3720/3725 Communication Controllers

Principles of Operation

System/370, 30xx, and 4300 Processors

IBM

IBM 3720/3725 Communication Controllers

Principles of Operation

IBM System/370, 30xx, and 4300 Processors

Publication Number
GA33-0013-6

File Number
S370/30xx/4300-09

Seventh Edition (September 1986)

This is a revision of GA33-0013-5, which is now obsolete.

- Some information which is present elsewhere has been removed to avoid unnecessary duplication.
- Explicit references to IBM 3725 have been rephrased. Where there is information specific to one model of the IBM 3725/3720 families there is a note stating which model the note applies to.
- Factual changes are indicated by a vertical line to the left of the change.

The information contained in this manual is subject to change from time to time. Any such changes will be reported in subsequent revisions or Technical Newsletters.

Publications are not stocked at the addresses given below. Requests for IBM publications should be made to an IBM representative or to a local IBM branch office.

A form for readers' comments is provided at the back of this publication. If the form has been removed, comments may be addressed to either of the following:

- International Business Machines Corporation, Department 6R1LG, 180 Kost Road, Mechanicsburg, PA 17055, USA
- IBM France, Centre d'Etude et Recherches, Service 0798, 06610 La Gaude, France

IBM may use or distribute any of the information supplied in any way it believes appropriate without incurring any obligation.

PREFACE

This publication is intended to help users to write a control program or modify an IBM-supplied Network Control Program or Emulation Program. The reader should have an understanding of basic data communication and a thorough knowledge of IBM System/370 channel Operations.

A prerequisite, model-dependant, publication is the:

- **Introduction to the IBM 3725 Model 1 Communication Controller** (GA33-0010),
- **Introduction to the IBM 3725 Model 2 Communication Controller** (GA33-0021),
- **Introduction to the IBM 3720/3721 Communication Controller** (GA33-0060-0).

A related, model-dependant, publication is the:

- **IBM 3725 Communication Controller Operating Guide** (GA33-0014),
- **IBM 3720/3721 Communication Controller Operator's Guide** (GA33-0065).

This manual can be used for both the IBM 3725 and IBM 3720/3721 Communication Controllers. Any information that refers to one and not the other is noted. The Introduction Manual for each Communication Controller gives specific characteristics for the system described.

This manual is divided into six chapters and seven appendixes.

Chapter 1. Characteristics of the IBM 3725 and IBM 3720/2721

Describes the system structure, the storage scheme, the registers, the interrupt scheme, and the interrupt levels assigned to each adapter.

Chapter 2. Instruction Set

Describes each of the instructions with their mnemonics, format, and condition codes.

Chapter 3. Central Control Unit (CCU)

Describes the operation and programming requirements of the Central Control Unit.

Chapter 4. Channel Adapter

Describes the operation and programming requirements of the Channel Adapter and how it communicates with the CCU.

Chapter 5. Communication Scanner

Describes the operation and programming requirements of the Communication Scanner and how it communicates with the CCU.

Chapter 6. Token-Ring Multiplexor

Describes the operation and programming requirements of the Token-Ring Multiplexor and how it communicates with the CCU.

Appendix A. CCU External Registers

Provides summary information on the CCU registers in convenient tabular form.

Appendix B. CA Input/Output Instruction Summary Charts

Provides summary information on the channel adapter input/output instructions in convenient tabular form.

Appendix C. Communication Scanner Commands

Provides summary information on the communication scanner commands in convenient tabular form.

Appendix D. MOSS Commands

Provides summary information on the MOSS commands in convenient tabular form.

Appendix E. Redrive Logic

Provides information on the use of the redrive logic.

Appendix F. Initial Program Load

Provides information on initial program loading (IPL).

Appendix G. Branch Trace

Provides information on branch tracing operations.

CONTENTS

FIGURES

CHAPTER 1. STRUCTURE OF THE COMMUNICATION CONTROLLER

This chapter describes the structure, the storage addressing scheme, the registers, the interrupt system, and the program levels used in the controller. The user needs a thorough understanding of these facilities in order to program the controller efficiently.

Figure 1-1 shows the structure of the controller. Only one channel adapter and one communication scanner are shown.

