2^0
4	Interrupt
5	Clear Fault
6	Disable Parity Check
7	
8	
9	Coupler Software Flag 1
A(10)	Coupler Software Flag 2
B(11)	
C(12)	
D(13)	
E(14)	
F(15)	

PAGING (DRUM) STATION NORMAL CHANNEL BIT ASSIGNMENTS

PAGING (DRUM) STATION
INPUT CHANNEL 5

Bit 0		
SAC-1	SBU Parity Error	
7000-0	SBU Parity Error	
7000-1	SBU Parity Error	
Index Compare	SBU Parity Error	
Bit 1		
SAC-1	$\overline{\text{Bank Counter}}\ 2^2$	
7000-0	$\overline{\text{Bank Counter}}\ 2^2$	
7000-1	$\overline{\text{Bank Counter}}\ 2^2$	
Index Compare	$\overline{\text{Bank Counter}}\ 2^2$	
Bit 2		
SAC-1	$\overline{\text{Bank Counter}}\ 2^1$	
7000-0	$\overline{\text{Bank Counter}}\ 2^1$	
7000-1	$\overline{\text{Bank Counter}}\ 2^1$	
Index Compare	$\overline{\text{Bank Counter}}\ 2^1$	
Bit 3		
SAC-1	$\overline{\text{Bank Counter}}\ 2^0$	
7000-0	$\overline{\text{Bank Counter}}\ 2^0$	
7000-1	$\overline{\text{Bank Counter}}\ 2^0$	
Index Compare	$\overline{\text{Bank Counter}}\ 2^0$	
Bit 4		
SAC-1	Transmission Parity Error	
7000-0	Drum Sector Count 2^6	
7000-1	Drum Sector Count 2^6	
Bit 5		
SAC-1	SAC Parity Error	
7000-0	Drum Sector Count 2^5	
7000-1	Drum Sector Count 2^5	
Bit 6		
SAC-1	Illegal	
7000-0	Drum Sector Count 2^4	
7000-1	Drum Sector Count 2^4	
Bit 7		
SAC-1	Busy	
7000-0	Drum Sector Count 2^3	
7000-1	Drum Sector Count 2^3	
Bit 8		
SAC-1	Term. Address Compare	
7000-0	Drum Sector Count 2^2	
7000-1	Drum Sector Count 2^2	
Bit 9		
SAC-1	First Rank Full	
7000-0	Drum Sector Count 2^1	
7000-1	Drum Sector Count 2^1	
Bit A		
SAC-1	Last Rank Full	
7000-0	Drum Sector Count 2^0	
7000-1	Drum Sector Count 2^0	
Bit B		
SAC-1	Control Function 2^1	
7000-0	Drum Sector Stable	
7000-1	Drum Sector Stable	
Bit C		
SAC-1	Control Function 2^0	
7000-0	Drum EOT Bit 2^1	
7000-1	Drum EOT Bit 2^1	
Bit D		
SAC-1	Control Function Strobe	
7000-0	Drum Abnormal	
7000-1	Drum Abnormal	
Bit E		
SAC-1		
7000-0	Drum Parity Error	
7000-1	Drum Parity Error	
Bit F		
SAC-1		
7000-0	Drum Not Ready	
7000-1	Drum Not Ready	

PAGING (DRUM) STATION
OUTPUT CHANNEL 5

Bit 0		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{15}	
7000-0	EDS/SDS†	
7000-1	EDS/SDS†	
Ch. 5 Fanin		
Bit 1		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{14}††	
7000-0	SBU SA/TA/Function Bit 2^{14}	
7000-1	SBU SA/TA/Function Bit 2^{14}	
Ch. 5 Fanin		
Bit 2		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{13}	
7000-0	SBU SA/TA/Function Bit 2^{13}	
7000-1	SBU SA/TA/Function Bit 2^{13}	
Ch. 5 Fanin		
Bit 3		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{12}	
7000-0	SBU SA/TA/Function Bit 2^{12}	
7000-1	SBU SA/TA/Function Bit 2^{12}	
Ch. 5 Fanin		
Bit 4		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{11}	
7000-0	SBU SA/TA/Function Bit 2^{11}	
7000-1	SBU SA/TA/Function Bit 2^{11}	
Ch. 5 Fanin		
Bit 5		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{10}	
7000-0	SBU SA/TA/Function Bit 2^{10}	
7000-1	SBU SA/TA/Function Bit 2^{10}	
Ch. 5 Fanin		
Bit 6		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{9}	
7000-0	SBU SA/TA/Function Bit 2^{9}	
7000-1	SBU SA/TA/Function Bit 2^{9}	
Ch. 5 Fanin		
Bit 7		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{8}	
7000-0	SBU SA/TA/Function Bit 2^{8}	
7000-1	SBU SA/TA/Function Bit 2^{8}	
Ch. 5 Fanin		
Bit 8		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{7}	
7000-0	SBU SA/TA/Function Bit 2^{7}	
7000-1	SBU SA/TA/Function Bit 2^{7}	
Ch. 5 Fanin		
Bit 9		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{6}	
7000-0	SBU SA/TA/Function Bit 2^{6}	
7000-1	SBU SA/TA/Function Bit 2^{6}	
Ch. 5 Fanin		
Bit A		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{5}	
7000-0	SBU SA/TA/Function Bit 2^{5}	
7000-1	SBU SA/TA/Function Bit 2^{5}	
Ch. 5 Fanin		
Bit B		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{4}	
7000-0	SBU SA/TA/Function Bit 2^{4}	
7000-1	SBU SA/TA/Function Bit 2^{4}	
Ch. 5 Fanin		
Bit C		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{3}	
7000-0	SBU SA/TA/Function Bit 2^{3}	
7000-1	SBU SA/TA/Function Bit 2^{3}	
Ch. 5 Fanin		
Bit D		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{2}	
7000-0	SBU SA/TA/Function Bit 2^{2}	
7000-1	SBU SA/TA/Function Bit 2^{2}	
Ch. 5 Fanin		
Bit E		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{1}	
7000-0	SBU SA/TA/Function Bit 2^{1}	
7000-1	SBU SA/TA/Function Bit 2^{1}	
Index Compare	SCU Key Identifier	
Bit F		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{0}	
7000-0	SBU SA/TA/Function Bit 2^{0}	
7000-1	SBU SA/TA/Function Bit 2^{0}	
Index Compare	SCU Request Identifier	

†EDS/SDS = End data sequence/start data sequence
††TA = Terminating address; SA = starting address

PAGING (DRUM) STATION
INPUT CHANNEL 6

Bit 0	
Scanner	7000-1 Drum Sector Count Bit 2^0
Ch. 5 Fanin	Bit 0
Ch. 7 Fanin	Bit 0
Bit 1	
Scanner	7000-1 Drum EOT Bit 2^0†
Ch. 5 Fanin	Bit 1
Ch. 7 Fanin	Bit 1
Bit 2	
Scanner	7000-0 Drum Sector Count Bit 2^0
Ch. 5 Fanin	Bit 2
Ch. 7 Fanin	Bit 2
Bit 3	
Scanner	7000-0 Drum EOT Bit 2^0
Ch. 5 Fanin	Bit 3
Ch. 7 Fanin	Bit 3
Bit 4	
Scanner	Compare Not Error
Ch. 5 Fanin	Bit 4
Ch. 7 Fanin	Bit 4
Bit 5	
Scanner	Compare REQ-1 Hit
SCU Int	$\overline{\text{Bank Counter } 2^2}$
Ch. 5 Fanin	Bit 5
Ch. 7 Fanin	Bit 5
Bit 6	
Scanner	Compare End of Table
SCU Int	$\overline{\text{Bank Counter } 2^1}$
Ch. 5 Fanin	Bit 6
Ch. 7 Fanin	Bit 6
Bit 7	
Scanner	Compare REQ-0 Hit
SCU Int	$\overline{\text{Bank Counter } 2^0}$
Ch. 5 Fanin	Bit 7
Ch. 7 Fanin	Bit 7
Bit 8	
Scanner	
Ch. 6 Fanin	Bit 8
Ch. 5 Fanin	Bit 8
Ch. 7 Fanin	Bit 8
Bit 9	
Scanner	7000-1 Drum Error
Ch. 6 Fanin	Bit 9
Ch. 5 Fanin	Bit 9
Ch. 7 Fanin	Bit 9
Bit A	
Scanner	7000-0 Drum Error
Ch. 6 Fanin	Bit A
Ch. 5 Fanin	Bit A
Ch. 7 Fanin	Bit A
Bit B	
Scanner	SAC-1 Not Busy
Ch. 6 Fanin	Bit B
Ch. 5 Fanin	Bit B
Ch. 7 Fanin	Bit B
Bit C	
Scanner	SAC-1 Flag
Ch. 6 Fanin	Bit C
Ch. 5 Fanin	Bit C
Ch. 7 Fanin	Bit C
Bit D	
Scanner	Compare Not Busy
Ch. 6 Fanin	Bit D
Ch. 5 Fanin	Bit D
Ch. 7 Fanin	Bit D
Bit E	
Scanner	
Ch. 6 Fanin	Bit E
Ch. 5 Fanin	Bit E
Ch. 7 Fanin	Bit E
Bit F	
Scanner	
Ch. 6 Fanin	Bit F
Ch. 5 Fanin	Bit F
Ch. 7 Fanin	Bit F

†$\overline{\text{EOT}}$ = End of Transfer

PAGING (DRUM) STATION
OUTPUT CHANNEL 6

Bit 0

SAC-1	SBU Starting Address
7000-0	SBU Starting Address
7000-1	SBU Starting Address
Index Compare	SBU Starting Address

Bit 1

SAC-1	SBU Terminating Address
7000-0	SBU Terminating Address
7000-1	SBU Terminating Address
Index Compare	SBU Terminating Address

Bit 2

SAC-1	Function
7000-0	Function
7000-1	Function
Index Compare	Load Request Register

Bit 3

SAC-1	Program Clear
7000-0	Program Clear
7000-1	Program Clear
Index Compare	Program Clear

Bit 4

SAC-1	Request
7000-0	
7000-1	
Index Compare	Start Search

Bit 5

SAC-1	Accept
7000-0	
7000-1	
Index Compare	Write Buffers

Bit 6

SAC-1	Select Status
7000-0	
7000-1	
Index Compare	Search Backward

Bit 7

SAC-1	Set Interrupt
7000-0	
7000-1	
Index Compare	One Request

Bit 8

Index Compare	Norm Chan 7 Select
	Norm Chan 6, Bit 8 Fanout

Bit 9

Index Compare	Norm Chan 5 Select
	Norm Chan 6, Bit 9 Fanout

Bit A

Index Compare	Norm Chan 6 Select
	Norm Chan 6, Bit A Fanout

Bit B

Scanner Select
Norm Chan 6, Bit B Fanout

Bit C

Connect Code 2^3
Norm Chan 6, Bit C Fanout

Bit D

Connect Code 2^2
Norm Chan 6, Bit D Fanout

Bit E

Connect Code 2^1
Norm Chan 6, Bit E Fanout

Bit F

Connect Code 2^0
Norm Chan 6, Bit F Fanout

PAGING (DRUM) STATION
INPUT CHANNEL 7

Bit 0		
SAC-1	SCU Data/RA Bit 2^{15}†	
7000-0		
7000-1		
Compare		
Bit 1		
SAC-1	SCU Data/RA Bit 2^{14}	
7000-0		
7000-1		
Compare		
Bit 2		
SAC-1	SCU Data/RA Bit 2^{13}	
7000-0		
7000-1		
Compare		
Bit 3		
SAC-1	SCU Data/RA Bit 2^{12}	
7000-0		
7000-1		
Compare		
Bit 4		
SAC-1	SCU Data/RA Bit 2^{11}	
7000-0		
7000-1		
Compare		
Bit 5		
SAC-1	SCU Data/RA Bit 2^{10}†	
7000-0		
7000-1		
Compare		
Bit 6		
SAC-1	SCU Data/RA Bit 2^{9}	
7000-0		
7000-1		
Compare	Delete Key Count Bit 2^{9}	
Bit 7		
SAC-1	SCU Data/RA Bit 2^{8}	
7000-0		
7000-1		
Compare	Delete Key Count Bit 2^{8}	
Bit 8		
SAC-1	SCU Data/RA Bit 2^{7}	
7000-0		
7000-1		
Compare	Delete Key Count Bit 2^{7}	
Bit 9		
SAC-1	SCU Data/RA Bit 2^{6}	
7000-0		
7000-1		
Compare	Delete Key Count Bit 2^{6}	
Bit A		
SAC-1	SCU Data/RA Bit 2^{5}	
7000-0		
7000-1		
Compare	Delete Key Count Bit 2^{5}	
Bit B		
SAC-1	SCU Data/RA Bit 2^{4}†	
7000-0		
7000-1		
Compare	Delete Key Count Bit 2^{4}	
Bit C		
SAC-1	SCU Data/RA Bit 2^{3}	
7000-0		
7000-1		
Compare	Delete Key Count Bit 2^{3}	
Bit D		
SAC-1	SCU Data/RA Bit 2^{2}	
7000-0		
7000-1		
Compare	Delete Key Count Bit 2^{2}	
Bit E		
SAC-1	SCU Data/RA Bit 2^{1}	
7000-0		
7000-1		
Compare	Delete Key Count Bit 2^{1}	
Bit F		
SAC-1	SCU Data/RA Bit 2^{0}	
7000-0		
7000-1		
Compare	Delete Key Count Bit 2^{0}	

†RA = Return address

PAGING (DRUM) STATION
OUTPUT CHANNEL 7

Bit 0

SAC-1	SBU SA Bit 2^{15}
Index Compare	Request Bit 2^{15}
Ch. 7 Fanout	Bit 2^{15}

Bit 1

SAC-1	SBU SA Bit 2^{14}
Index Compare	SBU SA/TA/Request Bit 2^{14}
Ch. 7 Fanout	Bit 2^{14}

Bit 2

SAC-1	SBU SA Bit 2^{13}
Index Compare	SBU SA/TA/Request Bit 2^{13}
Ch. 7 Fanout	Bit 2^{13}

Bit 3

SAC-1	SBU SA Bit 2^{12}
Index Compare	SBU SA/TA/Request Bit 2^{12}
Ch. 7 Fanout	Bit 2^{12}

Bit 4

SAC-1	SBU SA Bit 2^{11}
Index Compare	SBU SA/TA/Request Bit 2^{11}
Ch. 7 Fanout	Bit 2^{11}

Bit 5

SAC-1	SBU SA Bit 2^{10}
Index Compare	SBU SA/TA/Request Bit 2^{10}
Ch. 7 Fanout	Bit 2^{10}

Bit 6

SAC-1	SBU SA Bit 2^{9}
Index Compare	SBU SA/TA/Request Bit 2^{9}
Ch. 7 Fanout	Bit 2^{9}

Bit 7

SAC-1	SBU SA/Clear Parity Error Bit 2^{8}
Index Compare	SBU SA/TA/Request Bit 2^{8}
Ch. 7 Fanout	Bit 2^{8}

Bit 8

SAC-1	SBU SA/Clear Illegal Bit 2^{7}
Index Compare	SBU SA/TA/Request Bit 2^{7}
Ch. 7 Fanout	Bit 2^{7}

Bit 9

SAC-1	SBU SA/Clear Cont Func Bit 2^{6}
Index Compare	SBU SA/TA/Request Bit 2^{6}
Ch. 7 Fanout	Bit 2^{6}

Bit A

SAC-1	SBU SA/M.C. Bit 2^{5}
Index Compare	SBU SA/TA/Request Bit 2^{5}
Ch. 7 Fanout	Bit 2^{5}

Bit B

SAC-1	SBU SA Bit 2^{4}
Index Compare	SBU SA/TA/Request Bit 2^{4}
Ch. 7 Fanout	Bit 2^{4}

Bit C

SAC-1	SBU Select SAC + $\overline{\text{SCU}}$/SA Bit 2^{3}
Index Compare	SBU SA/TA/Request Bit 2^{3}
Ch. 7 Fanout	Bit 2^{3}

Bit D

SAC-1	SBU SA/Func Code Bit 2^{2}
Index Compare	Request Bit 2^{2}
Ch. 7 Fanout	Bit 2^{2}

Bit E

SAC-1	SBU SA/Func Code Bit 2^{1}
Index Compare	Request Bit 2^{1}
Ch. 7 Fanout	Bit 2^{1}

Bit F

SAC-1	SBU SA/Func Code Bit 2^{0}
Index Compare	Request Bit 2^{0}
Ch. 7 Fanout	Bit 2^{0}

DISK STATION NORMAL CHANNEL BIT ASSIGNMENTS

DISK STATION
INPUT CHANNEL 5

Bit 0		Bit 6		Bit B	
SAC-1	SBU Parity Error	SAC-1	Illegal	SAC-1	Control Function 2^1
SAC-2	SBU Parity Error	SAC-2	Illegal	SAC-2	Control Function 2^1
7000-0	SBU Parity Error	7000-0	Disk Sector Count 2^3/Seek Error	7000-0	Head Error
7000-1	SBU Parity Error	7000-1	Disk Sector Count 2^3/ Seek Error	7000-1	Head Error
Bit 1		**Bit 7**		**Bit C**	
SAC-1	$\overline{\text{Bank Counter}}$ 2^2	SAC-1	Busy	SAC-1	Control Function 2^0
SAC-2	$\overline{\text{Bank Counter}}$ 2^2	SAC-2	Busy	SAC-2	Control Function 2^0
7000-0	$\overline{\text{Bank Counter}}$ 2^2	7000-0	Disk Sector Count 2^2/Clock Error	7000-0	Disk Not On Cylinder/Write Error
7000-1	$\overline{\text{Bank Counter}}$ 2^2	7000-1	Disk Sector Count 2^2/Clock Error	7000-1	Disk Not On Cylinder/Write Error
Bit 2		**Bit 8**		**Bit D**	
SAC-1	$\overline{\text{Bank Counter}}$ 2^1	SAC-1	Term. Address Compare	SAC-1	Control Function Strobe
SAC-2	$\overline{\text{Bank Counter}}$ 2^1	SAC-2	Term. Address Compare	SAC-2	Control Function Strobe
7000-0	$\overline{\text{Bank Counter}}$ 2^1	7000-0	Disk Sector Count 2^1	7000-0	Disk Abnormal/Erase Error
7000-1	$\overline{\text{Bank Counter}}$ 2^1	7000-1	Disk Sector Count 2^1	7000-1	Disk Abnormal/Erase Error
Bit 3		**Bit 9**		**Bit E**	
SAC-1	$\overline{\text{Bank Counter}}$ 2^0	SAC-1	First Rank Full	SAC-1	
SAC-2	$\overline{\text{Bank Counter}}$ 2^0	SAC-2	First Rank Full	SAC-2	
7000-0	$\overline{\text{Bank Counter}}$ 2^0	7000-0	Disk Sector Count 2^0	7000-0	Disk Parity Error/Skew Error
7000-1	$\overline{\text{Bank Counter}}$ 2^0	7000-1	Disk Sector Count 2^0	7000-1	Disk Parity Error/Skew Error
Bit 4		**Bit A**		**Bit F**	
SAC-1	Transmission Parity Error	SAC-1	Last Rank Full	SAC-1	
SAC-2	Transmission Parity Error	SAC-2	Last Rank Full	SAC-2	
7000-0	Disk Sector Count 2^5/Voltage Error	7000-0	Logic Error	7000-0	Disk Not Ready/Offset
7000-1	Disk Sector Count 2^5/Voltage Error	7000-1	Logic Error	7000-1	Disk Not Ready/Offset

Bit 5	
SAC-1	SAC Parity Error
SAC-2	SAC Parity Error
7000-0	Disk Sector Count 2^4/Cylinder Select Error
7000-1	Disk Sector Count 2^4/Cylinder Select Error

DISK STATION
OUTPUT CHANNEL 5

Bit 0

SAC-1	SBU TA/SCU Addr/Data Bit 2^{15}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{15}
7000-0	EDS/SDS†
Ch. 5 Fanout	

Bit 1

SAC-1	SBU TA/SCU Addr/Data Bit 2^{14}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{14}
7000-0	SBU SA/TA/Function 2^{14}
Ch. 5 Fanout	

Bit 2

SAC-1	SBU TA/SCU Addr/Data Bit 2^{13}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{13}
7000-0	SBU SA/TA/Function Bit 2^{13}
Ch. 5 Fanout	

Bit 3

SAC-1	SBU TA/SCU Addr/Data Bit 2^{12}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{12}
7000-0	SBU SA/TA/Function Bit 2^{12}
Ch. 5 Fanout	

Bit 4

SAC-1	SBU TA/SCU Addr/Data Bit 2^{11}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{11}
7000-0	SBU SA/TA/Function Bit 2^{11}
Ch. 5 Fanout	

Bit 5

SAC-1	SBU TA/SCU Addr/Data Bit 2^{10}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{10}
7000-0	SBU SA/TA/Function Bit 2^{10}
Ch. 5 Fanout	

Bit 6

SAC-1	SBU TA/SCU Addr/Data Bit 2^{9}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{9}
7000-0	SBU SA/TA/Function Bit 2^{9}
Ch. 5 Fanout	

Bit 7

SAC-1	SBU TA/SCU Addr/Data Bit 2^{8}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{8}
7000-0	SBU SA/TA Function Bit 2^{8}
Ch. 5 Fanout	

Bit 8

SAC-1	SBU TA/SCU Addr/Data Bit 2^{7}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{7}
7000-0	SBU SA/TA/Function Bit 2^{7}
Ch. 5 Fanout	

Bit 9

SAC-1	SBU TA/SCU Addr/Data Bit 2^{6}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{6}
7000-0	SBU SA/TA/Function Bit 2^{6}
Ch. 5 Fanout	

Bit A

SAC-1	SBU TA/SCU Addr/Data Bit 2^{5}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{5}
7000-0	SBU SA/TA/Function Bit 2^{5}
Ch. 5 Fanout	

Bit B

SAC-1	SBU TA/SCU Addr/Data Bit 2^{4}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{4}
7000-0	SBU SA/TA/Function Bit 2^{4}
Ch. 5 Fanout	

Bit C

SAC-1	SBU TA/SCU Addr/Data Bit 2^{3}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{3}
7000-0	SBU SA/TA/Function Bit 2^{3}
Ch. 5 Fanout	

Bit D

SAC-1	SBU TA/SCU Addr/Data Bit 2^{2}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{2}
7000-0	SBU SA/TA/Function Bit 2^{2}
Ch. 5 Fanout	

Bit E

SAC-1	SBU TA/SCU Addr/Data Bit 2^{1}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{1}
7000-0	SBU SA/TA/Function Bit 2^{1}
Ch. 5 Fanout	

Bit F

SAC-1	SBU TA/SCU Addr/Data Bit 2^{0}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{0}
7000-0	SBU SA/TA/Function Bit 2^{0}
Ch. 5 Fanout	

†EDS/SDS = End data sequence/start data sequence

DISK STATION
INPUT CHANNEL 6

Bit 0

Scanner	7000-0 Disk On-Cylinder
Ch. 6 Fanin	
Ch. 5 Fanin	Bit 0
Ch. 7 Fanin	Bit 0

Bit 1

Scanner	7000-0 Disk Error
Ch. 6 Fanin	
Ch. 5 Fanin	Bit 1
Ch. 7 Fanin	Bit 1

Bit 2

Scanner	7000-0 Disk EOT Bit 2^1*
Ch. 6 Fanin	
Ch. 5 Fanin	Bit 2
Ch. 7 Fanin	Bit 2

Bit 3

Scanner	7000-0 Disk EOT 2^0
Ch. 6 Fanin	
Ch. 5 Fanin	Bit 3
Ch. 7 Fanin	Bit 3

Bit 4

Scanner	7000-1 Disk On-Cylinder
Ch. 6 Fanin	
Ch. 5 Fanin	Bit 4
Ch. 7 Fanin	Bit 4

Bit 5

Scanner	7000-1 Disk Error
SCU Int	$\overline{\text{Bank Counter } 2^2}$
Ch. 5 Fanin	Bit 5
Ch. 7 Fanin	Bit 5

Bit 6

Scanner	7000-1 Disk EOT 2^1
SCU Int	$\overline{\text{Bank Counter } 2^1}$
Ch. 5 Fanin	Bit 6
Ch. 7 Fanin	Bit 6

Bit 7

Scanner	7000-1 Disk EOT 2^0
SCU Int	$\overline{\text{Bank Counter } 2^0}$
Ch. 5 Fanin	Bit 7
Ch. 7 Fanin	Bit 7