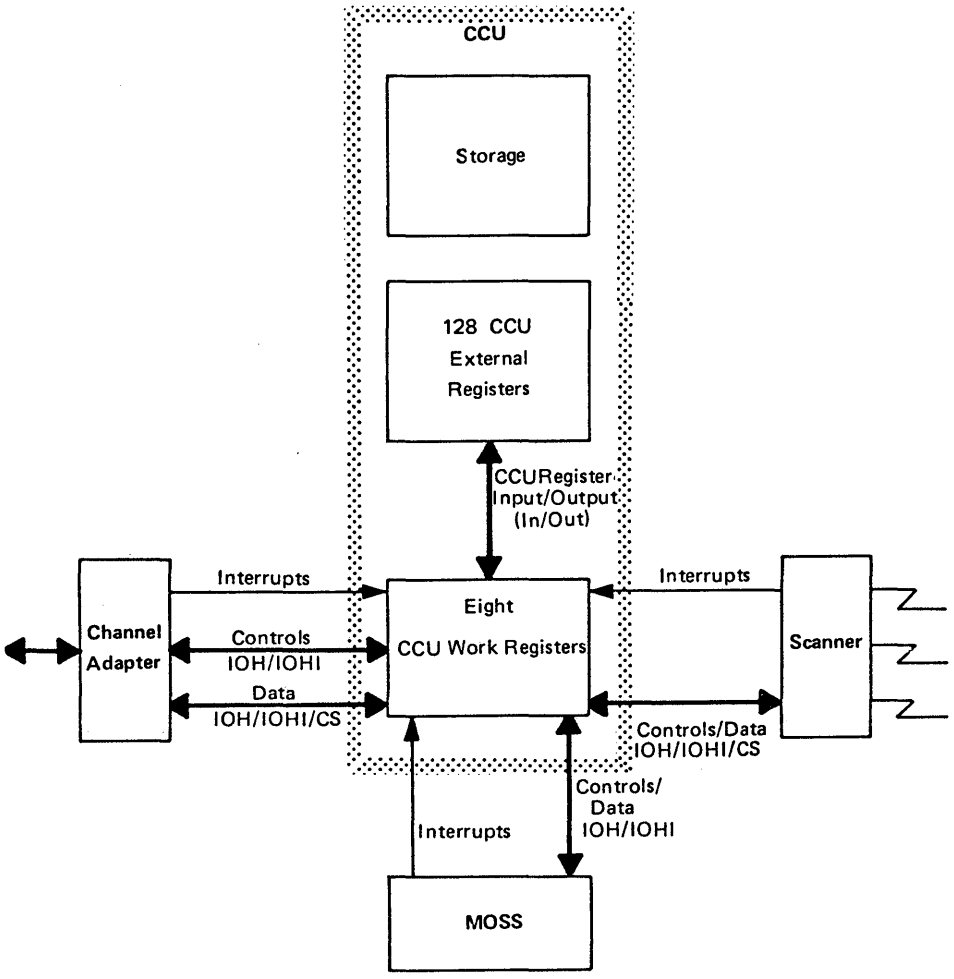


Figure 1-1. System Structure

STORAGE

Byte locations in storage are numbered consecutively starting with 0; the address of a byte is the same as its numbered position. A **group** of bytes in storage is addressed by the high-order byte of the group, the number of bytes in the group being either implied, or explicitly defined by the operation.

Storage Addressing Scheme

The storage addressing scheme uses a 22 bit address, contained in three bytes. For IBM 3725 the maximum storage size is 3 megabytes, for IBM 3720 it is 2 megabytes.

Note: In the remainder of this manual, addresses and registers capable of holding 22 bits are considered to be 3 bytes long.

The three bytes are called byte X, byte 0, and byte 1 as shown below:

Byte X	Byte 0	Byte 1
2 3 4 5 6 7	0 1 2 3 4 5 6 7	0 1 2 3 4 5 6 7

high order low order

Byte X is also called the 'extension byte'; bit 2 of this byte must be zero.

All the general registers, and all registers involved in storage addressing are structured in this way. Some of them (the High Resolution Timer/Utilization Counter, for example) use all 22 bits. All three bytes of the address form an integral part of the register regardless of the address being operated on. There are two exceptions only to this rule:

- Byte X is ignored for output (write) instructions that do not address storage. Its bits may be set to either 0 or 1.
- Byte X is set to all zeros for input (read) instructions that do not address storage.

Notes:

1. If an address is used which is less than the storage wrap point (4194303 bytes), but greater than the number of installed storage positions, an Address Exception Check occurs, and causes a level 1 interrupt or a CCU hard check. An Address Exception Check is also raised if a storage position is addressed for which the corresponding address exception key is 1.
2. Storage addressing wraps at 4194303. If an address as calculated exceeds this figure, then the true storage position addressed is the calculated address, minus 4194303. An Address Exception Check is raised if storage protection is enabled.

Storage Boundaries

Instructions and half/fullword operands must be located on integral halfword boundaries in storage; the address of a half/fullword must be a multiple of 2.

Storage addresses are expressed in binary form. Thus an integral address for a half/fullword must have the **last** binary digit equal to 0.

Storage Protection

The storage protection circuits contain three separate mechanisms:

Storage Protection by User Protection Key

Storage is divided into blocks of 2048 bytes, each block of storage being associated with a three-bit **storage key,** located in a special key storage. When an attempt is made to write in a storage location, or execute an instruction, the storage key is read from the corresponding position of the key storage. The storage keys are set up by the program.

Each user is also assigned a three-bit **protection key,** located in a hardware register. This key is active whenever the corresponding program level or cycle-stealing unit is active. The user protection keys are set up by the program.

Whenever a storage location is addressed, the storage key is read and compared with the user's protection key. Writing in the storage location or instruction fetching is allowed only if the two keys match. There are differences of operation depending on whether the storage position being addressed contains an instruction (instruction fetch phase), or data (execute phase).

If the storage location being addressed contains an **instruction,** the keys **must match,** otherwise a storage protect exception level 1 interrupt is set.

If the storage position being addressed contains **data,** the keys must satisfy one of the following conditions:

- The two keys are equal.
- The **storage key** is X'7'. The storage location is unprotected (for data only).
- The user **protection key** is X'0'. The user can operate anywhere in storage (for data only).

If none of these conditions are filled, a storage exception level 1 interrupt is set.

Notes:

1. When an EXIT instruction is executed in program levels 1 through 4, the user key is set to zero for that level.
2. When an EXIT instruction is executed in program level 5, the user key remains unchanged.
3. If an Output X'73' instruction is executed to alter the user key for the current program level, a branch instruction must follow **immediately.** This

branch instruction must cause a branch to a storage area having its storage key equal to the new user key.

4. If an Output X'73' instruction is executed which sets the user key for a block containing an instruction that would otherwise be among the next 4 to be executed, a branch instruction must be executed **immediately.** This branch instruction must cause a branch to a storage area having its storage key equal to the new user key.

5. Changing or setting the user key, storage key, read-only bit, or address exception bit by an Output X'73' instruction can only be done by a user whose user key is set to zero. If the user key is not zero, the instruction is executed, but the key is not altered.

Read-Only Protection

Each 2K block of storage also has a read-only bit. If this bit is set to 1, the block of storage may not be overwritten, but only read. Any attempt to write in a read-only protected area causes a Level 1 interrupt.

Addressing Exception Protection

Each block of storage is associated with a bit which indicates whether the block of storage is physically present or not. For 3725 this block size is 4K, for 3720 this is 256K. Any attempt to read or write in a non-existent storage location causes a Level 1 address exception interrupt.

CENTRAL CONTROL UNIT (CCU)

CCU Registers

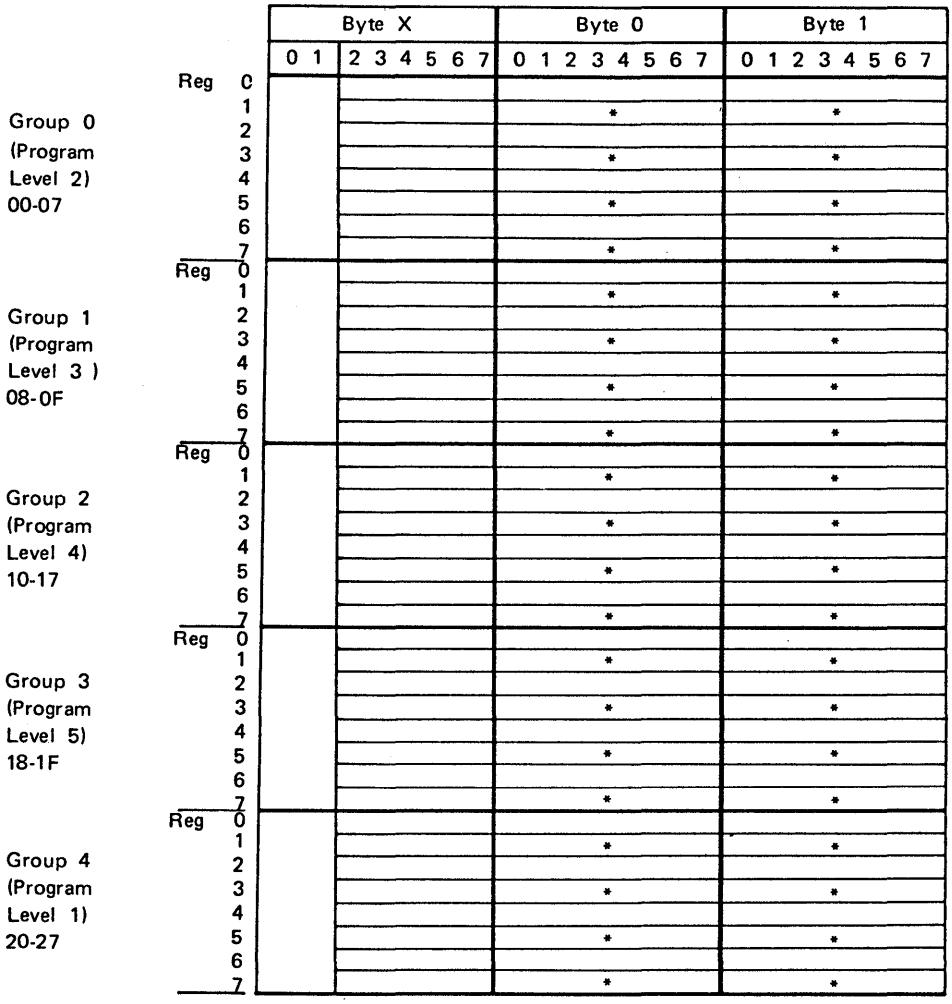
All the CCU registers are three bytes long, but the high-order byte is used only when the register in question contains an address; the high-order byte (also called byte X or the 'extension' byte) is then used for storing the high order bits of the address. The CCU has two types of register:

- Eight CCU working registers that are accessible by the program directly.
- 128 CCU external registers that are accessible to the program indirectly by means of input/output instructions. Most of these registers are located in high-speed local storage, and can be both written and read by the CCU program. Certain registers, however, are implemented in hardwar. Some of these registers are double registers; that is one register can only be written to by the program, the other can only be read out by the program. Consider register X'72' for example, where Input X'72' means 'load a general purpose register with the contents of the operator function select control'; Output X'72', however, means 'set the program display register with the contents of a general purpose register'.

The function of each CCU register is described in detail in Chapter 3.

CCU General Registers

The first 40 external registers (X'00' through X'27') are called **general registers.** They are located in local storage and are addressed as external I/O registers using the 'Input' and 'Output' instructions. The general registers have specific functions as shown in Figure 1-2 on page 1-6.



* Indicates selectable bytes of general registers

Figure 1-2. General Register Groups

The 40 general registers are divided into five groups, numbered 0 through 4, each containing eight registers. Within each group, the registers are numbered 0 through 7. Each group is assigned to a specific program level (see below under the heading **Program Levels).** When an interrupt level is exited using the EXIT instruction, and the interrupted program level re-entered, the contents of the corresponding group of eight local storage registers is transferred by hardware into the eight work registers. As instructions are executed, the eight local storage registers are updated with the contents of the work registers. This allows the control program working at one level to be interrupted by a higher priority level without the need for saving register contents. Figure 1-3 on page 1-7 below shows the relationship between the CCU working registers and the general registers.