Bit 8

Scanner	SAC-1 Not Busy
Ch. 6 Fanin	Bit 8
Ch. 5 Fanin	Bit 8
Ch. 7 Fanin	Bit 8

Bit 9

Scanner	SAC-2 Not Busy
Ch. 6 Fanin	Bit 9
Ch. 5 Fanin	Bit 9
Ch. 7 Fanin	Bit 9

Bit A

Scanner	SAC-1 Flag
Ch. 6 Fanin	Bit A
Ch. 5 Fanin	Bit A
Ch. 7 Fanin	Bit A

Bit B

Scanner	SAC-2 Flag
Ch. 6 Fanin	Bit B
Ch. 5 Fanin	Bit B
Ch. 7 Fanin	Bit B

Bit C

Scanner	
Ch. 6 Fanin	Bit C
Ch. 5 Fanin	Bit C
Ch. 7 Fanin	Bit C

Bit D

Scanner	
Ch. 6 Fanin	Bit D
Ch. 5 Fanin	Bit D
Ch. 7 Fanin	Bit D

Bit E

Scanner	
Ch. 6 Fanin	Bit E
Ch. 5 Fanin	Bit E
Ch. 7 Fanin	Bit E

Bit F

Scanner	
Ch. 6 Fanin	Bit F
Ch. 5 Fanin	Bit F
Ch. 7 Fanin	Bit F

*EOT = End of Transfer

DISK STATION
OUTPUT CHANNEL 6

Bit 0

SAC-1	SBU Starting Address
SAC-2	SBU Starting Address
7000-0	SBU Starting Address
7000-1	SBU Starting Address

Bit 1

SAC-1	SBU Term. Address
SAC-2	SBU Term. Address
7000-0	SBU Term. Address
7000-1	SBU Term. Address

Bit 2

SAC-1	Function
SAC-2	Function
7000-0	Function
7000-1	Function

Bit 3

SAC-1	Program Clear
SAC-2	Program Clear
7000-0	Program Clear
7000-1	Program Clear

Bit 4

SAC-1	Request
SAC-2	Request

Bit 5

SAC-1	Accept
SAC-2	Accept

Bit 6

SAC-1	Select Status
SAC-2	Select Status

Bit 7

SAC-1	Set Interrupt
SAC-2	Set Interrupt

Bit 8

†NC 7 Select
NC Bit 8 Fanout

Bit 9

NC 5 Select
Select Ch. 6
NC Bit 9 Fanout

Bit A

NC 6 Select
Select Ch. 7
NC Bit A Fanout

Bit B

Scanner Select
NC Bit B Fanout

Bit C

Connect Code 2^3
NC Bit C Fanout

Bit D

Connect Code 2^2
NC Bit D Fanout

Bit E

Connect Code 2^1
NC Bit E Fanout

Bit F

Connect Code 2^0
NC Bit F Fanout

†NC = Normal Channel

DISK STATION
INPUT CHANNEL 7

Bit 0		
	SAC-1	SCU Data/SBU RA 2^{15}†
	SAC-2	SCU Data/SBU RA 2^{15}
	7000-0	Return Function 2^{2}
	7000-1	Return Function 2^{2}
Bit 1		
	SAC-1	SCU Data/SBU RA 2^{14}
	SAC-2	SCU Data/SBU RA 2^{14}
Bit 2		
	SAC-1	SCU Data/SBU RA 2^{13}
	SAC-2	SCU Data/SBU RA 2^{13}
	7000-0	Return Function 2^{1}
	7000-1	Return Function 2^{1}
Bit 3		
	SAC-1	SCU Data/SBU RA 2^{12}
	SAC-2	SCU Data/SBU RA 2^{12}
	7000-0	Return Function 2^{0}
	7000-1	Return Function 2^{0}
Bit 4		
	SAC-1	SCU Data/SBU RA 2^{11}
	SAC-2	SCU Data/SBU RA 2^{11}
Bit 5		
	SAC-1	SCU Data/SBU RA 2^{10}
	SAC-2	SCU Data/SBU RA 2^{10}
Bit 6		
	SAC-1	SCU Data/SBU RA 2^{9}
	SAC-2	SCU Data/SBU RA 2^{9}
Bit 7		
	SAC-1	SCU Data/SBU RA 2^{8}
	SAC-2	SCU Data/SBU RA 2^{8}
Bit 8		
	SAC-1	SCU Data/SBU RA 2^{7}
	SAC-2	SCU Data/SBU RA 2^{7}
Bit 9		
	SAC-1	SCU Data/SBU RA 2^{6}
	SAC-2	SCU Data/SBU RA 2^{6}
Bit A		
	SAC-1	SCU Data/SBU RA 2^{5}
	SAC-2	SCU Data/SBU RA 2^{5}
Bit B		
	SAC-1	SCU Data/SBU RA 2^{4}
	SAC-2	SCU Data/SBU RA 2^{4}
Bit C		
	SAC-1	SCU Data/SBU RA 2^{3}
	SAC-2	SCU Data/SBU RA 2^{3}
Bit D		
	SAC-1	SCU Data/SBU RA 2^{2}
	SAC-2	SCU Data/SBU RA 2^{2}
Bit E		
	SAC-1	SCU Data/SBU RA 2^{1}
	SAC-2	SCU Data/SBU RA 2^{1}
Bit F		
	SAC-1	SCU Data/SBU RA 2^{0}
	SAC-2	SCU Data/SBU RA 2^{0}

†RA = Return address

DISK STATION
OUTPUT CHANNEL 7

Bit 0

SAC-1	SBU SA Bit 2^{15}
SAC-2	SBU SA Bit 2^{15}
7000-1	EDS/SDS†
Ch. 7 Fanout	Bit 2^{15}

Bit 1

SAC-1	SBU SA Bit 2^{14}
SAC-2	SBU SA Bit 2^{14}
7000-1	SBU SA/TA/Func Bit 2^{14}
Ch. 7 Fanout	Bit 2^{14}

Bit 2

SAC-1	SBU SA Bit 2^{13}
SAC-2	SBU SA Bit 2^{13}
7000-1	SBU SA/TA/Func Bit 2^{13}
Ch. 7 Fanout	Bit 2^{13}

Bit 3

SAC-1	SBU SA Bit 2^{12}
SAC-2	SBU SA Bit 2^{12}
7000-1	SBU SA/TA/Func Bit 2^{12}
Ch. 7 Fanout	Bit 2^{12}

Bit 4

SAC-1	SBU SA Bit 2^{11}
SAC-2	SBU SA Bit 2^{11}
7000-1	SBU SA/TA/Func Bit 2^{11}
Ch. 7 Fanout	Bit 2^{11}

Bit 5

SAC-1	SBU SA Bit 2^{10}
SAC-2	SBU SA Bit 2^{10}
7000-1	SBU SA/TA/Func Bit 2^{10}
Ch. 7 Fanout	Bit 2^{10}

Bit 6

SAC-1	SBU SA Bit 2^{9}
SAC-2	SBU SA Bit 2^{9}
7000-1	SBU SA/TA/Func Bit 2^{9}
Ch. 7 Fanout	Bit 2^{9}

Bit 7

SAC-1	Clear P.E./SBU SA Bit 2^{8}
SAC-2	Clear P.E./SBU SA Bit 2^{8}
7000-1	SBU SA/TA/Func Bit 2^{8}
Ch. 7 Fanout	Bit 2^{8}

Bit 8

SAC-1	Clear Illegal/SBU SA Bit 2^{7}
SAC-2	Clear Illegal/SBU SA Bit 2^{7}
7000-1	SBU SA/TA/Func Bit 2^{7}
Ch. 7 Fanout	Bit 2^{7}

Bit 9

SAC-1	Clear Cont Func/SBU SA Bit 2^{6}
SAC-2	Clear Cont Func/SBU SA Bit 2^{6}
7000-1	SBU SA/TA/Func Bit 2^{6}
Ch. 7 Fanout	Bit 2^{6}

Bit A

SAC-1	M.C./SBU SA Bit 2^{5}
SAC-2	SA/M.C./SBU SA Bit 2^{5}
7000-1	SBU SA/TA/Func Bit 2^{5}
Ch. 7 Fanout	Bit 2^{5}

Bit B

SAC-1	SBU SA Bit 2^{4}
SAC-2	SBU SA Bit 2^{4}
7000-1	SBU SA/TA/Func Bit 2^{4}
Ch. 7 Fanout	Bit 2^{4}

Bit C

SAC-1	Select SAC + $\overline{\text{SCU}}$/SBU SA Bit 2^{3}
SAC-2	Select SAC + $\overline{\text{SCU}}$/SBU SA Bit 2^{3}
7000-1	SBU SA/TA/Func Bit 2^{3}
Ch. 7 Fanout	Bit 2^{3}

Bit D

SAC-1	Func Code/SBU SA Bit 2^{2}
SAC-2	Func Code/SBU SA Bit 2^{2}
7000-1	SBU SA/TA/Func Bit 2^{2}
Ch. 7 Fanout	Bit 2^{2}

Bit E

SAC-1	Func Code/SBU SA Bit 2^{1}
SAC-2	Func Code/SBU SA Bit 2^{1}
7000-1	SBU SA/TA/Func Bit 2^{1}
Ch. 7 Fanout	Bit 2^{1}

Bit F

SAC-1	Func Code/SBU SA Bit 2^{0}
SAC-2	Func Code/SBU SA Bit 2^{0}
7000-1	SBU SA/TA/Func Bit 2^{0}
Ch. 7 Fanout	Bit 2^{0}

†EDS/SDS = End data sequence/start data sequence

844 SERVICE STATION NORMAL CHANNEL BIT ASSIGNMENTS

844 SERVICE STATION
INPUT CHANNEL 5

Bit 0		
SAC-1	SBU Parity Error	
SAC-2	SBU Parity Error	
STAR A	Memory P. E. SCU Coupler	
Bit 1		
SAC-1	$\overline{\text{Bank Counter}}\ 2^2$	
SAC-2	$\overline{\text{Bank Counter}}\ 2^2$	
STAR A	Sta. Int. X $\overline{\text{Bank Counter}}\ 2^2$ †	
Bit 2		
SAC-1	$\overline{\text{Bank Counter}}\ 2^1$	
SAC-2	$\overline{\text{Bank Counter}}\ 2^1$	
STAR A	Sta. Int. X $\overline{\text{Bank Counter}}\ 2^1$	
Bit 3		
SAC-1	$\overline{\text{Bank Counter}}\ 2^0$	
SAC-2	$\overline{\text{Bank Counter}}\ 2^0$	
STAR A	Sta. Int. X $\overline{\text{Bank Counter}}\ 2^0$	
Bit 4		
SAC-1	Transmission Parity Error	
SAC-2	Transmission Parity Error	
STAR A		
Bit 5		
SAC-1	SAC Parity Error	
SAC-2	SAC Parity Error	
STAR A		
Bit 6		
SAC-1	Illegal	
SAC-2	Illegal	
STAR A		
Bit 7		
SAC-1	Busy	
SAC-2	Busy	
STAR A		
Bit 8		
SAC-1	Term. Address Compare	
SAC-2	Term. Address Compare	
STAR A	Memory P. E. I/O Channel 1 ††	
Bit 9		
SAC-1	First Rank Full	
SAC-2	First Rank Full	
STAR A	Memory P. E. I/O Channel 2	
Bit A		
SAC-1	Last Rank Full	
SAC-2	Last Rank Full	
STAR A	Memory P. E. I/O Channel 3	
Bit B		
SAC-1	Control Function 2^1	
SAC-2	Control Function 2^1	
STAR A	Memory P. E. I/O Channel 4	
Bit C		
SAC-1	Control Function 2^0	
SAC-2	Control Function 2^0	
STAR A		
Bit D		
SAC-1	Control Function Strobe	
SAC-2	Control Function Strobe	
STAR A		
Bit E		
SAC-1	NU	
SAC-2	NU	
STAR A		
Bit F		
SAC-1	NU	
SAC-2	NU	
STAR A		

†STA INT = Station Interface
††P. E. = Parity Error

844 SERVICE STATION
OUTPUT CHANNEL 5

Bit 0		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{15}†	
SAC-2	SBU TA/SCU Addr/Data Bit 2^{15}	
Bit 1		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{14}	
SAC-2	SBU TA/SCU Addr/Data Bit 2^{14}	
Bit 2		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{13}	
SAC-2	SBU TA/SCU Addr/Data Bit 2^{13}	
Bit 3		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{12}	
SAC-2	SBU TA/SCU Addr/Data Bit 2^{12}	
Bit 4		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{11}	
SAC-2	SBU TA/SCU Addr/Data Bit 2^{11}	
Bit 5		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{10}	
SAC-2	SBU TA/SCU Addr/Data Bit 2^{10}	
Bit 6		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{9}	
SAC-2	SBU TA/SCU Addr/Data Bit 2^{9}	
Bit 7		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{8}	
SAC-2	SBU TA/SCU Addr/Data Bit 2^{8}	
Bit 8		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{7}	
SAC-2	SBU TA/SCU Addr/Data Bit 2^{7}	
Bit 9		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{6}	
SAC-2	SBU TA/SCU Addr/Data Bit 2^{6}	
Bit A		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{5}	
SAC-2	SBU TA/SCU Addr/Data Bit 2^{5}	
Bit B		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{4}	
SAC-2	SBU TA/SCU Addr/Data Bit 2^{4}	
Bit C		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{3}	
SAC-2	SBU TA/SCU Addr/Data Bit 2^{3}	
Bit D		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{2}	
SAC-2	SBU TA/SCU Addr/Data Bit 2^{2}	
Bit E		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{1}	
SAC-2	SBU TA/SCU Addr/Data Bit 2^{1}	
Bit F		
SAC-1	SBU TA/SCU Addr/Data Bit 2^{0}	
SAC-2	SBU TA/SCU Addr/Data Bit 2^{0}	

† TA = Terminating Address

844 SERVICE STATION
INPUT CHANNEL 6

Bit 0		
Scanner	Memory P.E. I/O Channel	
844	Unit Busy	
Bit 1		
Scanner	NU	
844	Unit 1 Selected	
Bit 2		
Scanner	SAC-1 Not Busy	
844	Unit 2 Selected	
Bit 3		
Scanner	SAC-2 Not Busy	
844	Unit 3 Selected	
Bit 4		
Scanner	SAC-1 Flag	
844		
Bit 5		
Scanner	SAC-2 Flag	
844		
Bit 6		
Scanner	NU	
844		
Bit 7		
Scanner	NU	
844		
Bit 8		
Scanner	I/O Interrupt Channel 1	
844		
Bit 9		
Scanner	I/O Interrupt Channel 2	
844		
Bit A		
Scanner	I/O Interrupt Channel 3	
844	Read Error	
Bit B		
Scanner	I/O Interrupt Channel 4	
844	Tag BCC Error*	
Bit C		
Scanner	NU	
844	Timing Chain	
Bit D		
Scanner	NU	
844	SBU Bank Counter 2^2	
Bit E		
Scanner	NU	
844	SBU Bank Counter 2^1	
Bit F		
Scanner	NU	
844	SBU Bank Counter 2^0	

*BCC = Block Check Code

844 SERVICE STATION
OUTPUT CHANNEL 6

Bit 0		
	SAC-1	SBU Starting Address
	SAC-2	SBU Starting Address
	STAR A	
Bit 1		
	SAC-1	SBU Terminating Address
	SAC-2	SBU Terminating Address
	STAR A	
Bit 2		
	SAC-1	Function
	SAC-2	Function
	STAR A	
Bit 3		
	SAC-1	Program Clear
	SAC-2	Program Clear
	STAR A	
Bit 4		
	SAC-1	Request
	SAC-2	Request
	STAR A	I/O Chan 4 Select
Bit 5		
	SAC-1	Accept
	SAC-2	Accept
	STAR A	I/O Chan 3 Select
Bit 6		
	SAC-1	Select Status
	SAC-2	Select Status
	STAR A	I/O Chan 2 Select
Bit 7		
	SAC-1	Set Interrupt
	SAC-2	Set Interrupt
	844	Load SBU Address
	STAR A	I/O Chan 1 Select
Bit 8		
	844	Release
	STAR A	Select SCU Coupler
Bit 9		
	844	Function
Bit A		
	844	Begin Transfer
Bit B		
		Scanner Select
Bit C		
	All Interfaces	Connect Code 2^3
Bit D		
	All Interfaces	Connect Code 2^2
Bit E		
	All Interfaces	Connect Code 2^1
Bit F		
	All Interfaces	Connect Code 2^0

844 SERVICE STATION
INPUT CHANNEL 7

Bit 0		
SAC-1	SCU Data/RA 2^{15}†	
SAC-2	SCU Data/RA 2^{15}	
844	Finished	
Bit 1		
SAC-1	SCU Data/RA 2^{14}	
SAC-2	SCU Data/RA 2^{14}	
844	Unit 1 On Sector/Seek Error	
Bit 2		
SAC-1	SCU Data/RA 2^{13}	
SAC-2	SCU Data/RA 2^{13}	
844	Unit 2 On Sector/Seek Error	
Bit 3		
SAC-1	SCU Data/RA 2^{12}	
SAC-2	SCU Data/RA 2^{12}	
844	Unit 3 On Sector/Seek Error	
Bit 4		
SAC-1	SCU Data/RA 2^{11}	
SAC-2	SCU Data/RA 2^{11}	
844	NU	
Bit 5		
SAC-1	SCU Data/RA 2^{10}	
SAC-2	SCU Data/RA 2^{10}	
844	NU	
Bit 6		
SAC-1	SCU Data/RA 2^{9}	
SAC-2	SCU Data/RA 2^{9}	
844	NU	
Bit 7		
SAC-1	SCU Data/RA 2^{8}	
SAC-2	SCU Data/RA 2^{8}	
844	NU	
Bit 8		
SAC-1	SCU Data/RA 2^{7}	
SAC-2	SCU Data/RA 2^{7}	
844	NU	
Bit 9		
SAC-1	SCU Data/RA 2^{6}	
SAC-2	SCU Data/RA 2^{6}	
844	NU	
Bit A		
SAC-1	SCU Data/RA 2^{5}	
SAC-2	SCU Data/RA 2^{5}	
844	Ready	
Bit B		
SAC-1	SCU Data/RA 2^{5}	
SAC-2	SCU Data/RA 2^{4}	
844	Transfer Error	
Bit C		
SAC-1	SCU Data/RA 2^{3}	
SAC-2	SCU Data/RA 2^{3}	
844	On Cylinder	
Bit D		
SAC-1	SCU Data/RA 2^{2}	
SAC-2	SCU Data/RA 2^{2}	
844	Seek Error	
Bit E		
SAC-1	SCU Data/RA 2^{1}	
SAC-2	SCU Data/RA 2^{1}	
844	Pack Unsafe	
Bit F		
SAC-1	SCU Data/RA 2^{0}	
SAC-2	SCU Data/RA 2^{0}	
844	Full	

†RA = Return address

844 SERVICE STATION
OUTPUT CHANNEL 7

Bit 0	
SAC-1	SBU SA Bit 2^{15}
SAC-2	SBU SA Bit 2^{15}
STAR A	
844	NU
Bit 1	
SAC-1	SBU SA Bit 2^{14}
SAC-2	SBU SA Bit 2^{14}
STAR A	
844	Select Unit 1
Bit 2	
SAC-1	SBU SA Bit 2^{13}
SAC-2	SBU SA Bit 2^{13}
STAR A	
844	Write Tag/Difference Select/Select Unit 2
Bit 3	
SAC-1	SBU SA Bit 2^{12}
SAC-2	SBU SA Bit 2^{12}
STAR A	
844	Write/Clear Fault/ Select Unit 3
Bit 4	
SAC-1	SBU SA Bit 2^{11}
SAC-2	SBU SA Bit 2^{11}
STAR A	
844	Full/Control Select
Bit 5	
SAC-1	SBU SA Bit 2^{10}
SAC-2	SBU SA Bit 2^{10}
STAR A	
844	Odd/Sector Select
Bit 6	
SAC-1	SBU SA Bit 2^{9}
SAC-2	SBU SA Bit 2^{9}
STAR A	
844	SBU Address/Cylinder Select
Bit 7	
SAC-1	Clear P.E./SBU SA Bit 2^{8}
SAC-2	Clear P.E./SBU SA Bit 2^{8}
STAR A	
844	SBU Address/DSU Address Control Bit 8
Bit 8	
SAC-1	Clear Illegal/SBU SA Bit 2^{7}
SAC-2	Clear Illegal/SBU SA Bit 2^{7}
STAR A	
844	SBU Address/DSU Address Control Bit 7
Bit 9	
SAC-1	Clear Cont. Function/SBU SA Bit 2^{6}
SAC-2	Clear Cont. Function/SBU SA Bit 2^{6}
STAR A	
844	SBU Address/DSU Address Control Bit 6
Bit A	
SAC-1	Master Clear/SBU SA Bit 2^{5}
SAC-2	Master Clear/SBU SA Bit 2^{5}
STAR A	
844	SBU Address/DSU Address Control Bit 5
Bit B	
SAC-1	SBU SA Bit 2^{4}
SAC-2	SBU SA Bit 2^{4}
STAR A	Clear Interrupt
844	SBU Address/DSU Address Control Bit 4
Bit C	
SAC-1	Select SAC + SCU/SBU SA Bit 2^{3}
SAC-2	Select SAC + SCU/SBU SA Bit 2^{3}
STAR A	Program Clear
844	SBU Address/DSU Address Control Bit 3
Bit D	
SAC-1	Function Code/SBU SA Bit 2^{2}
SAC-2	Function Code/SBU SA Bit 2^{2}
STAR A	Control Function 2^{1}
844	SBU Address/DSU Address Control Bit 2
Bit E	
SAC-1	Function Code/SBU SA Bit 2^{1}
SAC-2	Function Code/SBU SA Bit 2^{1}
STAR A	Control Function 2^{0}
844	SBU Address/DSU Address Control Bit 1
Bit F	
SAC-1	Function Code/SA Bit 2^{0}
SAC-2	Function Code/SA Bit 2^{0}
STAR A	Control Function Strobe
844	SBU Address/DSU Address Control Bit 0

844 STORAGE/MAGNETIC TAPE STATION NORMAL CHANNEL BIT ASSIGNMENTS

844 STORAGE/MAGNETIC TAPE STATION
INPUT CHANNEL 5

Bit 0		
SAC-1	SBU Parity Error	
SAC-2	SBU Parity Error	
3000-0		
3000-1		
Bit 1		
SAC-1	$\overline{\text{Bank Counter}}\ 2^2$	
SAC-2	$\overline{\text{Bank Counter}}\ 2^2$	
3000-0		
3000-1		
Bit 2		
SAC-1	$\overline{\text{Bank Counter}}\ 2^1$	
SAC-2	$\overline{\text{Bank Counter}}\ 2^1$	
3000-0	Interrupt Active	
3000-1	Interrupt Active	
Bit 3		
SAC-1	$\overline{\text{Bank Counter}}\ 2^0$	
SAC-2	$\overline{\text{Bank Counter}}\ 2^0$	
3000-0	SBU Parity Error	
3000-1	SBU Parity Error	
Bit 4		
SAC-1	Transmission Parity Error	
SAC-2	Transmission Parity Error	
3000-0	3000 Read Parity Error	
3000-1	3000 Read Parity Error	
Bit 5		
SAC-1	SAC Parity Error	
SAC-2	SAC Parity Error	
3000-0	Current Addr 2^{10}/Interrupt Line 3	
3000-1	Current Addr 2^{10}/Interrupt Line 3	
Bit 6		
SAC-1	Illegal	
SAC-2	Illegal	
3000-0	Current Addr 2^9/ Interrupt Line 2	
3000-1	Current Addr 2^9/ Interrupt Line 2	
Bit 7		
SAC-1	Busy	
SAC-2	Busy	
3000-0	Current Addr 2^8/ Interrupt Line 1	
3000-1	Current Addr 2^8/ Interrupt Line 1	
Bit 8		
SAC-1	Terminating Address Compare	
SAC-2	Terminating Address Compare	
3000-0	Current Address 2^7/ Interrupt Line 0	
3000-1	Current Address 2^7/ Interrupt Line 0	
Bit 9		
SAC-1	First Rank Full	
SAC-2	First Rank Full	
3000-0	Current Address 2^6/Byte Counter Rank 3	
3000-1	Current Address 2^6/Byte Counter Rank 3	
Bit A		
SAC-1	Last Rank Full	
SAC-2	Last Rank Full	
3000-0	Current Address 2^5/Byte Counter Rank 2	
3000-1	Current Address 2^5/Byte Counter Rank 2	
Bit B		
SAC-1	Control Function 2^1	
SAC-2	Control Function 2^1	
3000-0	Current Address 2^4/Byte Counter Rank 1	
3000-1	Current Address 2^4/Byte Counter Rank 1	
Bit C		
SAC-1	Control Function 2^0	
SAC-2	Control Function 2^0	
3000-0	Current Address 2^3/Byte Counter Rank 0	
3000-1	Current Address 2^3/Byte Counter Rank 0	
Bit D		
SAC-1	Control Function Strobe	
SAC-2	Control Function Strobe	
3000-0	Current Address 2^2/$\overline{\text{Phase Counter}}\ 2^2$	
3000-1	Current Address 2^2/$\overline{\text{Phase Counter}}\ 2^2$	
Bit E		
SAC-1	NU	
SAC-2	NU	
3000-0	Current Address 2^1/$\overline{\text{Phase Counter}}\ 2^1$	
3000-1	Current Address 2^1/$\overline{\text{Phase Counter}}\ 2^1$	
Bit F		
SAC-1	NU	
SAC-2	NU	
3000-0	Current Address 2^0/$\overline{\text{Phase Counter}}\ 2^0$	
3000-1	Current Address 2^0/$\overline{\text{Phase Counter}}\ 2^0$	