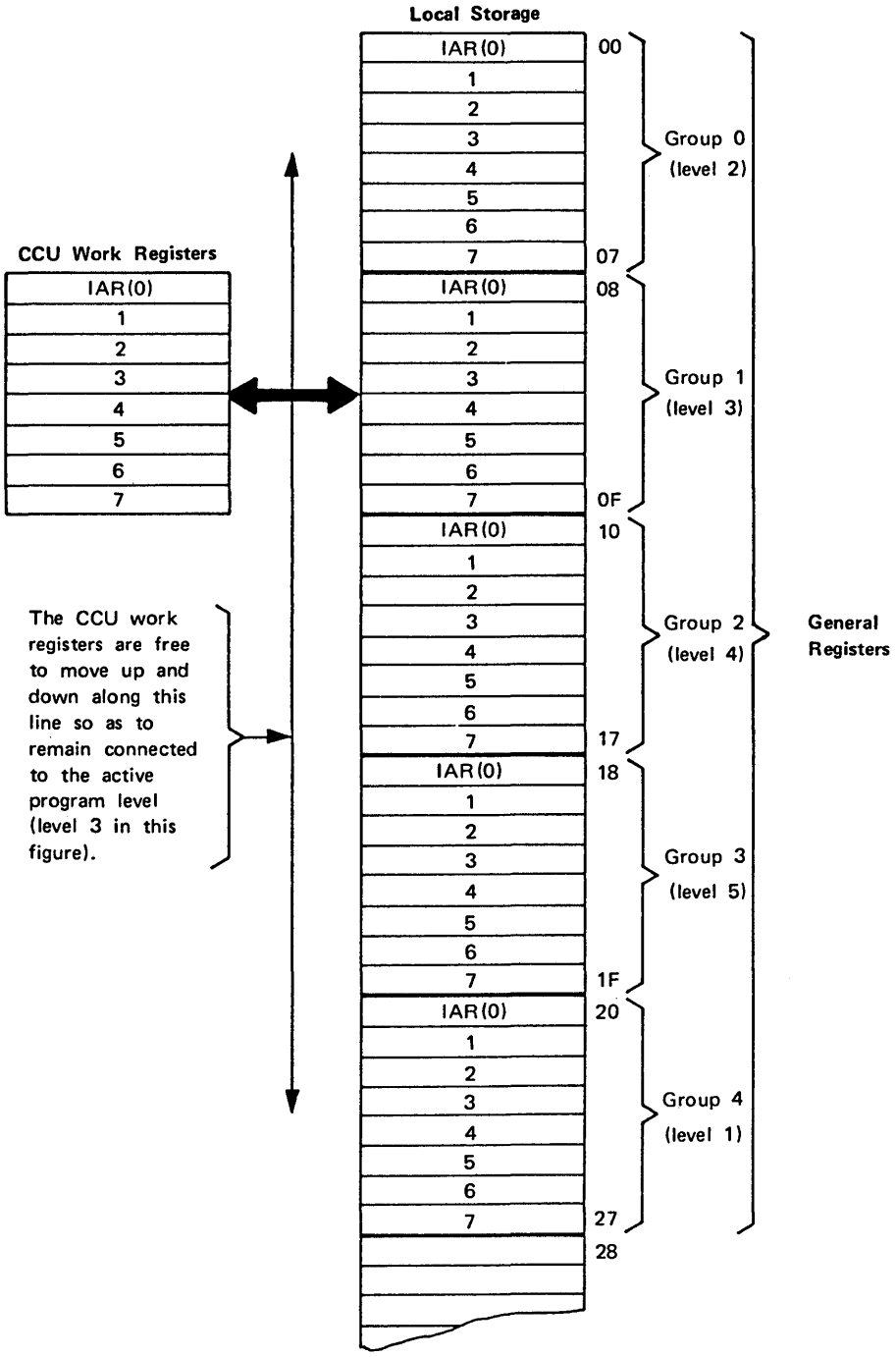


Figure 1-3. Work Registers and Local Storage

The general registers of the other (non-active) groups are considered as external registers by the active program level, which can access them in the normal way by means of CCU Register Input/Output instructions.

The first general register of each group (register 0) is used as the instruction address register (IAR) of the corresponding program level.

Condition Latches

Each program level has associated with it a pair of latches called the C and Z latches. They reflect the results of many of the instructions, and provide a means of branching on these results. Note, however, that some instructions do not change the C and Z latches; they stay set with their previous values.