844 STORAGE/MAGNETIC TAPE STATION
OUTPUT CHANNEL 5

Bit 0

SAC-1	SBU TA/SCU Addr Bit 2^{15}
SAC-2	SBU TA/SCU Addr Bit 2^{15}
3000-0	Compare Bit/Binary/Coded
3000-1	Compare Bit/Binary/Coded

Bit 1

SAC-1	SBU TA/SCU Addr Bit 2^{14}
SAC-2	SBU TA/SCU Addr Bit 2^{14}
3000-0	SBU SA/TA/TBA/Read/Write †
3000-1	SBU SA/TA/TBA/Read/Write

Bit 2

SAC-1	SBU TA/SCU Addr Bit 2^{13}
SAC-2	SBU TA/SCU Addr Bit 2^{13}
3000-0	SBU SA/TA/TBA/Connect/Function
3000-1	SBU SA/TA/TBA/Connect/Function

Bit 3

SAC-1	SBU TA/SCU Addr Bit 2^{12}
SAC-2	SBU TA/SCU Addr Bit 2^{12}
3000-0	SBU SA/TA/TBA/Odd/Even
3000-1	SBU SA/TA/TBA/Odd/Even

Bit 4

SAC-1	SBU TA/SCU Addr Bit 2^{11}
SAC-2	SBU TA/SCU Addr Bit 2^{11}
3000-0	SBU SA/TA/TBA/Fctn Code/Conn Code/Disable Request
3000-1	SBU SA/TA/TBA/Fctn Code/Conn Code/Disable Request

Bit 5

SAC-1	SBU TA/SCU Addr Bit 2^{10}
SAC-2	SBU TA/SCU Addr Bit 2^{10}
3000-0	SBU SA/TA/TBA/Fctn Code/Conn Code/Halt On Error
3000-1	SBU SA/TA/TBA/Fctn Code/Conn Code/Halt On Error

Bit 6

SAC-1	SBU TA/SCU Addr Bit 2^{9}
SAC-2	SBU TA/SCU Addr Bit 2^{9}
3000-0	SBU SA/TA/TBA/Fctn Code/Conn Code/Read Bckwrds
3000-1	SBU SA/TA/TBA/Fctn Code/Conn Code/Read Bckwrds

Bit 7

SAC-1	SBU TA/SCU Addr Bit 2^{8}
SAC-2	SBU TA/SCU Addr Bit 2^{8}
3000-0	SBU SA/TA/TBA/Fctn Code/Conn Code
3000-1	SBU SA/TA/TBA/Fctn Code/Conn Code

Bit 8

SAC-1	SBU TA/SCU Addr Bit 2^{7}
SAC-2	SBU TA/SCU Addr Bit 2^{7}
3000-0	SBU SA/TA/Fctn Code/Conn Code
3000-1	SBU SA/TA/Fctn Code/Conn Code

Bit 9

SAC-1	SBU TA/SCU Addr Bit 2^{6}
SAC-2	SBU TA/SCU Addr Bit 2^{6}
3000-0	SBU SA/TA/Fctn Code/Conn Code
3000-1	SBU SA/TA/Fctn Code/Conn Code

Bit A

SAC-1	SBU TA/SCU Addr Bit 2^{5}
SAC-2	SBU TA/SCU Addr Bit 2^{5}
3000-0	SBU SA/TA/Fctn Code/Conn Code
3000-1	SBU SA/TA/Fctn Code/Conn Code

Bit B

SAC-1	SBU TA/SCU Addr Bit 2^{4}
SAC-2	SBU TA/SCU Addr Bit 2^{4}
3000-0	SBU SA/TA/Fctn Code/Conn Code
3000-1	SBU SA/TA/Fctn Code/Conn Code

Bit C

SAC-1	SBU TA/SCU Addr Bit 2^{3}
SAC-2	SBU TA/SCU Addr Bit 2^{3}
3000-0	SBU SA/TA/Fctn Code/Conn Code
3000-1	SBU SA/TA/Fctn Code/Conn Code

Bit D

SAC-1	SBU TA/SCU Addr Bit 2^{2}
SAC-2	SBU TA/SCU Addr Bit 2^{2}
3000-0	SBU SA/TA/Fctn Code/Conn Code
3000-1	SBU SA/TA/Fctn Code/Conn Code

Bit E

SAC-1	SBU TA/SCU Addr Bit 2^{1}
SAC-2	SBU TA/SCU Addr Bit 2^{1}
3000-0	SBU SA/TA/Fctn Code/Conn Code
3000-1	SBU SA/TA/Fctn Code/Conn Code

Bit F

SAC-1	SBU TA/SCU Addr Bit 2^{0}
SAC-2	SBU TA/SCU Addr Bit 2^{0}
3000-0	SBU SA/TA/Fctn Code/Conn Code
3000-1	SBU SA/TA/Fctn Code/Conn Code

† TBA = Table Address

844 STORAGE/MAGNETIC TAPE STATION
INPUT CHANNEL 6

Bit 0		
	Scanner F	3000-0 TA Empty
	844-A	Unit Busy
	844-B	Unit Busy
	CH STAT*	Write Parity Error
Bit 1		
	Scanner F	3000-1 TA Empty
	844-A	Unit 1 Selected
	844-B	Unit 1 Selected
	CH STAT	End of Record
Bit 2		
	Scanner F	SAC-1 Flag
	844-A	Unit 2 Selected
	844-B	Unit 2 Selected
	CH STAT	Reject
Bit 3		
	Scanner F	SAC-2 Flag
	844-A	Unit 3 Selected
	844-B	Unit 3 Selected
	CH STAT	Reply
Bit 4		
	Scanner F	SAC-1 Not Busy
	844-A	Unit 4 Selected
	844-B	Unit 4 Selected
	CH STAT	Reserved
Bit 5		
	Scanner F	SAC-2 Not Busy
	844-A	Unit 5 Selected
	844-B	Unit 5 Selected
	CH STAT	Vert or Long Parity Error
Bit 6		
	Scanner F	3000-0 Response Line
	844-A	Unit 6 Selected
	844-B	Unit 6 Selected
	CH STAT	End of Operation
Bit 7		
	Scanner F	3000-0 Address Compare
	844-A	Unit 7 Selected
	844-B	Unit 7 Selected
	CH STAT	Lost Data
Bit 8		
	Scanner F	3000-1 Response Line
	844-A	Unit 8 Selected
	844-B	Unit 8 Selected
	CH STAT	800 BPI
Bit 9		
	Scanner F	3000-1 Address Compare
	844-A	Unit 9 Selected
	844-B	Unit 9 Selected
	CH STAT	556 BPI
Bit A		
	Scanner F	NU
	844-A	Read Error
	844-B	Read Error
	CH STAT	End of Tape
Bit B		
	Scanner F	NU
	844-A	Tag BCC Error†
	844-B	Tag BCC Error
	CH STAT	Load Point
Bit C		
	Scanner F	NU
	844-A	Timing Chain
	844-B	Timing Chain
	CH STAT	File Mark
Bit D		
	Scanner F	NU
	844-A	SBU Bank Counter 2^2
	844-B	SBU Bank Counter 2^2
	CH STAT	Write Enable
Bit E		
	Scanner F	NU
	844-A	SBU Bank Counter 2^1
	844-B	SBU Bank Counter 2^1
	CH STAT	Busy
Bit F		
	Scanner F	NU
	844-A	SBU Bank Counter 2^0
	844-B	SBU Bank Counter 2^0
	CH STAT	Ready

*CH STAT denotes 3000 Channel Status
†BCC = Block Check Code

844 STORAGE/MAGNETIC TAPE STATION
OUTPUT CHANNEL 6

Bit 0

SAC-1	SBU SA
SAC-2	SBU SA
3000-0	SBU SA
3000-1	SBU SA

Bit 1

SAC-1	SBU TA
SAC-2	SBU TA
3000-0	SBU TA
3000-1	SBU TA

Bit 2

SAC-1	Function
SAC-2	Function
3000-0	Hardware Function
3000-1	Hardware Function

Bit 3

SAC-1	Program Clear
SAC-2	Program Clear
3000-0	Program Clear
3000-1	Program Clear

Bit 4

SAC-1	Request (SCU Write)
SAC-2	Request (SCU Write)
3000-0	Channel Function
3000-1	Channel Function

Bit 5

SAC-1	Accept (SCU Read)
SAC-2	Accept (SCU Read)
3000-0	SBU TBA
3000-1	SBU TBA

Bit 6

SAC-1	Select Status
SAC-2	Select Status
3000-0	SBU Current Address
3000-1	SBU Current Accress

Bit 7

SAC-1	Set Interrupt
SAC-2	Set Interrupt
844-A	Load SBU Address
844-B	Load SBU Address

Bit 8

SAC-1	NU
SAC-2	NU
844-A	Release
844-B	Release

Bit 9

SAC-1	NU
SAC-2	NU
844-A	Function
844-B	Function

Bit A

SAC-1	NU
SAC-2	NU
844-A	Begin Transfer
844-B	Begin Transfer

Bit B

SAC-1	NU
SAC-2	NU
844-A	NU
844-B	NU

Bit C

All Interfaces	Connect Code 2^3

Bit D

All Interfaces	Connect Code 2^2

Bit E

All Interfaces	Connect Code 2^1

Bit F

All Interfaces	Connect Code 2^0

844 STORAGE/MAGNETIC TAPE STATION
INPUT CHANNEL 7

Bit 0

SAC-1	SCU Data/SBU RA Bit 2^{15}
SAC-2	SCU Data/SBU RA Bit 2^{15}
844-A	Finished
844-5	Finished

Bit 1

SAC-1	SCU Data/SBU RA Bit 2^{14}
SAC-2	SCU Data/SBU RA Bit 2^{14}
844-A	Unit 1 On Sector
844-B	Unit 1 on Sector

Bit 2

SAC-1	SCU Data/SBU RA Bit 2^{13}
SAC-2	SCU Data/SBU RA Bit 2^{13}
844-A	Unit 2 On Sector
844-B	Unit 2 On Sector

Bit 3

SAC-1	SCU Data/SBU RA Bit 2^{12}
SAC-2	SCU Data/SBU RA Bit 2^{12}
844-A	Unit 3 On Sector
844-B	Unit 3 On Sector

Bit 4

SAC-1	SCU Data/SBU RA Bit 2^{11}
SAC-2	SCU Data/SBU RA Bit 2^{11}
844-A	Unit 4 On Sector
844-B	Unit 4 On Sector

Bit 5

SAC-1	SCU Data/SBU RA Bit 2^{10}
SAC-2	SCU Data/SBU RA Bit 2^{10}
844-A	Unit 5 On Sector
844-B	Unit 5 On Sector

Bit 6

SAC-1	SCU Data/SBU RA Bit 2^{9}
SAC-2	SBU Data/SBU RA Bit 2^{9}
844-A	Unit 6 On Sector
844-B	Unit 6 On Sector

Bit 7

SAC-1	SCU Data/SBU RA Bit 2^{8}
SAC-2	SCU Data/SBU RA Bit 2^{8}
844-A	Unit 7 On Sector
844-B	Unit 7 On Sector

Bit 8

SAC-1	SCU Data/SBU RA Bit 2^{7}
SAC-2	SCU Data/SBU RA Bit 2^{7}
844-A	Unit 8 On Sector
844-B	Unit 8 On Sector

Bit 9

SAC-1	SCU Data/SBU RA Bit 2^{6}
SAC-2	SCU Data/SBU RA Bit 2^{6}
844-A	Unit 9 On Sector
844-B	Unit 9 On Sector

Bit A

SAC-1	SCU Data/SBU RA Bit 2^{5}
SAC-2	SCU Data/SBU RA Bit 2^{5}
844-A	Ready
844-B	Ready

Bit B

SAC-1	SCU Data/SBU RA Bit 2^{4}
SAC-2	SCU Data/SBU RA Bit 2^{4}
844-A	Transfer Error
844-B	Transfer Error

Bit C

SAC-1	SCU Data/SBU RA Bit 2^{3}
SAC-2	SCU Data/SBU RA Bit 2^{3}
844-A	On Cylinder
844-B	On Cylinder

Bit D

SAC-1	SCU Data/SBU RA Bit 2^{2}
SAC-2	SCU Data/SBU RA Bit 2^{2}
844-A	Seek Error
844-B	Seek Error

Bit E

SAC-1	SCU Data/SBU RA Bit 2^{1}
SAC-2	SCU Data/SBU RA Bit 2^{1}
844-A	Pack Unsafe
844-B	Pack Unsafe

Bit F

SAC-1	SCU Data/SBU RA Bit 2^{0}
SAC-2	SCU Data/SBU RA Bit 2^{0}
844-A	Full
844-B	Full

844 STORAGE/MAGNETIC TAPE STATION
OUTPUT CHANNEL 7

Bit 0		
	SAC-1	SBU SA Bit 2^{15}
	SAC-2	SBU SA Bit 2^{15}
	844-A	NU
	844-B	NU
Bit 1		
	SAC-1	SBU SA Bit 2^{14}
	SAC-2	SBU SA Bit 2^{14}
	844-A	Select Unit 1
	844-B	Select Unit 1
Bit 2		
	SAC-1	SBU SA Bit 2^{13}
	SAC-2	SBU SA Bit 2^{13}
	844-A	Write Tag/Difference Select/Select Unit 2
	844-B	Write Tag/Difference Select/Select Unit 2
Bit 3		
	SAC-1	SBU SA Bit 2^{12}
	SAC-2	SBU SA Bit 2^{12}
	844-A	Write/Clear Fault/Select Unit 3
	844-B	Write/Clear Fault/Select Unit 3
Bit 4		
	SAC-1	SBU SA Bit 2^{11}
	SAC-2	SBU SA Bit 2^{11}
	844-A	Full/Control Select/Select Unit 4
	844-B	Full/Control Select/Select Unit 4
Bit 5		
	SAC-1	SBU SA Bit 2^{10}
	SAC-2	SBU SA Bit 2^{10}
	844-A	Odd/Sector Select/Select Unit 5
	844-B	Odd/Sector Select/Select Unit 5
Bit 6		
	SAC-1	SBU SA Bit 2^{9}
	SAC-2	SBU SA Bit 2^{9}
	844-A	SBU Mem Addr 2^{14}/Cylinder Sel/Sel Unit 6
	844-B	SBU Mem Addr 2^{14}/Cylinder Sel/Sel Unit 6
Bit 7		
	SAC-1	Clear Parity Error/SBU SA Bit 2^{8}
	SAC-2	Clear Parity Error/SBU SA Bit 2^{8}
	844-A	SBU Mem Addr 2^{13}/DSU Addr & Cont 2^{8}/Sel Unit 7
	844-B	SBU Mem Addr 2^{13}/DSU Addr & Cont 2^{8}/Sel Unit 7
Bit 8		
	SAC-1	Clear Illegal/SBU SA Bit 2^{7}
	SAC-2	Clear Illegal/SBU SA Bit 2^{7}
	844-A	SBU Mem Addr 2^{12}/DSU Addr & Cont 2^{7}/Sel Unit 8
	844-B	SBU Mem Addr 2^{12}/DSU Addr & Cont 2^{7}/Sel Unit 8
Bit 9		
	SAC-1	Clear Control Function/SBU SA Bit 2^{6}
	SAC-2	Clear Control Function/SBU SA Bit 2^{6}
	844-A	SBU Mem Addr 2^{11}/DSU Addr & Cont 2^{6}/Sel Unit 9
	844-B	SBU Mem Addr 2^{11}/DSU Addr & Cont 2^{6}/Sel Unit 9
Bit A		
	SAC-1	Master Clear/SBU SA Bit 2^{5}
	SAC-2	Master Clear/SBU SA Bit 2^{5}
	844-A	SBU Mem Addr 2^{10}/DSU Addr & Control 2^{5}
	844-B	SBU Mem Addr 2^{10}/DSU Addr & Control 2^{5}
Bit B		
	SAC-1	SBU SA Bit 2^{4}
	SAC-2	SBU SA Bit 2^{4}
	844-A	SBU Mem Addr 2^{9}/DSU Addr & Control 2^{4}
	844-B	SBU Mem Addr 2^{9}/DSU Addr & Control 2^{4}
Bit C		
	SAC-1	SBU/SAC Select, SBU SA Bit 2^{3}
	SAC-2	SBU/SAC Select, SBU SA Bit 2^{3}
	844-A	SBU Mem Addr 2^{8}/DSU Addr & Control 2^{3}
	844-B	SBU Mem Addr 2^{8}/DSU Addr & Control 2^{3}
Bit D		
	SAC-1	Function Code/SBU SA Bit 2^{2}
	SAC-2	Function Code/SBU SA Bit 2^{2}
	844-A	SBU Mem Addr 2^{7}/DSU Addr & Control 2^{2}
	844-B	SBU Mem Addr 2^{7}/DSU Addr & Control 2^{2}
Bit E		
	SAC-1	Function Code/SBU SA Bit 2^{1}
	SAC-2	Function Code/SBU SA Bit 2^{1}
	844-A	SBU Mem Addr 2^{6}/DSU Addr & Control 2^{1}
	844-B	SBU Mem Addr 2^{6}/DSU Addr & Control 2^{1}
Bit F		
	SAC-1	Function Code/SBU SA Bit 2^{0}
	SAC-2	Function Code/SBU SA Bit 2^{0}
	844-A	SBU Mem Addr 2^{5}/DSU Addr & Control 2^{0}
	844-B	SBU Mem Addr 2^{5}/DSU Addr & Control 2^{0}

UNIT RECORD STATION NORMAL CHANNEL BIT ASSIGNMENTS

INPUT CHANNEL 5
3000 Channel 0

BIT	
0	End of Record
1	Reject
2	Reply
3	Parity Bit
4	2^{11}
5	2^{10}
6	2^{9}
7	2^{8}
8	2^{7}
9	2^{6}
A	2^{5}
B	2^{4}
C	2^{3}
D	2^{2}
E	2^{1}
F	2^{0}

OUTPUT CHANNEL 5
3000 Channel 0

BIT	
0	Function
1	Connect
2	Data Signal
3	Parity Bit
4	2^{11}
5	2^{10}
6	2^{9}
7	2^{8}
8	2^{7}
9	2^{6}
A	2^{5}
B	2^{4}
C	2^{3}
D	2^{2}
E	2^{1}
F	2^{0}

INPUT CHANNEL 6
3000 Channel 1

BIT	
0	End of Record
1	Reject
2	Reply
3	Parity Bit
4	2^{11}
5	2^{10}
6	2^{9}
7	2^{8}
8	2^{7}
9	2^{6}
A	2^{5}
B	2^{4}
C	2^{3}
D	2^{2}
E	2^{1}
F	2^{0}

OUTPUT CHANNEL 6
3000 Channel 1

BIT	
0	Function
1	Connect
2	Data Signal
3	Parity Bit
4	2^{11}
5	2^{10}
6	2^{9}
7	2^{8}
8	2^{7}
9	2^{6}
A	2^{5}
B	2^{4}
C	2^{3}
D	2^{2}
E	2^{1}
F	2^{0}

UNIT RECORD STATION

INPUT CHANNEL 7 3000 Channel		OUTPUT CHANNEL 7 3000 Channel	
BIT		BIT	
0	Response - C0	0	Status Select - C0†
1	Response - C1	1	Status Select - C1†
2	Parity Error (Chan 5)	2	Negate BCD - C0
3	Parity Error (Chan 6)	3	Negate BCD - C1
4	Interrupt Active (Chan 5)	4	Test Mode - C0
5	Interrupt Active (Chan 6)	5	Test Mode - C1
6		6	Word Mark C0
7		7	Word Mark C1
8		8	Suppress A/D C0
9		9	Suppress A/D C1
A		A	Master Clear - C0
B		B	Master Clear - C1
C		C	Write - C0
D		D	Write - C1
E		E	Read - C0
F		F	Read - C1

†C0 and C1 = channel 0 and channel 1

DISPLAY/EDIT STATION NORMAL CHANNEL BIT ASSIGNMENTS

Input Channel 5

Bit	
0	
1	
2	Not mode switch 1
3	Not mode switch 2
4	Not mode switch 3
5	Not mode switch 4
6	Not alert switch
7	Not repeat key
8	Not function strobe
9	Not keyboard data 2^6
A (10)	Not keyboard data 2^5
B (11)	Not keyboard data 2^4
C (12)	Not keyboard data 2^3
D (13)	Not keyboard data 2^2
E (14)	Not keyboard data 2^1
F (15)	Not keyboard data 2^0

Output Channel 5

Bit		
0	SCU-SDU Coupler	Data 2^{15}
1	SCU-SDU Coupler	Data 2^{14} (select offset)
2	SCU-SDU Coupler	Data 2^{13} (group select 2^1)
3	SCU-SDU Coupler	Data 2^{12} (group select 2^0)
4	SCU-SDU Coupler	Data 2^{11} (stack address)
5	SCU-SDU Coupler	Data 2^{10} (stack address)
6	SCU-SDU Coupler	Data 2^9 (stack address)
7	SCU-SDU Coupler	Data 2^8 (stack address)
8	SCU-SDU Coupler	Data 2^7 (stack address)
9	SCU-SDU Coupler	Data 2^6 (stack address)
A (10)	SCU-SDU Coupler	Data 2^5 (stack address)
B (11)	SCU-SDU Coupler	Data 2^4 (stack address)
C (12)	SCU-SDU Coupler	Data 2^3 (stack address)
D (13)	SCU-SDU Coupler	Data 2^2 (stack address)
E (14)	SCU-SDU Coupler	Data 2^1 (stack address)
F (15)	SCU-SDU Coupler	Data 2^0 (stack address)

Input Channel 6

Bit	
0	SCU-SDU coupler parity status
1	SDU not **Parity Bit Left**
2	SDU not **Parity Bit Right**
3	SCU-SDU coupler compare hit status
4	Keyboard interrupt monitor 1
5	Keyboard interrupt monitor 2
6	Keyboard interrupt monitor 3
7	Keyboard interrupt monitor 4
8	Keyboard interrupt monitor 5
9	Keyboard interrupt monitor 6
A (10)	Keyboard interrupt monitor 7
B (11)	Keyboard interrupt monitor 8
C (12)	Keyboard interrupt monitor 9
D (13)	Keyboard interrupt monitor 10
E (14)	Keyboard interrupt monitor 11
F (15)	Keyboard interrupt monitor 12

Output Channel 6

Bit	
0	
1	Strobe Address
2	Disable SCU-SDU coupler parity check
3	Select NC data
4	NC data **Parity Bit Left**
5	NC data **Parity Bit Right**
6	Select character compare
7	Select bit compare
8	Select double compare
9	Strobe keyboard output register (2, 3)
A (10)	Strobe keyboard output register (0, 1)
B (11)	Select keyboard input 2^4
C (12)	Select keyboard input 2^3
D (13)	Select keyboard input 2^2
E (14)	Select keyboard input 2^1
F (15)	Select keyboard input 2^0

Input Channel 7

Bit	
0	Keyboard interrupt monitor 13
1	Keyboard interrupt monitor 14
2	Keyboard interrupt monitor 15
3	Keyboard interrupt monitor 16
4	Keyboard interrupt monitor 17
5	Keyboard interrupt monitor 18
6	Keyboard interrupt monitor 19
7	Keyboard interrupt monitor 20
8	Keyboard interrupt monitor 21
9	Keyboard interrupt monitor 22
A (10)	Keyboard interrupt monitor 23
B (11)	Keyboard interrupt monitor 24
C (12)	Keyboard interrupt monitor 25
D (13)	Keyboard interrupt monitor 26
E (14)	Keyboard interrupt monitor 27
F (15)	Keyboard interrupt monitor 28

Output Channel 7

Bit	
0	Not used
1	Not used
2	Not used
3	Not used
4	Not used
5	Not used
6	Not used
7	Not used
8	Not used
9	Not used
A (10)	Not used
B (11)	Not used
C (12)	Not used
D (13)	Not used
E (14)	Not used
F (15)	Not used

STAR-1B SERVICE STATION NORMAL CHANNEL BIT ASSIGNMENTS

STAR-1B SERVICE STATION
INPUT CHANNEL 5

Bit 0

SAC-1	SBU parity error
SAC-2	SBU parity error
Compare	Null hit
STAR A	Mem P.E. SCU Coupler