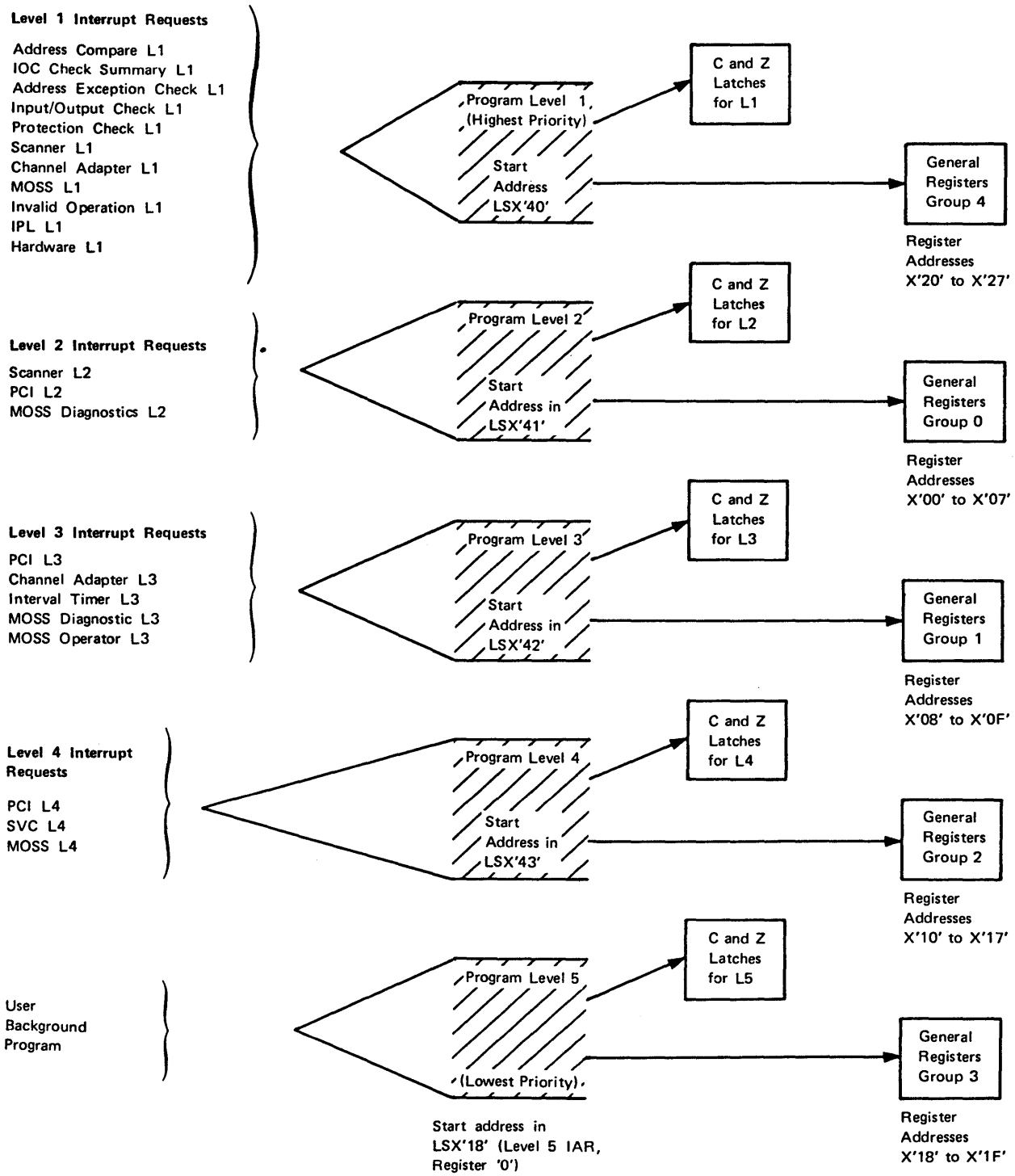
- C (carry) usually indicates a carry out of the high-order position of the register being operated on, but it can also mean 'less than' or 'not equal to' depending on the instruction.

- Z (zero) usually indicates that the result of the operation is zero, but it can also mean 'the two operands are equal'.

Program Levels

The controller hardware has five program levels. Each program level operates in a similar way to a subroutine, and is responsible for particular phases of operation.

The organization of the different program levels is shown in Figure 1-4 on page 1-9.


Level 1 Interrupt Requests
Address Compare L1
IOC Check Summary L1
Address Exception Check L1
Input/Output Check L1
Protection Check L1
Scanner L1
Channel Adapter L1
MOSS L1
Invalid Operation L1
IPL L1
Hardware L1
Program Level 1 (Highest Priority)
Start Address LSX'40'
C and Z Latches for L1
General Registers Group 4
Register Addresses X'20' to X'27'
Level 2 Interrupt Requests
Scanner L2
PCI L2
MOSS Diagnostics L2
Program Level 2
Start Address in LSX'41'
C and Z Latches for L2
General Registers Group 0
Register Addresses X'00' to X'07'
Level 3 Interrupt Requests
PCI L3
Channel Adapter L3
Interval Timer L3
MOSS Diagnostic L3
MOSS Operator L3
Program Level 3
Start Address in LSX'42'
C and Z Latches for L3
General Registers Group 1
Register Addresses X'08' to X'0F'
Level 4 Interrupt Requests
PCI L4
SVC L4
MOSS L4
Program Level 4
Start Address in LSX'43'
C and Z Latches for L4
General Registers Group 2
Register Addresses X'10' to X'17'
User Background Program
Program Level 5 (Lowest Priority)
Start address in LSX'18' (Level 5 IAR, Register '0')
C and Z Latches for L5
General Registers Group 3
Register Addresses X'18' to X'1F'

Each of the five program levels has a different priority; program level 1 has the highest priority, and program level 5 the lowest priority. Program levels 1 through 4 (referred to as the interrupt program levels) provide the connection between the hardware units and program level 5 (referred to as the background program level).

The functions assigned to the different program levels are as follows:

Background Program Level 5

This program level has the lowest priority level, and is only active when none of the other four levels requires service.

Program level 5 cannot interrupt any other program level. However, if program level 5 issues the 'exit' instruction, this generates a supervisor call interrupt at program level 4, and so allows level 5 to communicate with level 4.

Interrupt Program Level 4

The functions assigned to this level are:

- Program-controlled interrupts at level 4.
- Supervisor call (SVC) request (generated when the exit instruction is executed at level 5).
- MOSS interrupt level 4.

Interrupt Program Level 3

Level 3 is used for most of the interactions between the host processor and the channel adapter. The functions assigned to this level are:

- Program controlled interrupts at level 3.
- Channel adapter interrupts
- Interval timer interrupts at 100 ms intervals.
- MOSS diagnostic interrupts at level 3.
- MOSS operator interrupt (corresponding to the interrupt button function).

Level 3 interrupts are less critically time-dependent than those assigned to level 2.

Interrupt Program Level 2

Level 2 is used for most of the interactions between the host processor and the communication scanner. The functions assigned to this level are:

- Communication scanner interrupts.
- Program controlled interrupts at level 2.
- MOSS diagnostic interrupts at level 2.

Interrupt Program Level 1

This is the highest priority program level. It is entirely hardware driven, and is used to service 'trouble' and other unusual conditions. Conditions that can cause a level 1 interrupt are:

- Address Compare interrupts
- IOC Check Summary
- Address Exception Check
- Input/Output Check
- Protection Check
- Communication Scanner Checks
- Channel Adapter Checks
- MOSS level 1 interrupts
- Invalid Operations Check
- IPL Check
- Hardware level 1 interrupt (if in the 'bypass CCU check' mode).

Communication scanner and channel adapter checks may be masked if the CCU is in test mode.

Interrupts

The controller is an interrupt driven machine, operating in response to requests from the control program and from the hardware. Since these requests have varying degrees of urgency, a priority system is used. Each program, CCU, channel adapter, and communication scanner request is assigned a particular priority level. Any request for the use of the controller coming from either the control program or the hardware is called an **interrupt request.**

Each interrupt request is assigned to a **program level.** As we have seen above, the program levels are numbered from 1 to 5 and determine the priority structure. The priority level decreases as the program level number increases; level 1 has the highest priority, and level 5 the lowest.

The machine contains a mechanism that determines when, and in what order, interrupts may occur. If an interrupt request is allowed, the change from the current program level to the interrupting program level takes place **immediately** after completion of the current instruction. If several interrupt requests having different priorities are present at the same time, the one with the highest priority obtains the use of the controller. When a particular level is using the controller, it may be interrupted in its turn by a new interrupt request at a higher level.

When an interrupt occurs (after completion of the current instruction) instruction execution at the lower priority program level is suspended until execution at the higher level is completed. If a new interrupt request at the same level (or at a lower level) occurs, it is stacked until servicing of the current interrupt is terminated.

The controller will not allow a particular interrupt if any of the following conditions exist:

- A higher priority interrupt request is present.
- The new interrupt is at the same level as the one currently being processed.
- The interrupt request or the program level to be interrupted is masked.

At the moment that an interrupt is honored, a latch called the **interrupt entered** latch is set on. The 'interrupt entered' latch is a hardware latch that tells the controller that the associated program level has been entered. As long as this latch is on, no other interrupt requests to that level are honored. This prohibits interrupts at the same or at a lower level that could destroy essential information. The 'interrupt entered' latch is not reset when its program level is interrupted by a higher priority level. It can only be turned off by an 'exit' instruction at its own program level, or by a reset condition in the controller.

Figure 1-5 shows an example of a sequence of interrupts.

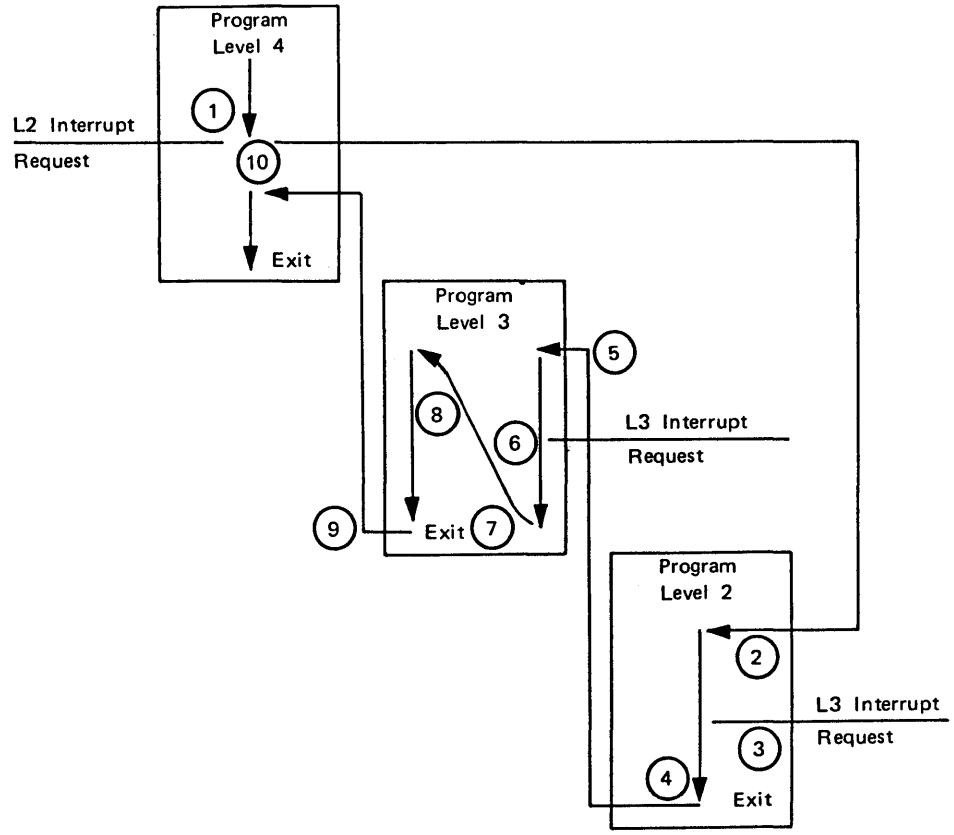


Figure 1-5. Interrupt Priority Example

Assume that the program is being executed at level 4.