Bit 1

SAC-1	$\overline{\text{Bank counter } 2^2}$
SAC-2	$\overline{\text{Bank counter } 2^2}$
Compare	$\overline{\text{Bank counter } 2^2}$
STAR A	Bank counter 2^2

Bit 2

SAC-1	$\overline{\text{Bank counter } 2^1}$
SAC-2	$\overline{\text{Bank counter } 2^1}$
Compare	$\overline{\text{Bank counter } 2^1}$
STAR A	Bank counter 2^1

Bit 3

SAC-1	$\overline{\text{Bank counter } 2^0}$
SAC-2	$\overline{\text{Bank counter } 2^0}$
Compare	$\overline{\text{Bank counter } 2^0}$
STAR A	Bank counter 2^0

Bit 4

SAC-1	Transmission Parity error
SAC-2	Transmission Parity error
Compare	Request one hit
STAR A	Mem P.E. I.O Chan 6

Bit 5

SAC-1	SAC parity error
SAC-2	SAC parity error
Compare	Request zero hit
STAR A	Mem P.E. I/O Chan 5

Bit 6

SAC-1	Illegal
SAC-2	Illegal
Compare	Delete key count 2^9
STAR A	Mem P.E. I/O Chan 4

Bit 7

SAC-1	Busy
SAC-2	Busy
Compare	Delete key count 2^8
STAR A	Mem P.E. I/O Chan 3

Bit 8

SAC-1	Term. address compare
SAC-2	Term. address compare
Compare	Delete key count 2^7
STAR A	Mem P.E. I/O Chan 2

Bit 9

SAC-1	First rank full
SAC-2	First rank full
Compare	Delete key count 2^6
STAR A	Mem P.E. I/O Chan 1

Bit A (10)

SAC-1	Last rank full
SAC-2	Last rank full
Compare	Delete key count 2^5
STAR A	

Bit B (11)

SAC-1	Control function 2^1
SAC-2	Control function 2^1
Compare	Delete key count 2^4
STAR A	

Bit C (12)

SAC-1	Control function 2^0
SAC-2	Control function 2^0
Compare	Delete key count 2^3
STAR A	

Bit D (13)

SAC-1	Control function strobe
SAC-2	Control function strobe
Compare	Delete key count 2^2
STAR A	

Bit E (14)

SAC-1	
SAC-2	
Compare	Delete key count 2^1
STAR A	

Bit F (15)

SAC-1	
SAC-2	
Compare	Delete key count 2^0
STAR A	

STAR -1B SERVICE STATION
OUTPUT CHANNEL 5

Bit 0	
SAC-1	SBU TA/SCU Addr/Data Bit 2^{15}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{15}
MDD A & B	Release/Write tags
Index Compare	Key Identification Bit
Bit 1	
SAC-1	SBU TA/SCU Addr/Data Bit 2^{14}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{14}
MDD A & B	Sector select/Full
Index Compare	Request Identification Bit
Bit 2	
SAC-1	SBU TA/SCU Addr/Data Bit 2^{13}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{13}
MDD A & B	Difference select/Proceed
Bit 3	
SAC-1	SBU TA/SCU Addr/Data Bit 2^{12}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{12}
MDD A & B	Clear/Odd Bank
Bit 4	
SAC-1	SBU TA/SCU Addr/Data Bit 2^{11}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{11}
MDD A & B	Logic No. 2^{2}/Page Addr 2^{14}
Bit 5	
SAC-1	SBU TA/SCU Addr/Data Bit 2^{10}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{10}
MDD A & B	Logic No. 2^{1}/Page Addr 2^{13}
Bit 6	
SAC-1	SBU TA/SCU Addr/Data Bit 2^{9}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{9}
MDD A & B	Logic No. 2^{0}/Page Addr 2^{12}
Bit 7	
SAC-1	SBU TA/SCU Addr/Data Bit 2^{8}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{8}
MDD A & B	Unit select/Page Addr 2^{11}
Bit 8	
SAC-1	SBU TA/SCU Addr/Data Bit 2^{7}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{7}
MDD A & B	Addr and cont. bit 7/Write
Bit 9	
SAC-1	SBU TA/SCU Addr/Data Bit 2^{6}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{6}
MDD A & B	Addr and cont. bit 6
Bit A (10)	
SAC-1	SBU TA/SCU Addr/Data Bit 2^{5}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{5}
MDD A & B	Addr and cont. bit 5/Hdr Addr 2^{10}
Bit B (11)	
SAC-1	SBU TA/SCU Addr/Data Bit 2^{4}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{4}
MDD A & B	Addr and cont. bit 4/Hdr Addr 2^{9}
STAR A	Clear interrupt
Bit C (12)	
SAC-1	SBU TA/SCU Addr/Data Bit 2^{3}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{3}
MDD A & B	Addr and cont. bit 3/Hdr Addr 2^{8}
STAR A	Program clear
Bit D (13)	
SAC-1	SBU TA/SCU Addr/Data Bit 2^{2}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{2}
MDD A & B	Addr and cont. bit 2/Hdr Addr 2^{7}
STAR A	Control function 2^{1}
Bit E (14)	
SAC-1	SBU TA/SCU Addr/Data Bit 2^{1}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{1}
MDD A & B	Addr and cont. bit 1/Hdr Addr 2^{6}
STAR A	Control function 2^{0}
Bit F (15)	
SAC-1	SBU TA/SCU Addr/Data Bit 2^{0}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{0}
MDD A & B	Addr and cont. bit 0/Hdr Addr 2^{5}
STAR A	Control function strobe

STAR-1B SERVICE STATION
INPUT CHANNEL 6

Bit 0		
	Scanner	Mem P.E. I/O Chan
	MDD A	Unit 0 selected
	MDD B	Unit 0 selected
Bit 1		
	Scanner	No Compare Parity Error
	MDD A	Unit 1 selected
	MDD B	Unit 1 selected
Bit 2		
	Scanner	SAC 1 flag
	MDD A	Unit 2 selected
	MDD B	Unit 2 selected
Bit 3		
	Scanner	SAC 2 flag
	MDD A	Unit 3 selected
	MDD B	Unit 3 selected
Bit 4		
	Scanner	SAC 1 not busy
	MDD A	Unit 4 selected
	MDD B	Unit 4 selected
Bit 5		
	Scanner	SAC 2 not busy
	MDD A	Unit 5 selected
	MDD B	Unit 5 selected
Bit 6		
	Scanner	Compare end of table
	MDD A	Unit 6 selected
	MDD B	Unit 6 selected
Bit 7		
	Scanner	Compare not busy
	MDD A	Unit 7 selected
	MDD B	Unit 7 selected
Bit 8		
	Scanner	I/O Interrupt Chan 1
	MDD A	Unit 8 selected
	MDD B	Unit 8 selected
Bit 9		
	Scanner	I/O Interrupt Chan 2
	MDD A	Full
	MDD B	Full
Bit A (10)		
	Scanner	I/O Interrupt Chan 3
	MDD A	Read error
	MDD B	Read error
Bit B (11)		
	Scanner	I/O Interrupt Chan 4
	MDD A	Tag BCC error
	MDD B	Tag BCC error
Bit C (12)		
	Scanner	I/O Interrupt Chan 5
	MDD A	End of page
	MDD B	End of page
Bit D (13)		
	Scanner	I/O Interrupt Chan 6
	MDD A	SBU $\overline{\text{bank count}}\ 2^2$
	MDD B	SBU $\overline{\text{bank count}}\ 2^2$
Bit E (14)		
	Scanner	
	MDD A	SBU $\overline{\text{bank count}}\ 2^1$
	MDD B	SBU $\overline{\text{bank count}}\ 2^1$
Bit F (15)		
	Scanner	
	MDD A	SBU $\overline{\text{bank count}}\ 2^0$
	MDD B	SBU $\overline{\text{bank count}}\ 2^0$

STAR-1B SERVICE STATION
OUTPUT CHANNEL 6

Bit 0		
SAC-1	SBU starting address	
SAC-2	SBU starting address	
Index Compare	SBU starting address	
STAR A	NU	
Bit 1		
SAC-1	SBU terminating address	
SAC-2	SBU terminating address	
Index Compare	SBU terminating address	
STAR A	Select SCU coupler	
Bit 2		
SAC-1	Function	
SAC-2	Function	
Index Compare	Load request register	
STAR A	I/O Channel 6 Select	
Bit 3		
SAC-1	Program clear	
SAC-2	Program clear	
Index Compare	Program clear	
STAR A	I/O Channel 5 Select	
Bit 4		
SAC-1	Request (SCU write)	
SAC-2	Request (SCU write)	
Index Compare	Start search	
STAR A	I/O Channel 4 Select	
Bit 5		
SAC-1	Accept (SCU read)	
SAC-2	Accept (SCU read)	
Index Compare	Write buffers	
STAR A	I/O Channel 3 Select	
Bit 6		
SAC-1	Select status	
SAC-2	Select status	
Index Compare	Search backward	
STAR A	I/O Channel 2 Select	
Bit 7		
SAC-1	Set interrupt	
SAC-2	Set interrupt	
Index Compare	One request	
STAR A	I/O Channel 1 Select	
Bit 8		
SAC-1		
SAC-2		
MDD A		Begin transfer
MDD B		Begin transfer
Bit 9		
SAC-1		
SAC-2		
MDD A		Clear end page
MDD B		Clear end page
Bit A		
SAC-1		
SAC-2		
MDD A		Load SBU address
MDD B		Load SBU address
Bit B		
SAC-1		
SAC-2		
MDD A		Control select
MDD B		Control select
Bit C		
SAC-1		
SAC-2		
MDD A		Function
MDD B		Function
Bit D		
All Interfaces		Connect code 2^2
Bit E		
All Interfaces		Connect code 2^1
Bit F		
All Interfaces		Connect code 2^0

STAR-1B SERVICE STATION
INPUT CHANNEL 7

Bit 0

SAC-1	SCU data/SBU RA bit 2^{15}
SAC-2	SCU data/SBU RA bit 2^{15}
MDD A	Unit 0 on sector/seek error
MDD B	Unit 0 on sector/seek error

Bit 1

SAC-1	SCU data/SBU RA bit 2^{14}
SAC-2	SCU data/SBU RA bit 2^{14}
MDD A	Unit 1 on sector/seek error
MDD B	Unit 1 on sector/seek error

Bit 2

SAC-1	SCU data/SBU RA bit 2^{13}
SAC-2	SCU data/SBU RA bit 2^{13}
MDD A	Unit 2 on sector/seek error
MDD B	Unit 2 on sector/seek error

Bit 3

SAC-1	SCU data/SBU RA bit 2^{12}
SAC-2	SCU data/SBU RA bit 2^{12}
MDD A	Unit 3 on sector/seek error
MDD B	Unit 3 on sector/seek error

Bit 4

SAC-1	SCU data/SBU RA bit 2^{11}
SAC-2	SCU data/SBU RA bit 2^{11}
MDD A	Unit 4 on sector/seek error
MDD B	Unit 4 on sector/seek error

Bit 5

SAC-1	SCU data/SBU RA bit 2^{10}
SAC-2	SCU data/SBU RA bit 2^{10}
MDD A	Unit 5 on sector/seek error
MDD B	Unit 5 on sector/seek error

Bit 6

SAC-1	SCU data/SBU RA bit 2^{9}
SAC-2	SCU data/SBU RA bit 2^{9}
MDD A	Unit 6 on sector/seek error
MDD B	Unit 6 on sector/seek error

Bit 7

SAC-1	SCU data/SBU RA bit 2^{8}
SAC-2	SCU data/SBU RA bit 2^{8}
MDD A	Unit 7 on sector/seek error
MDD B	Unit 7 on sector/seek error

Bit 8

SAC-1	SCU data/SBU RA bit 2^{7}
SAC-2	SCU data/SBU RA bit 2^{7}
MDD A	Unit 8 on sector/seek error
MDD B	Unit 8 on sector/seek error

Bit 9

SAC-1	SCU data/SBU RA bit 2^{6}
SAC-2	SCU data/SBU RA bit 2^{6}
MDD A	Busy
MDD B	Busy

Bit A (10)

SAC-1	SCU data/SBU RA bit 2^{5}
SAC-2	SCU data/SBU RA bit 2^{5}
MDD A	Ready
MDD B	Ready

Bit B (11)

SAC-1	SCU data/SBU RA bit 2^{4}
SAC-2	SCU data/SBU RA bit 2^{4}
MDD A	Seek error
MDD B	Seek error

Bit C (12)

SAC-1	SCU data/SBU RA bit 2^{3}
SAC-2	SCU data/SBU RA bit 2^{3}
MDD A	On cylinder
MDD B	On cylinder

Bit D (13)

SAC-1	SCU data/SBU RA bit 2^{2}
SAC-2	SCU data/SBU RA bit 2^{2}
MDD A	Pack unsafe
MDD B	Pack unsafe

Bit E (14)

SAC-1	SCU data/SBU RA bit 2^{1}
SAC-2	SCU data/SBU RA bit 2^{1}
MDD A	MDD error
MDD B	MDD error

Bit F (15)

SAC-1	SCU data/SBU RA bit 2^{0}
SAC-2	SCU data/SBU RA bit 2^{0}
MDD A	$\overline{\text{Finished}}$
MDD B	$\overline{\text{Finished}}$

STAR-1B SERVICE STATION
OUTPUT CHANNEL 7

Bit 0

SAC-1	SBU starting address bit 2^{15}
SAC-2	SBU starting address bit 2^{15}
Index Compare	Request bit 2^{15}

Bit 1

SAC-1	SBU starting address bit 2^{14}
SAC-2	SBU starting address bit 2^{14}
Index Compare	SA/TA/Request bit 2^{14}

Bit 2

SAC-1	SBU starting address bit 2^{13}
SAC-2	SBU starting address bit 2^{13}
Index Compare	SA/TA/Request bit 2^{13}

Bit 3

SAC-1	SBU starting address bit 2^{12}
SAC-2	SBU starting address bit 2^{12}
Index Compare	SA/TA/Request bit 2^{12}

Bit 4

SAC-1	SBU starting address bit 2^{11}
SAC-2	SBU starting address bit 2^{11}
Index Compare	SA/TA/Request bit 2^{11}

Bit 5

SAC-1	SBU starting address bit 2^{10}
SAC-2	SBU starting address bit 2^{10}
Index Compare	SA/TA/Request bit 2^{10}

Bit 6

SAC-1	SBU starting address bit 2^{9}
SAC-2	SBU starting address bit 2^{9}
Index Compare	SA/TA/Register bit 2^{9}

Bit 7

SAC-1	Clear parity error/SBU starting address bit 2^{8}
SAC-2	Clear parity error/SBU starting address bit 2^{8}
Index Compare	SA/TA/Register bit 2^{8}

Bit 8

SAC-1	Clear illegal/SBU starting address bit 2^{7}
SAC-2	Clear illegal/SBU starting address bit 2^{7}
Index Compare	SA/TA/Request Bit 2^{7}

Bit 9

SAC-1	Clear control function/SBU starting address bit 2^{6}
SAC-2	Clear control function/SBU starting address bit 2^{6}
Index Compare	SA/TA/Request bit 2^{6}

Bit A

SAC-1	Master clear/SBU starting address bit 2^{5}
SAC-2	Master clear/SBU starting address bit 2^{5}
Index Compare	SA/TA/Request bit 2^{5}

Bit B

SAC-1	SBU starting address bit 2^{4}
SAC-2	SBU starting address bit 2^{4}
Index Compare	SA/TA/Request bit 2^{4}

Bit C

SAC-1	Select SAC + $\overline{\text{SCU}}$/SBU starting address bit 2^{3}
SAC-2	Select SAC + $\overline{\text{SCU}}$/SBU starting address bit 2^{3}
Index Compare	SA/TA/Request bit 2^{3}

Bit D

SAC-1	Function code 2^{2}/SBU SA bit 2^{2}
SAC-2	Function code 2^{2}/SBU SA bit 2^{2}
Index Compare	Request bit 2^{2}

Bit E

SAC-1	Function code 2^{1}/SBU SA bit 2^{1}
SAC-2	Function code 2^{1}/SBU SA bit 2^{1}
Index Compare	Request bit 2^{1}

Bit F

SAC-1	Function code 2^{0}/SBU SA bit 2^{0}
SAC-2	Function code 2^{0}/SBU SA bit 2^{0} Data (virtual address bit 2^{33})
Index Compare	Request bit 2^{0}

STAR 100 SERVICE STATION NORMAL CHANNEL BIT ASSIGNMENTS

STAR-100 SERVICE STATION
INPUT CHANNEL 5

Bit 0		**Bit 6**		**Bit B**	
SAC-1	SBU Parity Error	SAC-1	Illegal	SAC-1	Control Function 2^1
SAC-2	SBU Parity Error	SAC-2	Illegal	SAC-2	Control Function 2^1
7000	SBU Parity Error	7000	Drum Sector Count 2^4	7000	Drum Sector Stable
STAR A	Memory P.E. SCU Coupler	STAR A		STAR A	Memory P.E. I/O Channel 4
Bit 1		**Bit 7**		**Bit C**	
SAC-1	$\overline{\text{Bank Counter}}\ 2^2$	SAC-1	Busy	SAC-1	Control Function 2^0
SAC-2	$\overline{\text{Bank Counter}}\ 2^2$	SAC-2	Busy	SAC-2	Control Function 2^0
7000	$\overline{\text{Bank Counter}}\ 2^2$	7000	Drum Sector Count 2^3	7000	Drum EOT 2^1
STAR A	Sta. Int. X $\overline{\text{Bank Counter}}\ 2^2$	STAR A		STAR A	Memory P.E. I/O Channel 5
Bit 2		**Bit 8**		**Bit D**	
SAC-1	$\overline{\text{Bank Counter}}\ 2^1$	SAC-1	Term. Address Compare	SAC-1	Control Function Strobe
SAC-2	$\overline{\text{Bank Counter}}\ 2^1$	SAC-2	Term. Address Compare	SAC-2	Control Function Strobe
7000	$\overline{\text{Bank Counter}}\ 2^1$	7000	Drum Sector Count 2^2	7000	Drum Abnormal
STAR A	Sta. Int. X $\overline{\text{Bank Counter}}\ 2^1$	STAR A	Memory P.E. I/O Channel 1	STAR A	Memory P.E. I/O Channel 6
Bit 3		**Bit 9**		**Bit E**	
SAC-1	$\overline{\text{Bank Counter}}\ 2^0$	SAC-1	First Rank Full	SAC-1	
SAC-2	$\overline{\text{Bank Counter}}\ 2^0$	SAC-2	First Rank Full	SAC-2	
7000	$\overline{\text{Bank Counter}}\ 2^0$	7000	Drum Sector Count 2^1	7000	Drum P.E.
STAR A	Sta. Int. X $\overline{\text{Bank Counter}}\ 2^0$	STAR A	Memory P.E. I/O Channel 2	STAR A	Memory P.E. I/O Channel 7
Bit 4		**Bit A**		**Bit F**	
SAC-1	Transmission Parity Error	SAC-1	Last Rank Full	SAC-1	
SAC-2	Transmission Parity Error	SAC-2	Last Rank Full	SAC-2	
7000	Drum Sector Count 2^6	7000	Drum Sector Count 2^0	7000	Drum Not Ready
STAR A		STAR A	Memory P.E. I/O Channel 3	STAR A	Memory P.E. I/O Channel 8
Bit 5					
SAC-1	SAC Parity Error				
SAC-2	SAC Parity Error				
7000	Drum Sector Count 2^5				
STAR A					

STAR-100 SERVICE STATION
OUTPUT CHANNEL 5

Bit 0

SAC-1	SBU TA/SCU Addr/Data Bit 2^{15}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{15}
7000	SBU SA/TA/Function Bit 2^{15}
Ch. 5 Fanout	

Bit 1

SAC-1	SBU TA/SCU Addr/Data Bit 2^{14}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{14}
7000	SBU SA/TA/Function Bit 2^{14}
Ch. 5 Fanout	

Bit 2

SAC-1	SBU TA/SCU Addr/Data Bit 2^{13}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{13}
7000	SBU SA/TA/Function Bit 2^{13}
Ch. 5 Fanout	

Bit 3

SAC-1	SBU TA/SCU Addr/Data Bit 2^{12}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{12}
7000	SBU SA/TA/Function Bit 2^{12}
Ch. 5 Fanout	

Bit 4

SAC-1	SBU TA/SCU Addr/ Data Bit 2^{11}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{11}
7000	SBU SA/TA/Function Bit 2^{11}
Ch. 5 Fanout	

Bit 5

SAC-1	SBU TA/SCU Addr/Data Bit 2^{10}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{10}
7000	SBU SA/TA/Function Bit 2^{10}
Ch. 5 Fanout	

Bit 6

SAC-1	SBU TA/SCU Addr/Data Bit 2^{9}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{9}
7000	SBU SA/TA/Function Bit 2^{9}
Ch. 5 Fanout	

Bit 7

SAC-1	SBU TA/SCU Addr/Data Bit 2^{8}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{8}
7000	SBU SA/TA/Function Bit 2^{8}
Ch. 5 Fanout	

Bit 8

SAC-1	SBU TA/SCU Addr/Data Bit 2^{7}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{7}
7000	SBU SA/TA/Function Bit 2^{7}
Ch. 5 Fanout	

Bit 9

SAC-1	SBU TA/SCU Addr/Data Bit 2^{6}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{6}
7000	SBU SA/TA/Function Bit 2^{6}
Ch. 5 Fanout	

Bit A

SAC-1	SBU TA/SCU Addr/Data Bit 2^{5}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{5}
7000	SBU SA/TA/Function Bit 2^{5}
Ch. 5 Fanout	

Bit B

SAC-1	SBU TA/SCU Addr/Data Bit 2^{4}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{4}
7000	SBU SA/TA/Function Bit 2^{4}
Ch. 5 Fanout	

Bit C

SAC-1	SBU TA/SCU Addr/Data Bit 2^{3}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{3}
7000	SBU SA/TA/Function Bit 2^{3}
Ch. 5 Fanout	

Bit D

SAC-1	SBU TA/SCU Addr/Data Bit 2^{2}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{2}
7000	SBU SA/TA/Function Bit 2^{2}
Ch. 5 Fanout	

Bit E

SAC-1	SBU TA/SCU Addr/Data Bit 2^{1}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{1}
7000	SBU SA/TA/Function Bit 2^{1}
Ch. 5 Fanout	

Bit F

SAC-1	SBU TA/SCU Addr/Data Bit 2^{0}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{0}
7000	SBU SA/TA/Function Bit 2^{0}
Ch. 5 Fanout	

STAR-100 SERVICE STATION
INPUT CHANNEL 6

Bit 0

Scanner	Memory P. E. I/O Channel
Ch. 5 Fanin	Bit 0
Ch. 7 Fanin	Bit 0

Bit 1

Scanner	Drum Sector Count 2^0
Ch. 5 Fanin	Bit 1
Ch. 7 Fanin	Bit 1

Bit 2

Scanner	SAC-1 Not Busy
Ch. 5 Fanin	Bit 2
Ch. 7 Fanin	Bit 2

Bit 3

Scanner	SAC-2 Not Busy
Ch. 5 Fanin	Bit 3
Ch. 7 Fanin	Bit 3

Bit 4

Scanner	SAC-1 Flag
Ch. 5 Fanin	Bit 4
Ch. 7 Fanin	Bit 4

Bit 5

Scanner	SAC-2 Flag
Ch. 5 Fanin	Bit 5
Ch. 7 Fanin	Bit 5

Bit 6

Scanner	Drum EOT 2^0 †
Ch. 5 Fanin	Bit 6
Ch. 7 Fanin	Bit 6

Bit 7

Scanner	Drum Error
Ch. 5 Fanin	Bit 7
Ch. 7 Fanin	Bit 7

Bit 8

Scanner	I/O Interrupt Channel 1
Ch. 5 Fanin	Bit 8
Ch. 7 Fanin	Bit 8

Bit 9

Scanner	I/O Interrupt Channel 2
Ch. 5 Fanin	Bit 9
Ch. 7 Fanin	Bit 9

Bit A

Scanner	I/O Interrupt Channel 3
Ch. 5 Fanin	Bit A
Ch. 7 Fanin	Bit A

Bit B

Scanner	I/O Interrupt Channel 4
Ch. 5 Fanin	Bit B
Ch. 7 Fanin	Bit B

Bit C

Scanner	I/O Interrupt Channel 5
Ch. 5 Fanin	Bit C
Ch. 7 Fanin	Bit C

Bit D

Scanner	I/O Interrupt Channel 6
Ch. 5 Fanin	Bit D
Ch. 7 Fanin	Bit D

Bit E

Scanner	I/O Interrupt Channel 7
Ch. 5 Fanin	Bit E
Ch. 7 Fanin	Bit E

Bit F

Scanner	I/O Interrupt Channel 8
Ch. 5 Fanin	Bit F
Ch. 7 Fanin	Bit F

† $\overline{\text{EOT}}$ = End of Transfer

STAR-100 SERVICE STATION
OUTPUT CHANNEL 6

Bit 0

SAC-1	SBU Starting Address
SAC-2	SBU Starting Address
7000	SBU Starting Address
STAR A	I/O Chan 8 Select