1. A level 2 interrupt request occurs.

2. The controller hardware sets the 'interrupt entered' latch of level 2, and forces a branch to the start address of program level 2, which starts running.

3. During the execution of the level 2 program, a level 3 interrupt request occurs. The interrupt is not honored, because the level 2 program being executed is of higher priority. The level 3 interrupt is not lost, however, but is stored temporarily.

4. The level 2 program signals that it has ended by means of an 'exit' instruction, which resets the level 2 'interrupt entered' latch.

5. The controller now allows the next highest interrupt level to be processed. The level 3 interrupt, previously noted, sets the level 3 'interrupt entered' latch and causes entry into the level 3 program at its starting address.

6. Another interrupt occurs at level 3 during the execution of this same level. This interrupt cannot be honored, as the same level is currently being executed. It is therefore stored temporarily as before.

7. The level 3 program signals its end by means of the 'exit' instruction, which resets the level 3 'interrupt entered' latch.

8. The pending level 3 interrupt sets again the level 3 'interrupt entered' latch, causing re-entry into the level 3 program at its starting address.

9. The level 3 program signals its end by means of the 'exit' instruction, which resets the level 3 'interrupt entered' latch.

10. No further interrupts are pending. However, the level 4 'interrupt entered' latch is still on, as the level 4 program has not run to completion (it has not yet signaled its end by means of the 'exit' instruction). The level 4 program therefore continues until it has finished its current task, and then executes the 'exit' instruction. The level 4 'interrupt entered' latch is turned off, and as no further interrupts are pending, the control program returns to background level 5.

There are times when it is not desirable to interrupt a program with a higher priority request. In these cases, a mask can be set to prevent interrupts to a particular program level.

When an interrupt occurs, instruction execution at that level begins with the instruction located at its starting address. The starting addresses of the interrupt levels are contained in registers X'40' through X'43'. These assignments are shown below:

Level	Register
1	X'40'
2	X'41'
3	X'42'
4	X'43'

The instruction strings beginning at the addresses contained in these locations direct the control program to the correct routine to handle a particular interrupt.

Notes:

1. The interrupt starting addresses must be loaded into the registers at IPL time using Output instructions X'40' through X'43'.

2. Only the beginning of the interrupt routine is located at the start address; the remainder of the routine can be located anywhere in storage, and is reached by means of a branch instruction.

3. Level 5 is not entered by an interrupt; instruction execution begins at the address in the level 5 Instruction Address Register (register 0, Output X'18').

Some routines may be used by more than one program level. However, the **execution** of the routine always occurs at the priority level of the currently active program level.

When a program level has completed its interrupt servicing, it must execute an 'exit' instruction (always the last instruction of an interrupt routine). The exit instruction causes the 'interrupt entered' latch for that level to be reset, and allows control to be passed to the next higher priority program level requiring service.

Notes:

1. A program controlled interrupt (PCI) is available at levels 2, 3, and 4. Level 5 cannot generate a PCI directly, but only via the 'Exit' instruction as described in the next note.

2. When the 'Exit' instruction is executed at program level 5, the operation is modified, and a supervisor call interrupt to level 4 (SVC L4) is set. This is the only way in which program level 5 can generate an interrupt request.

CHANNEL ADAPTER (CA)

The channel adapter communicates with the CCU in three ways:

1. By means of IOH and IOHI instructions. The IOH and IOHI instructions are used to move control information between the CCU and the channel adapter registers.

   These instructions may also be used by the control program to read and write data in the Programmed Input/Output (PIO) mode. As this mode of operation is slow compared to the Adapter Input/Output (AIO) method, it should normally be used only if throughput considerations allow.

2. By means of Cycle Stealing. Cycle stealing is used for the high speed transfer of data between the CCU and the channel adapter. Once the operation has been initialized by means of IOH or IOHI instructions, the operation continues without further intervention by the program until all the data has been transferred.

3. By means of interrupts. This is the method used by the channel adapter to obtain the attention of the control program. Channel adapter interrupts are at two different levels:

   a. At level 3 for normal interrupts.

   b. At level 1 for error interrupts.

The channel adapter and its connection with the CCU are described in detail in Chapter 4.

COMMUNICATION SCANNER

The communication scanner interacts with the CCU in three ways:

1. The IOH and IOHI instructions are used to move control information between the CCU and the communication scanner registers.

2. Cycle stealing is used for the high speed transfer of control information and data between the CCU and the communication scanner. Once the operation has been initialized by means of IOH or IOHI instructions, the operation continues without further intervention by the program until all the control information and data has been transferred.

3. Interrupts are used by the communication scanner to obtain the attention of the control program. Communication scanner interrupts are at two different levels:

   a. At level 2 for normal interrupts.

   b. At level 1 for error interrupts.

The communication scanner and its connection with the CCU are described in detail in Chapter 5.

TOKEN-RING MULTIPLEXOR

The token-ring multiplexor (TRM) interacts with the CCU in three ways:

1. IOH and IOHI instructions are used to move control information between the CCU and the TRM's RAM and registers.

2. By means of Direct Memory Access (DMA) transfers controlled by the TRM. Once

the operation has been initialized by means of IOH and IOHI instructions, the operation continues without further intervention by the program until all the control information and data has been transferred.

3. Interrupts are used between the control program and the TRM and can be read or written by the program in CCU.

The TRM and its connections to the CCU are described in detail in Chapter 6.