Bit 1

SAC-1	SBU Terminating Address
SAC-2	SBU Terminating Address
7000	SBU Terminating Address
STAR A	I/O Chan 7 Select

Bit 2

SAC-1	Function
SAC-2	Function
7000	Function
STAR A	I/O Chan 6 Select

Bit 3

SAC-1	Program Clear
SAC-2	Program Clear
7000	Program Clear
STAR A	I/O Chan 5 Select

Bit 4

SAC-1	Request
SAC-2	Request
7000	Disable (Error·Channel Master Clear)
STAR A	I/O Chan 4 Select

Bit 5

SAC-1	Accept
SAC-2	Accept
7000	
STAR A	I/O Chan 3 Select

Bit 6

SAC-1	Select Status
SAC-2	Select Status
7000	
STAR A	I/O Chan 2 Select

Bit 7

SAC-1	Set Interrupt
SAC-2	Set Interrupt
7000	
STAR A	I/O Chan 1 Select

Bit 8

Norm Chan 7 Select

STAR A	Select SCU coupler

Norm Chan 6, Bit 8 Fanout

Bit 9

Norm Chan 5 Select
Norm Chan 6, Bit 9 Fanout

Bit A

Norm Chan 6 Select
Norm Chan 6, Bit A Fanout

Bit B

Scanner Select
Norm Chan 6, Bit B Fanout

Bit C

Connect Code 2^3
Norm Chan 6, Bit C Fanout

Bit D

Connect Code 2^2
Norm Chan 6, Bit D Fanout

Bit E

Connect Code 2^1
Norm Chan 6, Bit E Fanout

Bit F

Connect Code 2^0
Norm Chan 6, Bit F Fanout

STAR-100 SERVICE STATION
INPUT CHANNEL 7

Bit 0		
	SAC-1	SCU Data/SBU RA Bit 2^{15}
	SAC-2	SCU Data/SBU RA Bit 2^{15}
Bit 1		
	SAC-1	SCU Data/SBU RA Bit 2^{14}
	SAC-2	SCU Data/SBU RA Bit 2^{14}
Bit 2		
	SAC-1	SCU Data/SBU RA Bit 2^{13}
	SAC-2	SCU Data/SBU RA Bit 2^{13}
Bit 3		
	SAC-1	SCU Data/SBU RA Bit 2^{12}
	SAC-2	SCU Data/SBU RA Bit 2^{12}
Bit 4		
	SAC-1	SCU Data/SBU RA Bit 2^{11}
	SAC-2	SCU Data/SBU RA Bit 2^{11}
Bit 5		
	SAC-1	SCU Data/SBU RA Bit 2^{10}
	SAC-2	SCU Data/SBU RA Bit 2^{10}
Bit 6		
	SAC-1	SCU Data/SBU RA Bit 2^{9}
	SAC-2	SCU Data/SBU RA Bit 2^{9}
Bit 7		
	SAC-1	SCU Data/SBU RA Bit 2^{8}
	SAC-2	SCU Data/SBU RA Bit 2^{8}
Bit 8		
	SAC-1	SCU Data/SBU RA Bit 2^{7}†
	SAC-2	SCU Data/SBU RA Bit 2^{7}
Bit 9		
	SAC-1	SCU Data/SBU RA Bit 2^{6}
	SAC-2	SCU Data/SBU RA Bit 2^{6}
Bit A		
	SAC-1	SCU Data/SBU RA Bit 2^{5}
	SAC-2	SCU Data/SBU RA Bit 2^{5}
Bit B		
	SAC-1	SCU Data/SBU RA Bit 2^{4}
	SAC-2	SCU Data/SBU RA Bit 2^{4}
Bit C		
	SAC-1	SCU Data/SBU RA Bit 2^{3}
	SAC-2	SCU Data/SBU RA Bit 2^{3}
Bit D		
	SAC-1	SCU Data/SBU RA Bit 2^{2}
	SAC-2	SCU Data/SBU RA Bit 2^{2}
Bit E		
	SAC-1	SCU Data/SBU RA Bit 2^{1}
	SAC-2	SCU Data/SBU RA Bit 2^{1}
Bit F		
	SAC-1	SCU Data/SBU RA Bit 2^{0}
	SAC-2	SCU Data/SBU RA Bit 2^{0}

†RA = Return address

STAR-100 SERVICE STATION
OUTPUT CHANNEL 7

Bit 0

SAC-1	SBU SA Bit 2^{15}
SAC-2	SBU SA Bit 2^{15}
STAR A	
Ch. 7 Fanout	Bit 2^{15}

Bit 1

SAC-1	SBU SA Bit 2^{14}
SAC-2	SBU SA Bit 2^{14}
STAR A	
Ch. 7 Fanout	Bit 2^{14}

Bit 2

SAC-1	SBU SA Bit 2^{13}
SAC-2	SBU SA Bit 2^{13}
STAR A	
Ch. 7 Fanout	Bit 2^{13}

Bit 3

SAC-1	SBU SA Bit 2^{12}
SAC-2	SBU SA Bit 2^{12}
STAR A	
Ch. 7 Fanout	Bit 2^{12}

Bit 4

SAC-1	SBU SA Bit 2^{11}
SAC-2	SBU SA Bit 2^{11}
STAR A	
Ch. 7 Fanout	Bit 2^{11}

Bit 5

SAC-1	SBU SA Bit 2^{10}
SAC-2	SBU SA Bit 2^{10}
STAR A	
Ch. 7 Fanout	Bit 2^{10}

Bit 6

SAC-1	SBU SA Bit 2^{9}
SAC-2	SBU SA Bit 2^{9}
STAR A	
Ch. 7 Fanout	Bit 2^{9}

Bit 7

SAC-1	Clear P.E./SBU SA Bit 2^{8} *
SAC-2	Clear P.E./SBU SA Bit 2^{8}
STAR A	
Ch. 7 Fanout	Bit 2^{8}

Bit 8

SAC-1	Clear Illegal/SBU SA Bit 2^{7}
SAC-2	Clear Illegal/SBU SA Bit 2^{7}
STAR A	
Ch. 7 Fanout	Bit 2^{7}

Bit 9

SAC-1	Clear Cont. Function/SBU SA Bit 2^{6}
SAC-2	Clear Cont. Function/SBU SA Bit 2^{6}
STAR A	
Ch. 7 Fanout	Bit 2^{6}

Bit A

SAC-1	Master Clear/SBU SA Bit 2^{5}
SAC-2	Master Clear/SBU SA Bit 2^{5}
STAR A	
Ch. 7 Fanout	Bit 2^{5}

Bit B

SAC-1	SBU SA Bit 2^{4}
SAC-2	SBU SA Bit 2^{4}
STAR A	Clear Interrupt
Ch. 7 Fanout	Bit 2^{4}

Bit C

SAC-1	Select SAC + $\overline{\text{SCU}}$/SBU SA Bit 2^{3}
SAC-2	Select SAC + $\overline{\text{SCU}}$/SBU SA Bit 2^{3}
STAR A	Program Clear
Ch. 7 Fanout	Bit 2^{3}

Bit D

SAC-1	Function Code/SBU SA Bit 2^{2}
SAC-2	Function Code/SBU SA Bit 2^{2}
STAR A	Control Function 2^{1}
Ch. 7 Fanout	Bit 2^{2}

Bit E

SAC-1	Function Code/SBU SA Bit 2^{1}
SAC-2	Function Code/SBU SA Bit 2^{1}
STAR A	Control Function 2^{0}
Ch. 7 Fanout	Bit 2^{1}

Bit F

SAC-1	Function Code/SBU SA Bit 2^{0}
SAC-2	Function Code/SBU SA Bit 2^{0}
STAR A	Control Function Strobe
Ch. 7 Fanout	Bit 2^{0}

*P.E. = Parity Error

STORAGE (MEDIA) STATION NORMAL CHANNEL BIT ASSIGNMENTS

STORAGE (MEDIA) STATION
INPUT CHANNEL 5

Bit 0		
	SAC-1	SBU Parity Error
	SAC-2	SBU Parity Error
	3000-0, -1	
Bit 1		
	SAC-1	$\overline{\text{Bank Counter } 2^2}$
	SAC-2	$\overline{\text{Bank Counter } 2^2}$
	3000-0, -1	
Bit 2		
	SAC-1	$\overline{\text{Bank Counter } 2^1}$
	SAC-2	$\overline{\text{Bank Counter } 2^1}$
	3000-0, -1	Interrupt Active
Bit 3		
	SAC-1	$\overline{\text{Bank Counter } 2^0}$
	SAC-2	$\overline{\text{Bank Counter } 2^0}$
	3000-0, -1	SBU Memory Parity Error
Bit 4		
	SAC-1	Transmission Parity Error
	SAC-2	Transmission Parity Error
	3000-0, -1	3000 Read Parity Error
Bit 5		
	SAC-1	SAC Parity Error
	SAC-2	SAC Parity Error
	3000-0, -1	Cur Adrs 2^{10}/ Interrupt Line 3
Bit 6		
	SAC-1	Illegal
	SAC-2	Illegal
	3000-0, -1	Cur Adrs 2^9/Interrupt Line 2
Bit 7		
	SAC-1	Busy
	SAC-2	Busy
	3000-0, -1	Cur Adrs 2^8/Interrupt Line 1
Bit 8		
	SAC-1	Term. Adrs Compare
	SAC-2	Term. Adrs Compare
	3000-0, -1	Cur Adrs 2^7/Interrupt Line 0
Bit 9		
	SAC-1	First Rank Full
	SAC-2	First Rank Full
	3000-0, -1	Cur Adrs 2^6/Byte Ctr Rank 3
Bit A		
	SAC-1	Last Rank Full
	SAC-2	Last Rank Full
	3000-0, -1	Cur Adrs 2^5/Byte Ctr Rank 2
Bit B		
	SAC-1	Control Func 2^1
	SAC-2	Control Func 2^1
	3000-0, -1	Cur Adrs 2^4/Byte Ctr Rank 1 *
Bit C		
	SAC-1	Control Func 2^0
	SAC-2	Control Func 2^0
	3000-0, -1	Cur Adrs 2^3/Byte Ctr Rank 0
Bit D		
	SAC-1	Control Func Strobe
	SAC-2	Control Func Strobe
	3000-0, -1	Cur Adrs 2^2/Phase Ctr 2^2
Bit E		
	SAC-1	
	SAC-2	
	3000-0, -1	Cur Adrs 2^1/Phase Ctr 2^1
Bit F		
	SAC-1	
	SAC-2	
	3000-0, -1	Cur Adrs 2^0/Phase Ctr 2^0

$\overline{\text{*Cur Adrs}}$ = Current Address
Ctr = Counter

STORAGE (MEDIA) STATION
OUTPUT CHANNEL 5

Bit 0

SAC-1	SBU TA/SCU Addr/Data Bit 2^{15}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{15}
MDD A & B	Release/Write Tags
3000-0, -1	Compare Bit/Binary/Coded

Bit 1

SAC-1	SBU TA/SCU Addr/Data Bit 2^{14}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{14}
MDD A & B	Sector Select/Full
3000-0, 01	SBU SA/TA/TBA/Read/Write*

Bit 2

SAC-1	SBU TA/SCU Addr/Data Bit 2^{13}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{13}
MDD A & B	Diff Select/Proceed
3000-0, -1	SBU SA/TA/TBA/Connect/Func

Bit 3

SAC-1	SBU TA/SCU Addr/Data Bit 2^{12}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{12}
MDD A & B	Clear/Odd
3000-0, -1	SBU SA/TA/TBA/Odd/Even

Bit 4

SAC-1	SBU TA/SCU Addr/Data Bit 2^{11}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{11}
MDD A & B	Logic No. 2^2/Page Address Bit 2^{14}
3000-0, -1	SBU SA/TA/TBA/Func Code/Disable Request/Conn Code

Bit 5

SAC-1	SBU TA/SCU Addr/Data Bit 2^{10}
SAC-2	SBU TA/SCU Addr/Data Bit 2^{10}
MDD A & B	Logic No. 2^1/Page Address Bit 2^{13}
3000-0, -1	SBU SA/TA/TBA/Func Code/Halt On Error/ Conn Code

Bit 6

SAC-1	SBU TA/SCU Addr/Data Bit 2^9
SAC-2	SBU TA/SCU Addr/Data Bit 2^9
MDD A & B	Logic No. 2^0/Page Address Bit 2^{12}
3000-0, -1	SBU SA/TA/TBA/Func Code/Rd Backwards/ Conn Code

Bit 7

SAC-1	SBU TA/SCU Addr/Data Bit 2^8
SAC-2	SBU TA/SCU Addr/Data Bit 2^8
MDD A & B	Unit Select/Page Address Bit 2^{11}
3000-0, -1	SBU SA/TA/TBA/Func Code/Conn Code

Bit 8

SAC-1	SBU TA/SCU Addr Bit 2^7
SAC-2	SBU TA/SCU Addr Bit 2^7
MDD A & B	Adrs and Cont Bit 7/Write
3000-0, -1	SBU SA/TA/Func Code/Conn Code

Bit 9

SAC-1	SBU TA/SCU Addr Bit 2^6
SAC-2	SBU TA/SCU Addr Bit 2^6
MDD A & B	Adrs & Cont Bit 6
3000-0, -1	SBU SA/TA/Func Code/Conn Code

Bit A

SAC-1	SBU TA/SCU Addr Bit 2^5
SAC-2	SBU TA/SCU Addr Bit 2^5
MDD A & B	Adrs & Cont Bit 5/Hdr Adrs 2^{10} †
3000-0, -1	SBU SA/TA/Func Code/Conn Code

Bit B

SAC-1	SBU TA/SCU Addr Bit 2^4
SAC-2	SBU TA/SCU Addr Bit 2^4
MDD A & B	Adrs & Cont Bit 4/Hdr Adrs 2^9
3000-0, -1	SBU SA/TA/Func Code/Conn Code

Bit C

SAC-1	SBU TA/SCU Addr Bit 2^3
SAC-2	SBU TA/SCU Addr Bit 2^3
MDD A & B	Adrs & Cont Bit 3/Hdr Adrs 2^8
3000-0, -1	SBU SA/TA/Func Code/ Conn Code

Bit D

SAC-1	SBU TA/SCU Addr Bit 2^2
SAC-2	SBU TA/SCU Addr Bit 2^2
MDD A & B	Adrs & Cont Bit 2/Hdr Adrs 2^7
3000-0, -1	SBU SA/TA/Func Code/ Conn Code

Bit E

SAC-1	SBU TA/SCU Addr Bit 2^1
SAC-2	SBU TA/SCU Addr Bit 2^1
MDD A & B	Adrs & Cont Bit 1/Hdr Adrs 2^6
3000-0, -1	SBU SA/TA/Func Code/ Conn Code

Bit F

SAC-1	SBU TA/SCU Addr Bit 2^0
SAC-2	SBU TA/SCU Addr Bit 2^0
MDD A & B	Adrs & Cont Bit 0/Hdr Adrs 2^5
3000-0, -1	SBU SA/TA/Func Code/ Conn Code

*TA = Terminating Address
SA = Starting Address
TBA = Table address

† Hdr Adrs = Header Address

STORAGE (MEDIA) STATION
INPUT CHANNEL 6

Bit 0

Scanner F	3000-0 TA Empty
MDD A & B	Unit 0 Selected
3000 Chan Status	Write P.E.

Bit 1

Scanner F	3000-1 TA Empty
MDD A & B	Unit 1 Selected
3000 Chan Status	End of Record

Bit 2

Scanner F	SAC-1 Flag
MDD A & B	Unit 2 Selected
3000 Chan Status	Reject

Bit 3

Scanner F	SAC-2 Flag
MDD A & B	Unit 3 Selected
3000 Chan Status	Reply

Bit 4

Scanner F	SAC-1 Not Busy
MDD A & B	Unit 4 Selected
3000 Chan Status	Reserved

Bit 5

Scanner F	SAC-2 Not Busy
MDD A & B	Unit 5 Selected
3000 Chan Status	Vert. or Long. Parity Error

*Vert. or Long. = vertical or longitudinal
**BPI = Bits Per Inch

Bit 6

Scanner F	3000-0 Response Line
MDD A & B	Unit 6 Selected
3000 Chan Status	End of Operation

Bit 7

Scanner F	3000-0 Address Compare
MDD A & B	Unit 7 Selected
3000 Chan Status	Lost Data

Bit 8

Scanner F	3000-1 Response Line
MDD A & B	Unit 8 Selected
3000 Chan Status	800 BPI**

Bit 9

Scanner F	3000-1 Address Compare
MDD A & B	Full
3000 Chan Status	556 BPI

Bit A

Scanner F	
MDD A & B	Read Error
3000 Chan Status	End of Tape

Bit B

Scanner F	
MDD A & B	Tag BCC Error
3000 Chan Status	Load Point

Bit C

Scanner F	
MDD A & B	End of Page
3000 Chan Status	File Mark

Bit D

Scanner F	$\overline{\text{Bank Counter } 2^2}$
MDD A & B	SBU $\overline{\text{Bank Counter } 2^2}$
3000 Chan Status	Write Enable

Bit E

Scanner F	$\overline{\text{Bank Counter } 2^1}$
MDD A & B	$\overline{\text{SBU Bank Counter } 2^1}$
3000 Chan Status	Busy

Bit F

Scanner F	$\overline{\text{Bank Counter } 2^0}$
MDD A & B	SBU $\overline{\text{Bank Counter } 2^0}$
3000 Chan Status	Ready

STORAGE (MEDIA) STATION
OUTPUT CHANNEL 6

Bit 0

SAC-1	SBU Starting Address
SAC-2	SBU Starting Address
MDD A & B	
3000-0, -1	SBU Starting Address

Bit 1

SAC-1	SBU Terminating Address
SAC-2	SBU Terminating Address
MDD A & B	
3000-0, -1	SBU Terminating Address

Bit 2

SAC-1	Function
SAC-2	Function
MDD A & B	
3000-0, -1	Hardware Function

Bit 3

SAC-1	Program Clear
SAC-2	Program Clear
MDD A & B	
3000-0, -1	Program Clear

Bit 4

SAC-1	Request (SCU Write)
SAC-2	Request (SCU Write)
MDD A & B	
3000-0, -1	Channel Function

Bit 5

SAC-1	Accept (SCU Read)
SAC-2	Accept (SCU Read)
MDD A & B	
3000-0, -1	SBU Table Address

Bit 6

SAC-1	Select Status
SAC-2	Select Status
MDD A & B	
3000-0, -1	SBU Current Address

Bit 7

SAC-1	Set Interrupt
SAC-2	Set Interrupt
MDD A & B	Load SBU Address

Bit 8

SAC-1	
SAC-2	
MDD A & B	Control Select

Bit 9

SAC-1	
SAC-2	
MDD A & B	Function

Bit A

SAC-1	
SAC-2	
MDD A & B	Begin Transfer

Bit B

SAC-1	
SAC-2	
MDD A & B	Clear End Page

Bit C

All Interfaces	Connect Code 2^3

Bit D

All Interfaces	Connect Code 2^2

Bit E

All Interfaces	Connect Code 2^1

Bit F

All Interfaces	Connect Code 2^0

STORAGE (MEDIA) STATION
INPUT CHANNEL 7

Bit 0		
	SAC-1	SCU Data/SBU RA Bit 2^{15}
	SAC-2	SCU Data/SBU RA Bit 2^{15}
	MDD A & B	Unit 0 on Sector/Seek Error
Bit 1		
	SAC-1	SCU Data/SBU RA Bit 2^{14}
	SAC-2	SCU Data/SBU RA Bit 2^{14}
	MDD A & B	Unit 1 on Sector/Seek Error
Bit 2		
	SAC-1	SCU Data/SBU RA Bit 2^{13}
	SAC-2	SCU Data/SBU RA Bit 2^{13}
	MDD A & B	Unit 2 on Sector/Seek Error
Bit 3		
	SAC-1	SCU Data/SBU RA Bit 2^{12}
	SAC-2	SCU Data/SBU RA Bit 2^{12}
	MDD A & B	Unit 3 on Sector/Seek Error
Bit 4		
	SAC-1	SCU Data/SBU RA Bit 2^{11}
	SAC-2	SCU Data/SBU RA Bit 2^{11}
	MDD A & B	Unit 4 on Sector/Seek Error
Bit 5		
	SAC-1	SCU Data/SBU RA Bit 2^{10}
	SAC-2	SCU Data/SBU RA Bit 2^{10}
	MDD A & B	Unit 5 on Sector/Seek Error
Bit 6		
	SAC-1	SCU Data/SBU RA Bit 2^{9}
	SAC-2	SCU Data/SBU RA Bit 2^{9}
	MDD A & B	Unit 6 on Sector/Seek Error
Bit 7		
	SAC-1	SCU Data/SBU RA Bit 2^{8}
	SAC-2	SCU Data/SBU RA Bit 2^{8}
	MDD A & B	Unit 7 on Sector/Seek Error
Bit 8		
	SAC-1	SCU Data/SBU RA Bit 2^{7}
	SAC-2	SCU Data/SBU RA Bit 2^{7}
	MDD A & B	Unit 8 on Sector/Seek Error
Bit 9		
	SAC-1	SCU Data/SBU RA Bit 2^{6}
	SAC-2	SCU Data/SBU RA Bit 2^{6}
	MDD A & B	Busy
Bit A		
	SAC-1	SCU Data/SBU RA Bit 2^{5}
	SAC-2	SCU Data/SBU RA Bit 2^{5}
	MDD A & B	Ready
Bit B		
	SAC-1	SCU Data/SBU RA Bit 2^{4}
	SAC-2	SCU Data/SBU RA Bit 2^{4}
	MDD A & B	Seek Error
Bit C		
	SAC-1	SCU Data/SBU RA Bit 2^{3}
	SAC-2	SCU Data/SBU RA Bit 2^{3}
	MDD A & B	On Cylinder
Bit D		
	SAC-1	SCU Data/SBU RA Bit 2^{2}
	SAC-2	SCU Data/SBU RA Bit 2^{2}
	MDD A & B	Pack Unsafe
Bit E		
	SAC-1	SCU Data/SBU RA Bit 2^{1}
	SAC-2	SCU Data/SBU RA Bit 2^{1}
	MDD A & B	MDD Error
Bit F		
	SAC-1	SCU Data/SBU RA Bit 2^{0}
	SAC-2	SCU Data/SBU RA Bit 2^{0}
	†MDD A & B	Finished

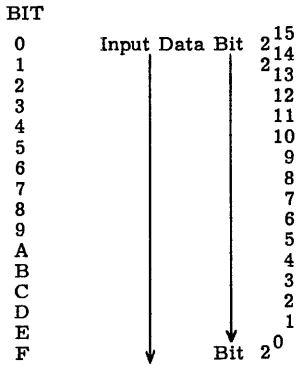
† MDD = Multiple Disk Drive

STORAGE (MEDIA) STATION
OUTPUT CHANNEL 7

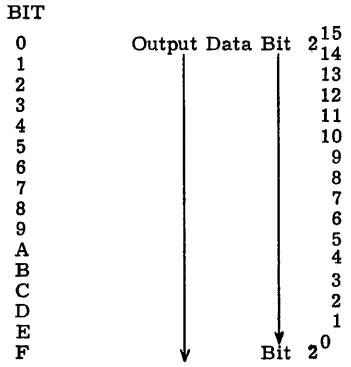
Bit 0		
	SAC-1	SBU SA Bit 2^{15}
	SAC-2	SBU SA Bit 2^{15}
Bit 1		
	SAC-1	SBU SA Bit 2^{14}
	SAC-2	SBU SA Bit 2^{14}
Bit 2		
	SAC-1	SBU SA Bit 2^{13}
	SAC-2	SBU SA Bit 2^{13}
Bit 3		
	SAC-1	SBU SA Bit 2^{12}
	SAC-2	SBU SA Bit 2^{12}
Bit 4		
	SAC-1	SBU SA Bit 2^{11}
	SAC-2	SBU SA Bit 2^{11}
Bit 5		
	SAC-1	SBU SA Bit 2^{10}
	SAC-2	SBU SA Bit 2^{10}
Bit 6		
	SAC-1	SBU SA Bit 2^{9}
	SAC-2	SBU SA Bit 2^{9}
Bit 7		
	SAC-1	Clear P.E./SBU SA Bit 2^{8}
	SAC-2	Clear P.E./SBU SA Bit 2^{8}
Bit 8		
	SAC-1	Clear Illegal/SCU SA Bit 2^{7}
	SAC-2	Clear Illegal/SCU SA Bit 2^{7}
Bit 9		
	SAC-1	Clear Cont Func/SBU SA Bit 2^{6}
	SAC-2	Clear Cont Func/SBU SA Bit 2^{6}
Bit A		
	SAC-1	Master Clear/SBU SA Bit 2^{5}
	SAC-2	Master Clear/SBU SA Bit 2^{5}
Bit B		
	SAC-1	SBU SA Bit 2^{4}
	SAC-2	SBU SA Bit 2^{4}
Bit C		
	SAC-1	Select SAC + $\overline{\text{SCU}}$/SBU SA Bit 2^{3}
	SAC-2	Select SAC + $\overline{\text{SCU}}$/SBU SA Bit 2^{3}
Bit D		
	SAC-1	Func Code 2^{2}/SBU SA Bit 2^{2}
	SAC-2	Func Code 2^{2}/SBU SA Bit 2^{2}
Bit E		
	SAC-1	Func Code 2^{1} SBU SA Bit 2^{1}
	SAC-2	Func Code 2^{1} SBU SA Bit 2^{1}
Bit F		
	SAC-1	Func Code 2^{0}/SBU SA Bit 2^{0}
	SAC-2	Func Code 2^{0}/SBU SA Bit 2^{0}

STAR-100 MAINTENANCE CONTROL UNIT NORMAL CHANNEL BIT ASSIGNMENTS

INPUT CHANNEL 0
MICRODRUM



OUTPUT CHANNEL 0
MICRODRUM



INPUT CHANNEL 1
KEYBOARD

BIT	
0	Control Panel Lockout
1	Character Code 2^6
2	↓
3	↓
4	↓
5	↓
6	↓
7	2^0
8	Alarm Disable (alert)
9	Repeat
A	Lockout Switch
B	Data Strobe
C	Mode 1
D	Mode 2
E	Mode 3
F	Mode 4

OUTPUT CHANNEL 1
DRUM

BIT	
0	Data Mode
1	Write Request
2	Read Request
3	Sector Address 2^4
4	2^3
5	2^2
6	2^1
7	2^0
8	Character Mark
9	Word Address 2^6
A	2^5
B	2^4
C	2^3
D	2^2
E	2^1
F	2^0

INPUT CHANNEL 2
Coupler A/Microdrum

BIT	
0	Coupler Time-out Status
1	Parity Error Fault Status
2	Read Parity Error Status
3	Illegal Status
4	Suspend Status
5	Invalid Status
6	Fault Status
7	Channel Busy Status
8	Coupler Ready Status
9	
A	
B	Sector Address 2^4
C	2^3
D	2^2
E	2^1
F	2^0

OUTPUT CHANNEL 2
Coupler A

BIT	
0	Initiate Functions
1	Function 2^2 B0
2	Function 2^1 B1
3	Function 2^0 B2
4	Interrupt
5	Clear Fault
6	Disable Parity Check
7	Block Clear
8	
9	Coupler Select 2^1
A	Coupler Select 2^0
B	
C	
D	
E	
F	

INPUT CHANNEL 3

BIT	
0	Microdrum Busy
1	Real-Time Strobe
2	Flag Coupler A
3	Flag Coupler B
4	Index Pulse (Printer)
5	Row Pulse (Printer)
6	Paper Strobe (Printer)
7	Start (Printer)
8	Page Eject (Printer)
9	
A	
B	
C	
D	
E	
F	

OUTPUT CHANNEL 3
Microdrum/Keyboard

BIT	
0	Checksum Mode
1	
2	
3	
4	
5	Deadman Strobe
6	Alert Indicator
7	Buzzer
8	Lockout Keyboard
9	Compare Lockout (Drum)
A	Head Address 2^5
B	2^4
C	2^3
D	2^2
E	2^1
F	2^0

INPUT CHANNEL 4

BIT	
0	End of Record
1	Reject
2	Reply
3	Parity Bit
4	Input Data Bit 2^{11}
5	2^{10}
6	2^9
7	2^8
8	2^7
9	2^6
A	2^5
B	2^4
C	2^3
D	2^2
E	2^1
F	2^0

OUTPUT CHANNEL 4

BIT	
0	Function
1	Connect
2	Data Signal
3	Parity Bit
4	Output Data Bit 2^{11}
5	2^{10}
6	2^9
7	2^8
8	2^7
9	2^6
A	2^5
B	2^4
C	2^3
D	2^2
E	2^1
F	2^0

INPUT CHANNEL 5

BIT	
0	Respond Line Active
1	Transmit Parity Error
2	Interrupt Active
3	
4	
5	
6	
7	
8	
9	
A	
B	
C	
D	
E	
F	

OUTPUT CHANNEL 5

BIT	
0	Status Select
1	Negate BCD
2	Test Mode
3	Word Mark
4	Suppress Assy/Disassy
5	Master Clear
6	Write
7	Read
8	Channel Switch 2^1
9	Channel Switch 2^0
A	
B	
C	Head Select
D	Difference Select
E	Control Select
F	Sector Select

INPUT CHANNEL 6A
854 Disk Storage Drive

BIT	
0	Selected Unit 0
1	Selected Unit 1
2	Selected Unit 2
3	Selected Unit 3
4	Selected Unit 4
5	Selected Unit 5
6	Selected Unit 6
7	Selected Unit 7
8	Flag
9	Sector Mark
A	Index Mark
B	Ready
C	Seek Error
D	On Cylinder
E	On Sector
F	Fault

OUTPUT CHANNEL 6A
854 Disk Storage Drive

BIT	
0	Address and Control Bit 7
1	Address and Control Bit 6
2	Address and Control Bit 5
3	Address and Control Bit 4
4	Address and Control Bit 3
5	Address and Control Bit 2
6	Address and Control Bit 1
7	Address and Control Bit 0
8	Reset Flag
9	Hold Sector Mark
A	Hold Index Mark
B	Write
C	Zero Search
D	Unit Select 2^2
E	Unit Select 2^1
F	Unit Select 2^0

INPUT CHANNEL 6B
Printer

BIT	
0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
A	
B	
C	
D	Out of Paper
E	Drum Latch Switch
F	29V Alarm

OUTPUT CHANNEL 6B
Printer

BIT	
0	Start Indicator
1	Select 6 Lines/inch
2	Advance Ribbon
3	Advance Paper
4	Hammer Enable
5	Compare
6	
7	
8	Hammer Address 7
9	Hammer Address 6
A	Hammer Address 5
B	Hammer Address 4
C	Hammer Address 3
D	Hammer Address 2
E	Hammer Address 1
F	Hammer Address 0

INPUT CHANNEL 6C
Coupler B

BIT	
0	Coupler Time-Out Status
1	Parity Error Fault Status
2	Read Parity Error Status
3	Illegal Status
4	Suspend Status
5	Invalid Status
6	Fault Status
7	Channel Busy Status
8	Coupler Ready Status
9	
A	
B	
C	Disable Parity Check
D	
E	
F	

OUTPUT CHANNEL 6C
Coupler B

BIT	
0	Initiate Functions
1	Function 2^2
2	Function 2^1
3	Function 2^0
4	Interrupt
5	Clear Fault
6	Disable Parity Check
7	
8	
9	
A	
B	
C	
D	
E	
F	

INPUT CHANNEL 7
Disk Storage Drive

BIT	
0	Input Data Bit 0
1	Input Data Bit 1
2	Input Data Bit 2
3	Input Data Bit 3
4	Input Data Bit 4
5	Input Data Bit 5
6	Input Data Bit 6
7	Input Data Bit 7
8	Input Data Bit 8
9	Input Data Bit 9
A	Input Data Bit A
B	Input Data Bit B
C	Input Data Bit C
D	Input Data Bit D
E	Input Data Bit E
F	Input Data Bit F

OUTPUT CHANNEL 7
Disk Storage Drive

BIT	
0	Output Data Bit 0
1	Output Data Bit 1
2	Output Data Bit 2
3	Output Data Bit 3
4	Output Data Bit 4
5	Output Data Bit 5
6	Output Data Bit 6
7	Output Data Bit 7
8	Output Data Bit 8
9	Output Data Bit 9
A	Output Data Bit A
B	Output Data Bit B
C	Output Data Bit C
D	Output Data Bit D
E	Output Data Bit E
F	Output Data Bit F

INPUT CHANNEL 8

BIT	
0	CIAR† Bit 00
1	CIAR Bit 01
2	CIAR Bit 02
3	CIAR Bit 03
4	CIAR Bit 04
5	CIAR Bit 05
6	CIAR Bit 06
7	CIAR Bit 07
8	CIAR Bit 08
9	CIAR Bit 09
A	CIAR Bit 10
B	CIAR Bit 11
C	CIAR Bit 12
D	CIAR Bit 13
E	CIAR Bit 14
F	CIAR Bit 15

†CIAR = Current instruction address register

OUTPUT CHANNEL 8

BIT	
0	SAC Master Clear
1	Stop
2	Step
3	Run
4	Store Associative Register
5	Load Associative Register
6	Stream/FP Master Clear*
7	Clear Fault
8	Not used
9	Sync
A	Not used
B	Read
C	Register Select 2^3
D	Register Select 2^2
E	Register Select 2^1
F	Register Select 2^0

*FP = Floating Point

INPUT CHANNEL 9

BIT	
0	CIAR Bit 16
1	CIAR Bit 17
2	CIAR Bit 18
3	CIAR Bit 19
4	CIAR Bit 20
5	CIAR Bit 21
6	CIAR Bit 22
7	CIAR Bit 23
8	CIAR Bit 24
9	CIAR Bit 25
A	CIAR Bit 26
B	CIAR Bit 27
C	CIAR Bit 28
D	CIAR Bit 29
E	CIAR Bit 30
F	CIAR Bit 31

OUTPUT CHANNEL 9

BIT	
0	Sweep Mode
1	Interrupt Gate
2	Block Instruction Execution Overlap
3	Channel Register Select Bit 2^1
4	Channel Register Select Bit 2^0
5	Clock Adjust
6	Decrease Clock Frequency
7	Increase Clock Frequency
8	Delay Trailing Edge
9	Delay Leading Edge
A	Move Clocks
B	Panel Designator Bit 2^4
C	Panel Designator Bit 2^3
D	Panel Designator Bit 2^2
E	Panel Designator Bit 2^1
F	Panel Designator Bit 2^0

INPUT CHANNEL A

BIT	
0	CIAR Bit 32
1	CIAR Bit 33
2	CIAR Bit 34
3	CIAR Bit 35
4	CIAR Bit 36
5	CIAR Bit 37
6	CIAR Bit 38
7	CIAR Bit 39
8	CIAR Bit 40
9	CIAR Bit 41
A	CIAR Bit 42
B	CIAR Bit 43
C	CIAR Bit 44
D	CIAR Bit 45
E	CIAR Bit 46
F	CIAR Bit 47

OUTPUT CHANNEL A

BIT	
0	Not Used/Select Memory Group 2^1/Absolute Addressing
1	Disable Channel 1/Select Memory Group 2^0/Test Associative Registers
2	Disable Channel 2/Delay Memory Strobe/Test Data Paths
3	Disable Channel 3/Advance Memory Strobe/Test Invisible Package
4	Disable Channel 4/Disable Memory Parity Check/Not Used
5	Disable Channel 5/Disable Memory Parity Check/Not Used
6	Disable Channel 6/Disable Memory Parity Check/Not Used
7	Disable Channel 7/Disable Memory Parity Check/Not Used
8	Disable Channel 8/Stop on I/O Parity Fault/Not Used
9	Disable Channel 9/Not Used/Not Used
A	Disable Channel 10/Not Used/Not Used
B	Disable Channel 11/Not Used/Not Used
C	Disable Channel 12/Swap 131K/Not Used
D	Not Used/Phase 16/Not Used
E	Not Used/Swap 262K/Not Used
F	Not Used/Swap 524K/Not Used

INPUT CHANNEL B

BIT	
0	Display Register Bit 00
1	Display Register Bit 01
2	Display Register Bit 02
3	Display Register Bit 03
4	Display Register Bit 04
5	Display Register Bit 05
6	Display Register Bit 06
7	Display Register Bit 07
8	Display Register Bit 08
9	Display Register Bit 09
A	Display Register Bit 10
B	Display Register Bit 11
C	Display Register Bit 12
D	Display Register Bit 13
E	Display Register Bit 14
F	Display Register Bit 15

OUTPUT CHANNEL B

BIT	
0	Not Used
1	External Flag Channel 1
2	External Flag Channel 2
3	External Flag Channel 3
4	External Flag Channel 4
5	External Flag Channel 5
6	External Flag Channel 6
7	External Flag Channel 7
8	External Flag Channel 8
9	External Flag Channel 9
A	External Flag Channel 10
B	External Flag Channel 11
C	External Flag Channel 12
D	Not Used
E	"1" Upper Bounds "0" Lower Bounds
F	"1" Bounds in Upper 524K "0" Bounds in Lower 524K

INPUT CHANNEL C

BIT	
0	Display Register Bit 16
1	Display Register Bit 17
2	Display Register Bit 18
3	Display Register Bit 19
4	Display Register Bit 20
5	Display Register Bit 21
6	Display Register Bit 22
7	Display Register Bit 23
8	Display Register Bit 24
9	Display Register Bit 25
A	Display Register Bit 26
B	Display Register Bit 27
C	Display Register Bit 28
D	Display Register Bit 29
E	Display Register Bit 30
F	Display Register Bit 31

OUTPUT CHANNEL C

BIT	
0	Bounds Sword Address Bit 00
1	Bounds Sword Address Bit 01
2	Bounds Sword Address Bit 02
3	Bounds Sword Address Bit 03
4	Bounds Sword Address Bit 04
5	Bounds Sword Address Bit 05
6	Bounds Sword Address Bit 06
7	Bounds Sword Address Bit 07
8	Bounds Sword Address Bit 08
9	Bounds Sword Address Bit 09
A	Bounds Sword Address Bit 10
B	Bounds Sword Address Bit 11
C	Bounds Sword Address Bit 12
D	Bounds Sword Address Bit 13
E	Bounds Sword Address Bit 14
F	Bounds Sword Address Bit 15

INPUT CHANNEL D

BIT	
0	Display Register Bit 32
1	Display Register Bit 33
2	Display Register Bit 34
3	Display Register Bit 35
4	Display Register Bit 36
5	Display Register Bit 37
6	Display Register Bit 38
7	Display Register Bit 39
8	Display Register Bit 40
9	Display Register Bit 41
A	Display Register Bit 42
B	Display Register Bit 43
C	Display Register Bit 44
D	Display Register Bit 45
E	Display Register Bit 46
F	Display Register Bit 47

OUTPUT CHANNEL D

BIT	
0	Check Bounds on Memory Reads
1	Check Bounds on Memory Writes
2	Check Bounds on CPU References
3	Check Bounds on Channel References
4	Stop CPU on Bounds Hit
5	Enable Bounds Check
6	Count A
7	Count B
8	Clear Counter Overflow Bits
9	Stop - Counter A
A	Stop - Counter B
B	Carry A1
C	Carry A2
D	Carry B1
E	Carry B2
F	"0" Counter A Specifications "1" Counter B Specifications

INPUT CHANNEL E

BIT	
0	Display Register Bit 48
1	Display Register Bit 49
2	Display Register Bit 50
3	Display Register Bit 51
4	Display Register Bit 52
5	Display Register Bit 53
6	Display Register Bit 54
7	Display Register Bit 55
8	Display Register Bit 56
9	Display Register Bit 57
A	Display Register Bit 58
B	Display Register Bit 59
C	Display Register Bit 60
D	Display Register Bit 61
E	Display Register Bit 62
F	Display Register Bit 63

OUTPUT CHANNEL E

BIT	
0	Counter A1/B1 Event Select Bit
1	Counter A1/B1 Event Select Bit
2	Counter A1/B1 Event Select Bit
3	Counter A1/B1 Event Select Bit
4	Counter A1/B1 Event Select Bit
5	Counter A2/B2 Event Select Bit
6	Counter A2/B2 Event Select Bit
7	Counter A2/B2 Event Select Bit
8	Counter A2/B2 Event Select Bit
9	Counter A2/B2 Event Select Bit
A	Not Used
B	All Jobs Enable
C	Monitor Mode Mask
D	Job Mode Mask
E	Data Flag Bit 56 Mask
F	Data Flag Bit 57 Mask

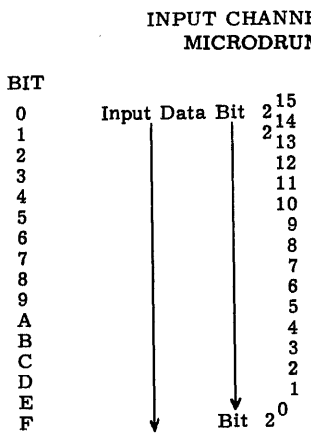
INPUT CHANNEL F

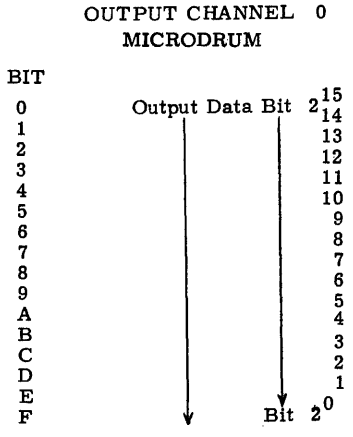
BIT	
0	Memory Parity Fault
1	Microcode Memory Parity Fault
2	Multiple Match
3	Absolute Sword Bounds Hit
4	Event Stop
5	Not Used
6	Not Used
7	Monitor Mode
8	Temperature/Dew Point Alarm
9	MG1 Power Failure
A	Section Power Failure
B	System Power Margin +5%
C	System Power Margin -5%
D	MG2 Power Failure
E	CPU Idle
F	CPU Stopped

OUTPUT CHANNEL F

BIT	
0	Function Select Bit 2^7
1	Function Select Bit 2^6
2	Function Select Bit 2^5
3	Function Select Bit 2^4
4	Function Select Bit 2^3
5	Function Select Bit 2^2
6	Function Select Bit 2^1
7	Function Select Bit 2^0
8	Mask Bit 2^7
9	Mask Bit 2^6
A	Mast Bit 2^5
B	Mask Bit 2^4
C	Mask Bit 2^3
D	Mask Bit 2^2
E	Mask Bit 2^1
F	Mask Bit 2^0

STAR-65 MAINTENANCE CONTROL UNIT NORMAL CHANNEL BIT ASSIGNMENTS


INPUT CHANNEL 0
MICRODRUM
BIT
0
1
2
3
4
5
6
7
8
9
A
B
C
D
E
F
Input Data Bit
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1
Bit 2^0


OUTPUT CHANNEL 0
MICRODRUM
BIT
0
1
2
3
4
5
6
7
8
9
A
B
C
D
E
F
Output Data Bit
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1
Bit 2^0

INPUT CHANNEL 1
KEYBOARD

BIT	
0	Control Panel Lockout
1	Character Code 2^6
2	
3	
4	
5	
6	
7	2^0
8	Alarm Disable (alert)
9	Repeat
A	
B	Data Strobe
C	Mode 1
D	Mode 2
E	Mode 3
F	Mode 4

OUTPUT CHANNEL 1
DRUM

BIT	
0	Data Mode
1	Write Request
2	Read Request
3	Sector Address 2^4
4	2^3
5	2^2
6	2^1
7	2^0
8	Character Mark
9	Word Address 2^6
A	2^5
B	2^4
C	2^3
D	2^2
E	2^1
F	2^0

INPUT CHANNEL 2
Coupler A/Microdrum

BIT	
0	Coupler Time-out Status
1	Parity Error Fault Status
2	Read Parity Error Status
3	Illegal Status
4	Suspend Status
5	Invalid Status
6	Fault Status
7	Channel Busy Status
8	Coupler Ready Status
9	
A	
B	Sector Address 2^4
C	2^3
D	2^2
E	2^1
F	2^0

OUTPUT CHANNEL 2
Coupler A

BIT	
0	Initiate Functions
1	Function 2^2 B0
2	Function 2^1 B1
3	Function 2^0 B2
4	Interrupt
5	Clear Fault
6	Disable Parity Check
7	
8	
9	Coupler Select 2^1
A	Coupler Select 2^0
B	
C	
D	
E	
F	

INPUT CHANNEL 3
Scanner

BIT	
0	Microdrum Busy
1	Real-Time Strobe
2	Channel Flag
3	
4	Index Pulse (Printer)
5	Row Pulse (Printer)
6	
7	Start (Printer)
8	Page Eject (Printer)
9	CM2 Stop
A	CM1 Stop
B	Keypoint
C	
D	
E	
F	

OUTPUT CHANNEL 3
Microdrum/Keyboard

BIT	
0	1 = Checksum Mode
1	
2	
3	
4	
5	Deadman Strobe
6	Alert Indicator
7	Buzzer
8	Lockout Keyboard Indicator
9	Compare Lockout (Microdrum)
A	Head Address 2^5
B	2^4
C	2^3
D	2^2
E	2^1
F	2^0

INPUT CHANNEL 4

BIT	
0	End of Record
1	Reject
2	Reply
3	Parity Bit
4	Input Data Bit 2^{11}
5	2^{10}
6	2^9
7	2^8
8	2^7
9	2^6
A	2^5
B	2^4
C	2^3
D	2^2
E	2^1
F	2^0

OUTPUT CHANNEL 4

BIT	
0	Function
1	Connect
2	Data Signal
3	Parity Bit
4	Output Data Bit 2^{11}
5	2^{10}
6	2^9
7	2^8
8	2^7
9	2^6
A	2^5
B	2^4
C	2^3
D	2^2
E	2^1
F	2^0

INPUT CHANNEL 5
3000 Channel

BIT	
0	Respond Line Active
1	Transmit Parity Error
2	Interrupt Active
3	
4	
5	
6	
7	
8	
9	
A	
B	
C	
D	
E	
F	

OUTPUT CHANNEL 5
3000 Channel/854 Disk Drive

BIT		
0	Status Select	
1	Negate BCD	
2	Test Mode	
3	Word Mark	
4	Suppress Assy/Disassy	
5	Master Clear	
6	Write	
7	Read	
8	Channel 6 Switch Select	Bit 8 and 9 = 0, Disk
9	Channel 6 Switch Select	Bit 8 = 0, Bit 9 = 1, Printer
A		
B		
C	Head Select	
D	Difference Select	
E	Control Select	
F	Sector Select	

INPUT CHANNEL 6
Disk/Printer

BIT	
0	Select Unit 0
1	Select Unit 1
2	Select Unit 2
3	Select Unit 3
4	Reserved
5	Reserved
6	Reserved
7	Reserved
8	Flag
9	Sector Mark
A	Index Mark
B	Selected Ready
C	Seek Error
D	On Cylinder/Out of Paper
E	On Sector/Drum Latch Switch
F	Printer Fault/29 V Alarm

OUTPUT CHANNEL 6
Disk/Printer

BIT	
0	Address Bit 0/Start Ind.
1	Address Bit 1/Select 6 Lines
2	Address Bit 2/Advance Ribbon
3	Address Bit 3/Advance Paper
4	Address Bit 4/Hammer Enable
5	Address Bit 5/Compare
6	Address Bit 6
7	Address Bit 7
8	Reset Flag/Hammer Address Bit 7
9	Hold Sector Mark/Hammer Address Bit 6
A	Hold Index Mark/Hammer Address Bit 5
B	Write/Hammer Address Bit 4
C	Zero Search/Hammer Address Bit 3
D	Unit Select 2^2/Hammer Address Bit 2
E	Unit Select 2^1/Hammer Address Bit 1
F	Unit Select 2^0/Hammer Address Bit 0

INPUT CHANNEL 7
Disk Storage Drive

BIT	
0	Input Data Bit 0
1	Input Data Bit 1
2	Input Data Bit 2
3	Input Data Bit 3
4	Input Data Bit 4
5	Input Data Bit 5
6	Input Data Bit 6
7	Input Data Bit 7
8	Input Data Bit 8
9	Input Data Bit 9
A	Input Data Bit A
B	Input Data Bit B
C	Input Data Bit C
D	Input Data Bit D
E	Input Data Bit E
F	Input Data Bit F

OUTPUT CHANNEL 7
Disk Storage Drive

BIT	
0	Output Data Bit 0
1	Output Data Bit 1
2	Output Data Bit 2
3	Output Data Bit 3
4	Output Data Bit 4
5	Output Data Bit 5
6	Output Data Bit 6
7	Output Data Bit 7
8	Output Data Bit 8
9	Output Data Bit 9
A	Output Data Bit A
B	Output Data Bit B
C	Output Data Bit C
D	Output Data Bit D
E	Output Data Bit E
F	Output Data Bit F

INPUT CHANNEL 8

BIT

0 Not Used
1
2
3
4
5
6
7
8
9
A
B
C
D
E
F

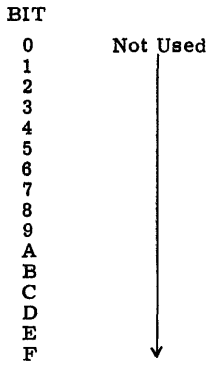
OUTPUT CHANNEL 8

BIT

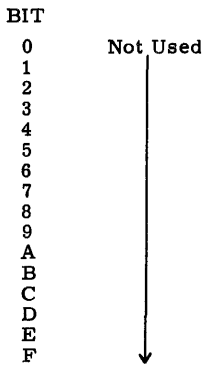
0 Not Used
1
2
3
4
5
6
7
8
9
A
B
C
D
E
F

†CIAR = Current instruction address register

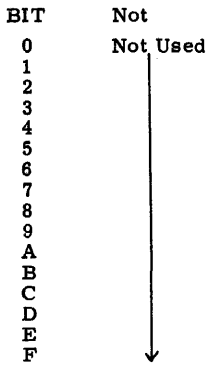
INPUT CHANNEL 9



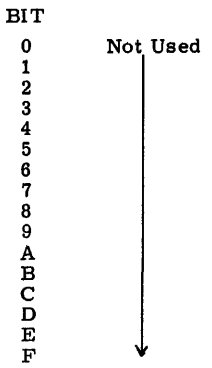
OUTPUT CHANNEL 9



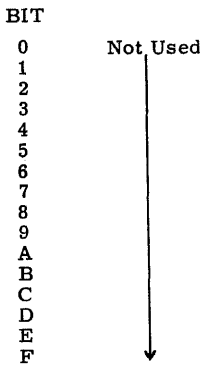
INPUT CHANNEL A



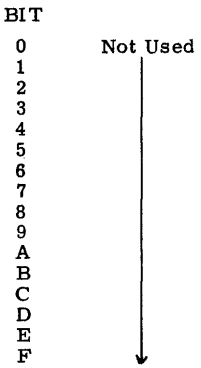
OUTPUT CHANNEL A



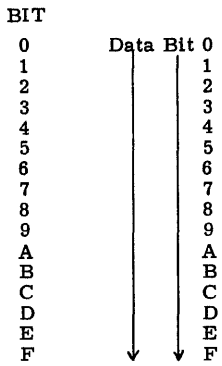
INPUT CHANNEL B



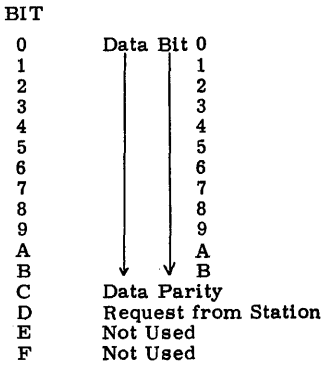
OUTPUT CHANNEL B



INPUT CHANNEL C
(Control Interface)



OUTPUT CHANNEL C
(Control Interface)



INPUT CHANNEL D
(Control Interface)

BIT	
0	Input Data Parity
1	Accept from Processor
2	Monitor Mode
3	Not Used
4	
5	
6	
7	
8	
9	
A	
B	
C	
D	
E	
F	

OUTPUT CHANNEL D
(Control Interface)

BIT	
0	Function 0
1	1
2	2
3	3
4	4
5	5
6	Function Parity
7	I/O Maintenance
8	MC to Processor
9	MC to I/O 1
A	MC to I/O 2
B	MC to I/O 3
C	MC to Storage 0
D	MC to Storage 1
E	MC to Storage 2
F	MC to Storage 3

INPUT CHANNEL E
(Control Interface)

BIT	
0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
A	
B	
C	
D	
E	
F	

OUTPUT CHANNEL E
(Control Interface)

BIT	
0	Bank Mode 0
1	Bank Mode 1
2	Bank Mode 2
3	Bank Mode 3
4	Not Used
5	Not Used
6	Not Used
7	Not Used
8	RCF 0 to Storage 0
9	RCF 1 to Storage 0
A	RCF 0 to Storage 1
B	RCF 1 to Storage 1
C	RCF 0 to Storage 2
D	RCF 1 to Storage 2
E	RCF 0 to Storage 3
F	RCF 1 to Storage 3

INPUT CHANNEL F (Control Interface) BIT		OUTPUT CHANNEL F (Control Interface) BIT	
0	Not Used	0	Clear 0 (Clear Data Bits 0-B)
1		1	Clear 1 (Clear Data Bits C-F, Parity, and Function)
2		2	
3		3	
4		4	Set 0 (Strobe Chan C Bits 0-F and Chan D Bits 0-7)
5		5	Set 1 (Strobe Chan D Bits 8-F, Chan E Bits 0-F, and Chan F Bits 8-F
6		6	
7		7	
8		8	RCF 0 To I/O 0
9		9	RCF 1 To I/O 0
A		A	RCF 0 To I/O 1
B		B	RCF 1 To I/O 1
C		C	RCF 0 To I/O 2
D		D	RCF 1 To I/O 2
E		E	
F		F	

INTERFACE NORMAL CHANNEL BIT ASSIGNMENTS B

Appendix B lists normal channel bit assignments by interface for several of the most common interfaces in the SBU. For a listing of all bits used in any particular station, refer to appendix A.

TABLE B-1. SAC-1 AND SAC-2 SBU INTERFACE NIC BIT ASSIGNMENTS

	Paging (Drum) Station†	Disk Station	844 Storage Magnetic Tape Station	STAR-1B Service Station	STAR-100 Service Station	Storage (Media) Station	844 Service Station
NIC-5, Bit 0	SBU Parity Error						
1	$\overline{\text{Bank Counter}}$ 2^2	(Same as Drum Station)	(Same as Drum Station)	(Same as Drum Station)	(Same as Drum Station)	(Same as Drum Station)	(Same as Drum Station)
2	2^1						
3	2^0						
4	Transmission Parity Error						
5	SAC Parity Error						
6	Illegal						
7	Busy						
8	Term Addr Compare						
9	First Rank Full						
A	Last Rank Full						
B	Control Function 2^1						
C	Control Function 2^0						
D	Control Function Strobe						
E	NU						
F	NU						
NIC-6, Bit 0							
1							
2			SAC-1 Flag	(Same as 844 Storage Station)	SAC-1 Not Busy	(Same as 844 Storage Station)	(Same as STAR-100 Service Station)
3			SAC-2 Flag		SAC-2 Not Busy		
4			SAC-1 Not Busy		SAC-1 Flag		
5			SAC-2 Not Busy		SAC-2 Flag		
6							
7							
8		SAC-1 Not Busy					
9		SAC-2 Not Busy					
A		SAC-1 Flag					
B	SAC-1 Not Busy	SAC-2 Flag					
C	SAC-1 Flag						
D							
E							
F							
NIC-7, Bit 0	SCU Data/SBU $\overline{\text{RA Bit}}$ 2^{15}	(Same as Drum Station)	(Same as Drum Station)	(Same as Drum Station)	(Same as Drum Station)	(Same as Drum Station)	(Same as Drum Station)
1							
2							
3							
4							
5							
6							
7							
8							
9							
A							
B							
C							
D							
E							
F	2^0						

NU = not used
RA = return address
† Paging station uses SAC-1 interface only

TABLE B-2. SAC-1 AND SAC-2 SBU INTERFACE NOC BIT ASSIGNMENTS

	Paging (Drum) Station†	Disk Station	844 Storage Magnetic Tape Station	844 Service Station	STAR-1B Service Station	STAR-100 Service Station	Storage (Media) Station
NOC-5, Bit 0	SBU TA/SCU Addr/Data Bit 2^{15}			SBU TA/Data Bit 2^{15}			
1	↓	(Same as Drum Station)	(Same as Drum Station)	↓	(Same as Drum Station)	(Same as Drum Station)	(Same as Drum Station)
2	↓	↓	↓	↓	↓	↓	↓
3	↓	↓	↓	↓	↓	↓	↓
4	↓	↓	↓	↓	↓	↓	↓
5	↓	↓	↓	↓	↓	↓	↓
6	↓	↓	↓	↓	↓	↓	↓
7	↓	↓	↓	↓	↓	↓	↓
8	↓	↓	↓	↓	↓	↓	↓
9	↓	↓	↓	↓	↓	↓	↓
A	↓	↓	↓	↓	↓	↓	↓
B	↓	↓	↓	↓	↓	↓	↓
C	↓	↓	↓	↓	↓	↓	↓
D	↓	↓	↓	↓	↓	↓	↓
E	↓	↓	↓	↓	↓	↓	↓
F	2^0			2^0			
NOC-6, Bit 0	SBU Starting Address	(Same as Drum Station)	(Same as Drum Station)	(Same as Drum Station)	SBU Starting address	(Same as Drum Station)	(Same as Drum Station)
1	SBU Terminating Address	↓	↓	↓	SBU TA	↓	↓
2	Function	↓	↓	↓	Function	↓	↓
3	Program Clear	↓	↓	↓	Program Clear	↓	↓
4	Request	↓	↓	↓	Request	↓	↓
5	Accept	↓	↓	↓	Accept	↓	↓
6	Select Status	↓	↓	↓	Select Status	↓	↓
7	Set Interrupt	↓	↓	↓	Set Interrupt	↓	↓
8	↓	↓	↓	↓	↓	↓	↓
9	↓	↓	↓	↓	↓	↓	↓
A	↓	↓	↓	↓	↓	↓	↓
B	↓	↓	↓	↓	↓	↓	↓
C	Connect Code 2^3	↓	↓	↓	Connect Code 2^3	↓	↓
D	↓	↓	↓	↓	↓	↓	↓
E	↓	↓	↓	↓	↓	↓	↓
F	2^0				2^0		
NOC-7, Bit 0	SBU SA Bit 2^{15}	(Same as Drum Station)	(Same as Drum Station)	(Same as Drum Station)	(Same as Drum Station)	(Same as Drum Station)	(Same as Drum Station)
1	↓	↓	↓	↓	↓	↓	↓
2	↓	↓	↓	↓	↓	↓	↓
3	↓	↓	↓	↓	↓	↓	↓
4	↓	↓	↓	↓	↓	↓	↓
5	↓	↓	↓	↓	↓	↓	↓
6	2^9	↓	↓	↓	↓	↓	↓
7	Clear Parity Error/ SBU SA Bit 2^8	↓	↓	↓	↓	↓	↓
8	Clear Illegal/SBU SA Bit 2^7	↓	↓	↓	↓	↓	↓
9	Clear Cont. Func./SBU SA Bit 2^6	↓	↓	↓	↓	↓	↓
A	Master Clear/SBU SA Bit 2^5	↓	↓	↓	↓	↓	↓
B	SBU SA Bit 2^4	↓	↓	↓	↓	↓	↓
C	Select SAC + $\overline{\text{SCU}}$/SBU SA Bit 2^3	↓	↓	↓	↓	↓	↓
D	Function Code/SBU SA Bit 2^2	↓	↓	↓	↓	↓	↓
E	2^1	↓	↓	↓	↓	↓	↓
F	2^0						

Addr = address
SA = starting address
TA = terminating address
† Paging station uses SAC-1 interface only

TABLE B-3. 7000-0 AND 7000-1 SBU INTERFACE NIC BIT ASSIGNMENTS

		Paging (Drum) Station	Disk Station	STAR-100 Service Station †
NIC-5, Bit	0	SBU Parity Error	SBU Parity Error	(Same as Drum Station)
	1	Bank Counter 2^2	Bank Counter 2^2	
	2	2^1	2^1	
	3	2^0	2^0	
	4	Drum Sector Count 2^6	Disk Sector Count 2^5	
	5	2^5	2^4	
	6	2^4	2^3	
	7	2^3	2^2	
	8	2^2	2^1	
	9	2^1	2^0	
	A	2^0	NU	
	B	Drum Sector Stable	NU	
	C	Drum EOT Bit 2^1	Disk Not on Cylinder	
	D	Drum Abnormal	Disk Abnormal	
	E	Drum Parity Error	Disk Parity Error	
	F	Drum Not Ready	Disk Not Ready	
NIC-6, Bit	0	7000-1 Drum SC Bit 2^0	7000-0 Disk on Cylinder	
	1	7000-1 Drum EOT Bit 2^0	7000-0 Disk Error	Drum SC 2^0
	2	7000-0 Drum SC Bit 2^0	7000-0 Disk EOT Bit 2^1	
	3	7000-0 Drum EOT Bit 2^0	7000-0 Disk EOT Bit 2^0	
	4		7000-1 Disk on Cylinder	
	5		7000-1 Disk Error	
	6		7000-1 Disk EOT 2^1	Drum EOT 2^0
	7		7000-1 Disk EOT 2^0	Drum Error
	8			
	9	7000-1 Drum Error		
	A	7000-0 Drum Error		
	B			
	C			
	D			
	E			
	F			
NIC-7, Bit	0	NU		NU
	1		Return Function 2^2	
	2		2^1	
	3		Return Function 2^0	
	4			
	5			
	6			
	7			
	8			
	9			
	A			
	B			
	C			
	D			
	E			
	F			

EOT = end of transfer
NU = not used
SC = sector count
† Only one 7000 interface used

TABLE B-4. 7000-0 AND 7000-1 SBU INTERFACE NOC BIT ASSIGNMENTS

		Paging (Drum) Station	Disk Station	STAR-100 Service Station
NOC-5, Bit	0	EDS/SDS	(Same as Drum Station)	(Same as Drum Station)
	1	SBU SA/TA/Function Bit 2^{14}		
	2			
	3			
	4			
	5			
	6			
	7			
	8			
	9			
	A			
	B			
	C			
	D			
	E			
	F	2^0		
NOC-6, Bit	0	SBU Starting Address	(Same as Drum Station)	SBU Starting Address
	1	SBU Terminating address		SBU Terminating Addr
	2	Function		Function
	3	Program Clear		Program Clear
	4	NU		Disable Error + Channel MC
	5			
	6			
	7			
	8			
	9			
	A			
	B			
	C	Connect Code 2^3		Connect Code 2^3
	D			
	E			
	F	2^0		2^0
NOC-7, Bit	0	NU	EDS/SDS	
	1		SBU SA TA	NU
	2		Function Bit 2^{14}	
	3			
	4			
	5			
	6			
	7			
	8			
	9			
	A			
	B			
	C			
	D			
	E			
	F		2^0	

NU = not used
SA = starting address
TA = terminating address
EDS/SDS = end data sequence/start data sequence

TABLE B-5. 3000-0 AND 3000-1 SBU INTERFACE NIC BIT ASSIGNMENTS

	844 Storage/Mag Tape Station	Storage (Media) Station
NIC-5, Bit 0	NU	(Same as Mag Tape Station)
1	NU	
2	Interrupt Active	
3	SBU Parity Error	
4	Read Parity Error	
5	Current Addr 2^{10}/Int Line 3	
6	2^{9} 2	
7	2^{8} 1	
8	2^{7} 0	
9	2^{6}/Byte Counter Rank 3	
A	2^{5} 2	
B	2^{4} 1	
C	2^{3} 0	
D	2^{2}/Phase Counter 2^{2}	
E	2^{1} 2^{1}	
F	2^{0} 2^{0}	
NIC-6, Bit 0	Write Parity Error/TA Empty 0	(Same as Mag Tape Station)
1	End of Record/TA Empty 1	
2	Reject	
3	Reply	
4	Reserved	
5	Vert or Long Parity Error	
6	End of Operation/Response, 3000-0	
7	Lost Data/Adrs Comp, 3000-0	
8	800 BPI/Response, 3000-1	
9	556 BPI/Adrs Comp, 3000-1	
A	End of Tape	
B	Load Point	
C	File Mark	
D	Write Enable	
E	Busy	
F	Ready	
NIC-7, Bit 0	NU	NU
1		
2		
3		
4		
5		
6		
7		
8		
9		
A		
B		
C		
D		
E		
F		

Addr = address
BPI = bits per inch
C1 = channel 1
Err = error
Int = interrupt
Vert or Long = vertical or longitudinal
NU = not used

TABLE B-6. 3000-0 AND 3000-1 SBU INTERFACE NOC BIT ASSIGNMENTS

	844 Storage/Mag Tape Station	Storage (Media) Station
NOC-5, Bit 0	Compare Bit/Binary/Coded	(Same as Mag Tape Station)
1	SBU SA/TA/TBA/Read/Write	
2	SBU SA/TA/TBA/CONN/FCTN	
3	/Odd/Even SEL	
4	/FCTN Code/Dis Req/Conn Code/	
5	/HOE	
6	/RE BA	
7		
8		
9		
A		
B		
C		
D		
E		
F		
NOC-6, Bit 0	SBU SA	(Same as Mag Tape Station)
1	SBU TA	
2	Hardware Function	
3	Program Clear	
4	Channel Function	
5	SBU TBA	
6	SBU Current Address	
7		
8		
9		
A		
B		
C	Connect Code 2^{3}	
D		
E	2^{0}	
F		
NOC-7, Bit 0	NU	NU
1		
2		
3		
4		
5		
6		
7		
8		
9		
A		
B		
C		
D		
E		
F		

C1 = channel 1
HOE = halt on error
CONN = connect
DIS Req = disable request
FCTN = function
RE BA = Read backward
SA = starting address
SEL = select
TA = terminating address
TBA = table address

TABLE B-7. INDEX COMPARE SBU INTERFACE NIC BIT ASSIGNMENTS

		Paging (Drum) Station	STAR-1B Service Station
NIC-5, Bit	0	SBU Parity Error	Null Hit
	1	Bank Counter 2^2	Bank Counter 2^2
	2	2^1	2^1
	3	2^0	2^0
	4		Request One Hit
	5		Request Zero Hit
	6		Delete Key Count 2^9
	7		
	8		
	9		
	A		
	B		
	C		
	D		
	E		
	F		2^0
NIC-6, Bit	0		
	1		No Compare Parity Error
	2		
	3		
	4	No Compare Parity Error	
	5	Compare Req-1 Hit	
	6	Compare End of Table	Compare End of Table
	7	Compare Req-0 Hit	Compare Not Busy
	8		
	9		
	A		
	B		
	C		
	D	Compare Not Busy	
	E		
	F		
NIC-7, Bit	0		NU
	1		
	2		
	3		
	4		
	5		
	6	Delete Key Count Bit 2^9	
	7		
	8		
	9		
	A		
	B		
	C		
	D		
	E		
	F	2^0	

NU = not used
REQ = request

TABLE B-8. INDEX COMPARE SBU INTERFACE NOC BIT ASSIGNMENTS

		Paging (Drum) Station	STAR-1B Service Station
NOC-5, Bit	0		Key ID
	1		Request ID
	2		
	3		
	4		
	5		
	6		
	7		
	8		
	9		
	A		
	B		
	C		
	D		
	E	Key ID	
	F	Request ID	
NOC-6, Bit	0	SBU Starting Address	(Same as Drum Station)
	1	SBU Terminating Address	
	2	Load Request Register	
	3	Program Clear	
	4	Start Search	
	5	Write Buffers	
	6	Search Backward	
	7	One Request	
	8		
	9		
	A		
	B		
	C	Connect Code 2^3	
	D	2^2	
	E	2^1	
	F	2^0	
NOC-7, Bit	0	Request Bit 2^{15}	(Same as Drum Station)
	1	SBU SA/TA/Request Bit 2^{14}	
	2		
	3		
	4		
	5		
	6		
	7		
	8		
	9		
	A		
	B		
	C		
	D		
	E		
	F	2^0	

ID = identification bit
SA and TA = starting and terminating address, respectively

TABLE B-9. STAR A SBU INTERFACE NIC BIT ASSIGNMENTS

	844 Service Station	STAR-1B Service Station	STAR-100 Service Station
NIC-5, Bit 0	MEM P.E. /SCU Chan	SBU P.E. - SCU	MEM P.E. /SCU Chan
1	STA INT X Bank CTR 2^2	STA INT X Bank Count 2^2	STA INT X Bank CTR 2^2
2	2^1	2^1	2^1
3	2^0	2^0	2^0
4		MEM P.E. I/O Chan 6	
5		5	
6		4	
7		3	
8	MEM P.E. I/O Chan 1	2	MEM P.E. I/O Chan 1
9	2	1	2
A	3		3
B	4		4
C			5
D			6
E			7
F			8
NIC-6, Bit 0	MEM P.E. I/O Chan	MEM P.E. I/O Chan	MEM P.E. I/O Chan
1			
2			
3			
4			
5			
6			
7			
8	I/O Interrupt-Chan 1	I/O Int-Chan 1	I/O Interrupt-Chan 1
9	2	1	2
A	3	2	3
B	4	3	4
C		4	5
D		5	6
E		6	7
F			8
NIC-7, Bit 0	NU	NU	NU
1			
2			
3			
4			
5			
6			
7			
8			
9			
A			
B			
C			
D			
E			
F			

CTR = counter
I/O = input/output
MEM P.E. = memory parity error
NU = not used
STA INT = station interface

TABLE B-10. STAR A SBU INTERFACE NOC BIT ASSIGNMENTS

	844 Service Station	STAR-1B Service Station	STAR-100 Service Station
NOC-5, Bit 0	NU		NU
1			
2			
3			
4			
5			
6			
7			
8			
9			
A			
B		Clear Interrupt	
C		Program Clear	
D		Control Function 2^1	
E		Control Function 2^0	
F		Cont Fctn Strobe	
NOC-6, Bit 0			I/O Chan 8 Select
1		I/O Chan 6 Select	7
2		5	6
3		4	5
4	I/O Chan 4 Select	3	4
5	3	2	3
6	2	1	2
7	1		1
8			
9			
A			
B			
C	Connect Code 2^3	Connect Code 2^3	Connect Code 2^3
D	2^2	2^2	2^2
E	2^1	2^1	2^1
F	2^0	2^0	2^0
NOC-7, Bit 0		NU	
1			
2			
3			
4			
5			
6			
7			
8			
9			
A			
B	Clear Interrupt		Clear Interrupt
C	Program Clear		Program Clear
D	Control Function 2^1		Control Function 2^1
E	Control Function 2^0		Control Function 2^0
F	Cont Fctn Strobe		Cont Fctn Strobe

I/O = input/output
NU = not used
Cont Fctn = control function

TABLE B-11. 844 SBU INTERFACE NIC BIT ASSIGNMENTS

	844 Service Station	844 Storage Station
NIC-5, Bit 0	NU	NU
1		
2		
3		
4		
5		
6		
7		
8		
9		
A		
B		
C		
D		
E		
F		
NIC-6, Bit 0	Unit Busy	Unit Busy
1	Unit 1 Selected	Unit 1 Selected
2	2	2
3	3	3
4	NU	4
5		5
6		6
7		7
8		8
9		9
A	Read Error	Read Error
B	Tag BCC Error	Tag BCC Error
C	Timing Chain	Timing Chain
D	SBU Bank Counter 2^2	SBU Bank CTR 2^2
E	2^1	2^1
F	2^0	2^0
NIC-7, Bit 0	Finished	Finished
1	Unit 1 on Sector/Seek Error	Unit 1 on Sector
2	2	2
3	3	3
4	NU	4
5		5
6		6
7		7
8		8
9		9
A	Ready	Ready
B	Transfer Error	Transfer Error
C	On Cylinder	On Cylinder
D	Seek Error	Seek Error
E	Pack Unsafe	Pack Unsafe
F	Full	Full

BCC = block check code
CTR = counter
NU = not used

TABLE B-12. 844 SBU INTERFACE NOC BIT ASSIGNMENTS

	844 Service Station	844 Storage Station
NOC-5, Bit 0	NU	NU
1		
2		
3		
4		
5		
6		
7		
8		
9		
A		
B		
C		
D		
E		
F		
NOC-6, Bit 0		
1		
2		
3		
4		
5		
6		
7	Load SBU Address	Load SBU Address
8	Release	Release
9	Function	Function
A	Begin Transfer	Begin Transfer
B	NU	NU
C	Connect Code 2^3	Connect Code 2^3
D	2^2	2^2
E	2^1	2^1
F	2^0	2^0
NOC-7, Bit 0	NU	NU
1	Select Unit 1	Select Unit 1
2	2/Write Tag/Diff Sel	2/Write Tag/DIFF SELECT
3	3/Write/Clear Fault	3/Write/Clear Fault
4	Full/Control Select	4/Full/Control Select
5	Odd Bank/Sector Select	5/Odd/Sector Select
6	SBU MEM Address/Cylinder Select	6/SBU MEM ADDR 2^{14}/ Cylinder Sel
7	Address/Control Bit 8	7/SBU MEM ADDR 2^{13}/ DSU ADDR and CONT 2^8
8		8/SBU MEM ADDR 2^{12}/ DSU ADDR and CONT 2^7
9		9/SBU MEM ADDR 2^{11}/ DSU ADDR and CONT 2^6
A		SBU MEM ADDR 2^{10}/DSU ADDR and CONT 2^5
B		2^9 2^4
C		2^8 2^3
D		2^7 2^2
E		2^6 2^1
F	Bit 0	2^5 2^0

DIFF SEL = difference select
DSU ADDR and CONT = disk storage unit address and control
MEM ADDR = memory address NU = not used

TABLE B-13. MDD (841) SBU INTERFACE NIC BIT ASSIGNMENTS

	STAR-1B Service Station	Storage (Media) Station
NIC-5, Bit 0	NU	NU
1		
2		
3		
4		
5		
6		
7		
8		
9		
A		
B		
C		
D		
E		
F		
NIC-6, Bit 0	Unit 0 Selected	(Same as STAR-1B Service Station)
1	1	
2	2	
3	3	
4	4	
5	5	
6	6	
7	7	
8	8	
9	Full	
A	Read Error	
B	Tag BCC Error	
C	End of Page	
D	SBU Bank Counter 2^2	
E	2^1	
F	2^0	
NIC-7, Bit 0	Unit 0 on Sector/Seek Error	(Same as STAR-1B Service Station)
1	1	
2	2	
3	3	
4	4	
5	5	
6	6	
7	7	
8	8	
9	Busy	
A	Ready	
B	Seek Error	
C	On Cylinder	
D	Pack Unsafe	
E	MDD Error	
F	Finished	

BCC = block check code
MDD = multiple disk drive
NU = not used

TABLE B-14. MDD (841) SBU INTERFACE NOC BIT ASSIGNMENTS

	STAR-1B Service Station	Storage (Media) Station
NOC-5, Bit 0	Release/Write Tags	Release/Write Tags
1	Sector Select/Full	Sector Select/Full
2	Diff Select/Proceed	Diff Select/Proceed
3	Clear/Odd	Clear/Odd
4	Logic No. 2^2/Page Addr 2^{14}	Logic No. 2^2/Page Address 2^{14}
5	2^1/ 2^{13}	2^1/ 2^{13}
6	2^0/ 2^{12}	2^0/ 2^{12}
7	Unit Select/ 2^{11}	Unit Select/ 2^{11}
8	Addr and Cont Bit 7/Write	Addr and Cont Bit 7/Write
9	6	6
A	5/HDR ADDR 2^{10}	5/Hdr Addr 2^{10}
B	4/ 2^9	4/ 2^9
C	3/ 2^8	3/ 2^8
D	2/ 2^7	2/ 2^7
E	1/ 2^6	1/ 2^6
F	0/ 2^5	0/ 2^5
NOC-6, Bit 0		
1		
2		
3		
4		
5		
6		
7		Load SBU Address
8	Begin Transfer	Control Select
9	Clear End Page	Function
A	Load SBU Address	Begin Transfer
B	Control Select	Clear End Page
C	Function	Connect Code 2^3
D	Connect Code 2^2	2^2
E	2^1	2^1
F	2^0	2^0
NOC-7, Bit 0	NU	NU
1		
2		
3		
4		
5		
6		
7		
8		
9		
A		
B		
C		
D		
E		
F		

ADDR and CONT = address and control
DIFF = difference
NU = not used
HDR = header

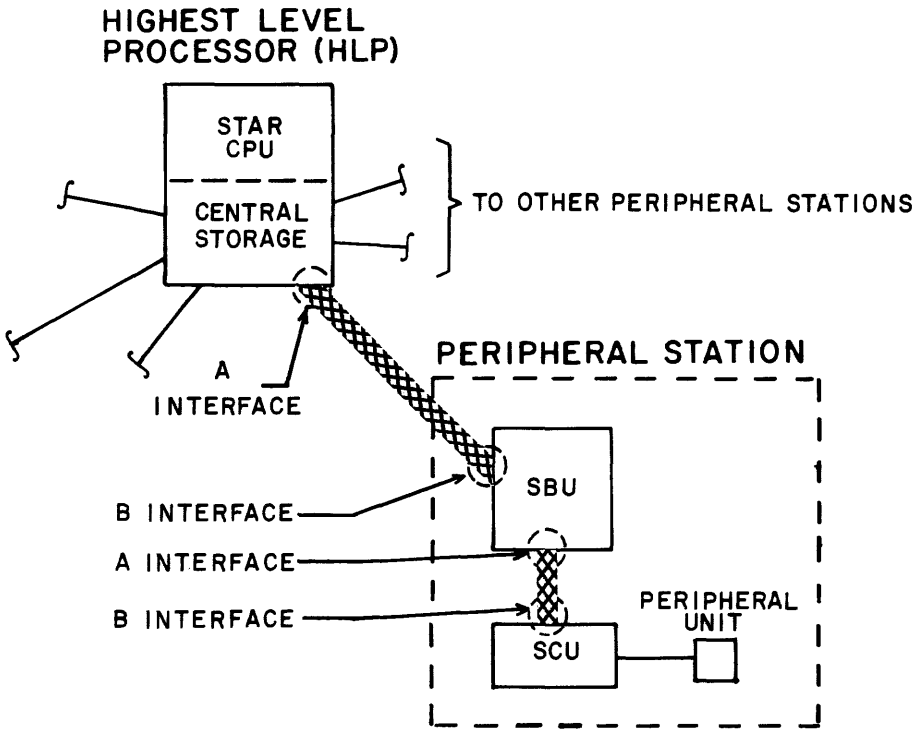
STAR DATA CHANNEL CHARACTERISTICS C

Appendix C provides a general description of the STAR data channel, its operation, and definitions of its line functions.

The characteristics of the STAR data channel apply to the data transfer and communication processes between system elements having interfaces designated as A and B (Figure C-1). In this configuration, an element having an A interface occupies a position higher in system hierarchy or closer to the center of the system than does an element having a B interface. Both elements can be stations with all communication between a B interface and the system's center passing through an A interface. Control of a data transfer is initiated by a B interface and is achieved by allowing B to have access to storage areas at a level or levels higher than B in the system. In addition to communicating with data areas, the B interface may communicate with system software at a higher level. Therefore, the B interface is considered as an active functional unit or subsystem which performs a defined function on behalf of the total system.

The STAR data channel is divided into two parts which are completely independent of each other and which operate asynchronously with respect to each other. They are as follows:

1. Data and function lines used by B to communicate with storage area within A. (Refer to Access to Storage Areas in this appendix.)
2. Control and interrupt lines used by A and B to coordinate and control their overall functions. (Refer to Communications With Software in this appendix.)



KEY:

STAR DATA CHANNEL

Notes:

SBU = station buffer unit
SCU = station control unit
CPU = central processing unit

Figure C-1. Element Identification For STAR Data Channel

ACCESS TO STORAGE AREAS

Data transfer always takes place in multiples of 32-bit words; that is, two transfers of 16 bits each are initiated by B sending a 32-bit address field to A accompanied by the required function. An address field is defined as a right-justified word address. Data then flows in either direction as determined by the specific function.

Data is always transferred as four 8-bit bytes with byte 0 being the most significant byte. Byte 0 and byte 1 are always transferred first. Four bytes are always transferred even though they are not necessarily stored in memory. (Refer to Address Field and Byte Selection in this appendix.)

Functions recognized by the CDC STAR data channel are described as follows:

1.	Read	Read one word from the specified address.
2.	Block read	Read from consecutively increasing addresses starting at the specified address until B sends a nondata function code.
3.	Write	Write the data words which follow into consecutively increasing storage addresses until a nondata function code is received. The write function allows data to be aligned to byte boundaries through the use of byte selection bits sent with the storage address.
4.	Write reverse	Write the data words which follow into consecutively decreasing addresses starting at the specified address until a nondata function code is received. The write function allows data to be aligned to byte boundaries through the use of byte selection bits sent with the storage address.
5.	End operation	This is a null operation which can be used to terminate a block read, write, or write reverse operation.
6.	Data	This is a function code used to qualify the presence of the most significant half of a data word on the interface.
7.	Null	This is a function code used to signify that the data lines contain the least significant half of an address or 32-bit data transfer.

By means of functions 1 through 7, B can use the CDC STAR data channel as if it were a direct connection. Since it is necessary to protect storage areas from the malfunction of B, this protection must be implemented by the A interface. To do this, following the receipt of a B-initiated element (address, B to A data, or EOP) during which a PEFA or IFA was generated, A inhibits operation of the data/address field; that is, it does not store it as data or use it as a new address. A remains in this inhibited state until it receives a valid error-free address element. Buffered data, received prior to the error, may or may not be stored correctly.

The lines involved in the storage access function of the interface are:

1. From B to A
   16 data lines plus 2 parity lines
   3 function lines plus 1 parity line
   2 timing lines request from B (RFB) and an accept from B (AFB)
2. From A to B
   16 data lines plus 2 parity lines
   1 parity error line
   1 illegal line
   2 timing lines request from A (RFA) and an accept from A (AFA)

Transmissions from B to A start by B sending a pulse on the RFB line followed, after a defined interval, by pulses representing the desired 1 bits on the data lines, function lines, and parity lines. When A checks the received parity and legality of the function and is ready to receive another transmission, it returns a pulse on the AFA line followed, after a defined interval, by pulses as appropriate on the parity error and illegal lines.

A transmission from A to B is started by A sending a pulse on the RFA line followed, after a defined interval, by pulses on the data lines, illegal line, and parity error lines representing the desired 1 bits. When B is ready to receive another transmission and accepts this transfer, it replys with a pulse on the AFB line.

A typical sequence involving the reading of one word from storage would be as follows:

B to A function = (read), data = (most significant half of address field)
B to A function = (null), data = (least significant half of address field)

(Pause for A to access storage)

A to B, most significant half of data word
A to B, least significant half of data word

The details of the meanings of parity errors and illegal functions are discussed in subsequent paragraphs of this section. In block operation, the address and function are only sent to A at the beginning of the block and from that point all transmission consists of data words until B ends the operation. During block transfer operations, the rate of transmission is regulated from either the A or B element by means of the timing lines.

COMMUNICATION WITH SOFTWARE

In performing its defined function, B accesses storage areas within A independently of software operations within the system. At some points, the functions of A and B must coordinate with each other (without stopping data transfers to do it) in order to signal new tasks to be performed or old tasks completed. This part of the interface transmits system functions such as master clear, autoload, start new task, and interrupt. Some of the system functions required are generated by operator action and some by direct software or hardware intervention.

The lines involved in this function of the interface are:

1. From B to A; 1 interrupt line
2. From A to B; 2 control lines, 1 control strobe line

Communication from B to A is achieved by B placing a message in a defined storage area and a pulse on the interrupt line. It is assumed that an interrupt pulse received by A will be directed to the appropriate system routine by circuits within A or higher-level equipment in the system.

A communicates with B by sending a pulse on the control strobe line followed, after a defined interval, by a pulse or pulses on either or both of the control lines. A control strobe pulse which is not followed by at least one pulse on the control lines is defined as an invalid sequence. As a result, any of three valid control codes may be present and are defined as follows:

Channel flag	A message for B has been placed in the predefined message area.
External flag	This code indicates that B should master clear and enter an autoload sequence.

Suspend	Stop using the interface for any further transfers and go into a standby mode. (Refer to Description of System Control Codes in this appendix for details.)

LINES ACROSS THE STAR DATA CHANNEL

From A to B (25 lines):	
DFA (0-15)	Data from A (16 lines)
DPFA (0,1)	Data parity from A (2 lines)
RFA	Request from A
AFA	Accept from A
PEFA	Parity error from A
IFA	Illegal from A
CSFA	Control strobe from A
CFA (0,1)	Control lines from A (2 lines)
From B to A (25 lines):	
DFB (0-15)	Data from B (16 lines)
DPFB (0,1)	Data parity from B (2 lines)
RFB	Request from B
AFB	Accept from B
FFB (0-2)	Function lines from B (3 lines)
FPFB	Function parity from B
IFB	Interrupt from B

DEFINITIONS OF LINE FUNCTIONS

A general description of line functions is given in the following paragraphs, but for a detailed description of the use of these lines to form a sequence, refer to Star Data Channel Interface Elements and Sequences in this appendix.

DATA FROM A (DFA)

DFA consists of a group of 16 pulsed lines, DFA0 to DFA15, DFA0 being the most significant. DFA lags RFA by a defined time. (Refer to Timing Rules.)

DATA PARITY FROM A (DPFA)

DPFA consists of two pulsed lines transmitted with data from A. DPFA forms odd parity with DFA, bits 0-7 and DPFA1 forms odd parity with DFA, bits 8-15.

DATA FROM B (DFB)

DFB consists of a group of 16 pulsed lines, DFB0 to DFB15, DFB0 being the most significant. DFB lags RFB by a defined time. (Refer to Timing Rules.)

DATA PARITY FROM B (DPFB)

DPFB consists of two pulsed lines transmitted with data from B. DPFB0 forms odd parity with DFB, bits 0-7 and DPFB1 forms odd parity with DFB, bits 8-15.

REQUEST FROM A (RFA)

RFA is a single pulse from A to B which precedes the data, parity error, and illegal pulses by a defined time.

REQUEST FROM B (RFB)

RFB is a single pulse from B to A which precedes the data, B functions, and B function parity pulses by a defined time. (Refer to Timing Rules.)

ACCEPT FROM A (AFA)

AFA is a single pulse from A to B which acknowledges the receipt of a request from B and its associated function and data. AFA precedes the illegal and parity error indications by a defined time. (Refer to Timing Rules.)

On transmission of AFA, A can accept a new request from B.

ACCEPT FROM B (AFB)

AFB is a single pulse from B to A which acknowledges the receipt of request from A and its associated data, parity error, and illegal indications.

On transmission of AFB, B can accept a new request from A.

PARITY ERROR FROM A (PEFA)

PEFA is a pulse from A to B which indicates one of two conditions.

1. When the pulse is associated with an accept signal from A (AFA), the data or function last received from B had incorrect parity.
2. When the pulse is associated with a request signal from A (RFA), an error has occurred within A (memory parity).

PEFA lags the accept or request from A by a defined time. (Refer to Timing Rules.)

ILLEGAL FROM A (IFA)

IFA is a pulse from A to B associated with an accept or request signal from A which indicates that A has rejected the last requested function or that an error has occurred in the sequence.

IFA lags the accept or request signal from A by a defined time. (Refer to Timing Rules.)

FUNCTION FROM B (FFB)

FFB consists of three pulsed lines transmitted with data from B. These lines are coded to identify the function of the request from B.

INTERRUPT FROM B (IFB)

IFB is a pulse from B to A which sets an external interrupt flag in A.

CONTROL STROBE FROM A (CSFA)

CSFA is a pulse from A to B which precedes the control signals from A by a defined time. (Refer to Timing Rules.)

CONTROL FROM A (CFA)

CFA consists of two pulsed lines from A to B which are coded to indicate control functions from A. CFA lags the control strobe from A by a defined time. (Refer to Timing Rules.)

STORAGE FUNCTIONS FROM B

TABLE C-1. STORAGE ACCESS FUNCTION CODES

FFB0	FFB1	FFB2	Function
0	0	0	Null
0	0	1	Read (one word)
0	1	0	Write
0	1	1	Special function
1	0	0	Data
1	0	1	Block read
1	1	1	End of operation
1	1	0	Write reverse

ADDRESS FIELD AND BYTE SELECTION

Those functions which present a storage address on the data lines use a 28-bit right-justified word address. Byte selection bits 0 through 3 (BSB0-3) are transmitted on DFB0-3 with the most significant 12 bits of the address on DFB4-15. The address is the address of a 32-bit word.

Byte selection bits are zero except during write or write reverse sequences. During write and write reverse, the first data word transferred is subject to byte selection. If BSB0 equals 1, byte 0 of the first data word is not written into storage and the corresponding byte position in storage remains unchanged.

Similarly, if BSB1, 2, and/or 3 equal 1, it indicates that bytes 1, 2, and/or 3, respectively, of the first data word are not to be written into storage and their corresponding byte positions in storage remain unchanged. Byte selection bits can be equal to 1 in any combination.

Partial word operations resulting from the use of byte selection are executed by A such that other operations at the same word address from another functional unit do not cause lost data.

Data words following the first 32-bit data word of write or write reverse sequences are unaffected by byte selection.

DESCRIPTION OF FUNCTIONS

All functions except null and data end the previous operation.

Read (001)

This function instructs A that the data bus contains the highest order 16 bits of the next address field. The lower order 16 bits of the next address field are accompanied by null. The read function notifies A that one word is to be transferred from the specified address.

Block Read (101)

This function instructs A that the data bus contains the highest order 16 bits of the next address field. The lower order 16 bits of the next address field are accompanied by null.

The block read function notifies A that several words will be transferred starting at the specified address. The read address is automatically incremented within A with each data word transferred.

Special Function (011)

This function is used by B to perform special control or diagnostic procedures when both A and B have the facilities for their implementation. Specifically, these functions are used only when B is to perform a special function other than normal input/output. Uses of the function must be determined at both A and B before implementation.

Write (010)

This function instructs A that the data bus contains the highest order 16 bits of the next address field. The lower order 16 bits of the next address field are accompanied by null.

The write address is automatically incremented within A with each data word transferred.

Data (100)

This function instructs A that the data bus contains the highest order 16 bits of a write data transfer.

Write Reverse (110)

This function instructs A that the data bus contains the highest order 16 bits of the next address field. The lower 16 bits of the next address field are accompanied by null.

The write reverse function notifies A to write the data words which follow into consecutively decreasing storage addresses at the specified address until B sends a nondata function code. The write reverse function allows data to be aligned on byte boundaries through the use of byte selection bits sent with the storage address.

The write reverse address is automatically decremented within A with each data word transferred.

End of Operation (111)

This function notifies A that the current read or write function (if any) is ended and instructs A to store the data buffers (if any) associated with that function. The end of operation function should not be sent during a 32-bit word transfer.

Null (000)

Null accompanies the second half of an address field or data word.

TABLE C-2. SYSTEM CONTROL CODES

Line 1	Line 2	Function
0	0	Invalid
0	1	Channel flag
1	0	External flag
1	1	Suspend

DESCRIPTION OF SYSTEM CONTROL CODES

System control codes (Table C-2) are transmitted to B by sending a pulse on CSFA followed, after a defined time, by pulses representing 1 bits on CFA0 and CFA1. Element B must interpret the two-bit code thus received as follows:

Invalid (00)

This code indicates that a malfunction has occurred in the control transmission and B must assume that a suspend code was intended and then generate an interrupt to inform A of the malfunction.

Channel Flag (01)

This code indicates that a message concerning normal communications from the system software has been placed in the prearranged area in storage.

External Flag (10)

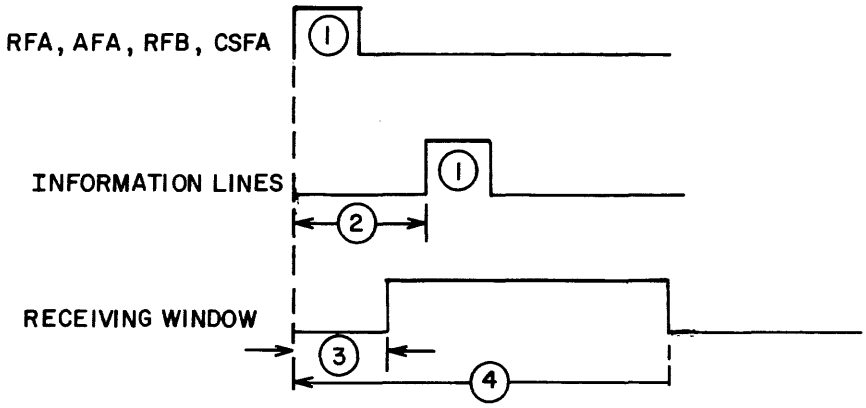
This code indicates that B should master clear and enter an autoload sequence.

Suspend (11)

This code indicates that B should immediately stop using the data channel, that is, send no more pulses. Normal operation can be resumed by A sending a channel flag code to B.

TIMING RULES

The lines RFA, RFB, AFA,and CSFA are used to strobe information pulses occurring on other lines of the interface. This mode of operation minimizes the time interval during which spurious pulses can be accepted by the receiving logic. Figure C-2 shows the specified relationship between the timing pulses and information pulses.



(1) 15 nsec MIN., 35 nsec MAX. PULSE WIDTH MEASURED AT TRANSMITTER
(2) 35 nsec MIN., 60 nsec MAX. MEASURED AT TRANSMITTER
(3) 22 nsec MAX., MEASURED AT RECEIVER
(4) 88 nsec MIN., MEASURED AT RECEIVER

Figure C-2. Timing Relationships

CHECKING PROCEDURES

The responsibility for monitoring and acting upon error situations whenever they are detected rests with B. Hence, B has to be notified of all error conditions that A detects even though A may have logged the error condition.

DATA AND ADDRESS ERRORS

If a memory parity error is detected in A, the parity error line is pulsed following the next RFA signal. If a parity error is detected in data or address from B, the parity error line is pulsed following the next AFA signal.

FUNCTION CHECKING

The function lines from B are accompanied by an odd parity indication (FPFB). If A detects a parity error in the function, the parity error from A (PEFA), and illegal from A (IFA) lines are pulsed with the next AFA signal (IFA is optional).

Control lines from A are checked for the code of 00. Since A never intentionally transmits this code, B reacts in the manner prescribed for the suspend code if the 00 code is received. Receiving this invalid code means that a pulse or pulses was or were lost and obligates B to inform A of the malfunction via the interrupt mechanism in addition to B's reaction to the suspend code.

FUNCTION VALIDITY

A function parity error or illegal function translation discovered in A is passed to B by a pulse on the illegal line (IFA) following the next AFA signal. The only lines which are continually monitored and hence susceptible to single spurious noise pulses are:

RFA
AFA
RFB
AFB
CSFA
IFB

Of these signals, IFB is the only signal that cannot be checked by other means.

STAR DATA CHANNEL SEQUENCES

This subsection contains information on the read and write sequences passing through the STAR data channel between A and B interfaces.

WRITING N 32-BIT WORDS INTO SBU OR HLP MEMORY

Control Line	Function	Data Transferred	Indication
1. RFB	Write	DFB - MS 16 bits of address	
2. AFA			IFA, PEFA (1)
3. RFB	Null	DFB - LS 16 bits of address	
4. AFA			IFA, PEFA (1)
5. RFB	Data	DFB - MS 16 bits of data	
6. AFA			IFA, PEFA (2)
7. RFB	Null	DFB - LS 16 bits of data	
8. AFA			IFA, PEFA (2)
Repeat steps 5, 6, 7, and 8 (N-1) times			
9. RFB	Read Write End of operation (ends operation) Block read Write reverse		

NOTES

PEFA (1) indicates a function or address parity error.

N=1 is allowable.

PEFA (2) indicates a function or data parity error.

READING N 32-BIT WORDS FROM SBU OR HLP MEMORY

Control Line	Function	Data Transferred	Indication
1. RFB	Block read	DFB - MS 16 bits of address	
2. AFA			IFA, PEFA (1)
3. RFB	Null	DFB - LS 16 bits of address	
4. AFA			IFA, PEFA (1)
5. RFA		DFA - MS 16 bits of data	IFA, PEFA (2)
6. AFB			
7. RFA		DFA - LS 16 bits of data	IFA, PEFA (2)

Repeat steps 6 and 7 (2N-2) times

8. RFB	RFB terminates read.	
	Continue according to function.	

NOTES

PEFA (1) indicates a function or address parity error.

PEFA (2) indicates a memory parity error in A.

N=1 is allowable.

COMMENT SHEET

MANUAL TITLE CONTROL DATA® STAR Peripheral Stations

Hardware Reference Manual

PUBLICATION NO. 60405000 REVISION B

FROM: NAME: _____

BUSINESS ADDRESS: _____

COMMENTS:

This form is not intended to be used as an order blank. Your evaluation of this manual will be welcomed by Control Data Corporation. Any errors, suggested additions or deletions, or general comments may be made below. Please include page number references and fill in publication revision level as shown by the last entry on the Record of Revision page at the front of the manual. Customer engineers are urged to use the TAR.

CUT ALONG LINE

AA3419 REV. 11/69 PRINTED IN U.S.A.

NO POSTAGE STAMP NECESSARY IF MAILED IN U. S. A.

FOLD ON DOTTED LINES AND STAPLE

STAPLE

STAPLE

FOLD

FOLD

FIRST CLASS
PERMIT NO. 8241

MINNEAPOLIS, MINN.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

POSTAGE WILL BE PAID BY

CONTROL DATA CORPORATION
Technical Publications Department
4201 North Lexington Avenue
Arden Hills, Minnesota 55112

CUT ALONG LINE

MD 220

FOLD

FOLD

CORPORATE HEADQUARTERS, 8100 34th AVE. SO., MINNEAPOLIS, MINN. 55440
SALES OFFICES AND SERVICE CENTERS IN MAJOR CITIES THROUGHOUT THE WORLD

LITHO IN U.S.A.