MAINTENANCE AND OPERATOR SUBSYSTEM (MOSS)

The MOSS interacts with the CCU and main storage in several ways:

1. The MOSS can read and write anywhere in main storage. This facility must be used with great care.

2. The MOSS communicates with the CCU using the top 2K of main storage which is reserved for this purpose. Two 32-byte areas of this high storage are called the 'In Mailbox' and the 'Out Mailbox'. Most of the normal communications between the CCU and the MOSS take place via these mailboxes. When a mailbox has been filled, the CCU (or the MOSS) requests an interrupt to inform the MOSS (or the CCU) that the mailbox is available. See (3) below for details of the interrupt.

3. The following interrupts are used:

   a. Level 4 for service interrupts. This is the procedure used by the MOSS to inform the CCU that it has filled a mailbox. A similar procedure is used by the CCU to inform the MOSS that it has filled a mailbox.

   b. Level 3 which is used as a console interrupt key.

   c. Levels 2 and 3 for diagnostic interrupts.

   d. Level 1 for error interrupts.

4. The CCU Register Input and CCU Register Output instructions are used to move control information between the CCU and the MOSS registers.

CHAPTER 2. INSTRUCTION SET

The Communication Controller makes use of a set of instructions that can be used to tailor the control program to meet the specific needs of the data communication system. For the 3725 there are 53 instructions, for the 3720 there are 54.

This chapter gives the general instruction formats, followed by a detailed description of each instruction.

Important Note: Readers familiar with the instruction set of the IBM 3704/5 Communications Controllers should note that there are minor differences between the 3704/5 and the 372x in the following instructions:

Branch and Link	Load Address
CCU Register Input	Load Halfword
CCU Register Output	Store
Exit	Store Character
Insert Character	Store Character and Count
Insert Character and Count	Store Halfword
Load	

The 3720 has an additional instruction, No Operation. It has no mnemonic and no format, simply X'0060'.

Programming Note

Modifying an instruction during program execution is not advisable. If instruction modification is done, at least four CCU cycles must be allowed before executing the modified instruction. Failure to observe this rule may cause unpredictable results.

INSTRUCTION FORMAT

Instructions have a length of one half word, with the exception of the 'Load Address', 'Branch and Link', and 'Adapter Input/Output Immediate' instructions which have a length of two halfwords. There are eight basic instruction formats:

- Register to Immediate Operand Instructions (RI)
- Register to Register Instructions (RR)
- Register to Storage Instructions (RS)
- Register to Storage with Additional Operations Instructions (RSA)
- Branch Operations (RT)
- Register to Immediate Address Instructions (RA)
- Exit Instruction (EXIT)
- Input/Output Instructions (RE)

INSTRUCTION SET SUMMARY

Figure 2-1 on page 2-2 shows the basic mnemonic names and assembler operand field designations for each instruction. For the explanation of the terms occurring in the 'Operand Field Format' column, refer to the notes in the next section for each different type of instruction.

Instruction	Format Code	Mnemonic	Operand Field Format
Adapter Input/Output	RR	IOH	R1,R2
Adapter Input/Output Immediate	RI	IOHI	R,I
Add Character Register	RR	ACR	R1(N1),R2(N2)
Add Halfword Register	RR	AHR	R1,R2
Add Register	RR	AR	R1,R2
Add Register Immediate	RI	ARI	R(N),I
AND Character Register	RR	NCR	R1(N1),R2(N2)
AND Halfword Register	RR	NHR	R1,R2
AND Register	RR	NR	R1,R2
AND Register Immediate	RI	NRI	R(N),I
Branch	RT	B	T
Branch and Link	RA	BAL	R,A
Branch and Link Register	RR	BALR	R1,R2
Branch on Bit	RT	BB	R(N,M),T
Branch on Count	RT	BCT	R(N),T
Branch on C Latch	RT	BCL	T
Branch on Z Latch	RT	BZL	T
CCU Register Input	RE	IN	R,E
CCU Register Output	RE	OUT	R,E
Compare Character Register	RR	CCR	R1(N1),R2(N2)
Compare Halfword Register	RR	CHR	R1,R2
Compare Register	RR	CR	R1,R2
Compare Register Immediate	RI	CRI	R(N),I
Exclusive OR Character Register	RR	XCR	R1(N1),R2(N2)
Exclusive OR Halfword Register	RR	XHR	R1,R2
Exclusive OR Register	RR	XR	R1,R2
Exclusive OR Register Immediate	RI	XRI	R(N),I
Exit	EXIT	EXIT	-
Insert Character	RS	IC	R(N),D(B)
Insert Character and Count	RSA	ICT	R(N),B
Load	RS	L	R,D(B)
Load Address	RA	LA	R,A
Load Character Register	RR	LCR	R1(N1),R2(N2)
Load Character with Offset Reg	RR	LCOR	R1(N1),R2(N2)
Load Halfword	RS	LH	R,D(B)
Load Halfword Register	RR	LHR	R1,R2
Load Halfword with Offset Reg	RR	LHOR	R1,R2
Load Register	RR	LR	R1,R2
Load Register Immediate	RI	LRI	R(N),I
Load with Offset Register	RR	LOR	R1,R2
OR Character Register	RR	OCR	R1(N1),R2(N2)
OR Halfword Register	RR	OHR	R1,R2
OR Register	RR	OR	R1,R2
OR Register Immediate	RI	ORI	R(N),I
Store	RS	ST	R,D(B)
Store Character	RS	STC	R(N),D(B)
Store Character and Count	RSA	STCT	R(N),B
Store Halfword	RS	STH	R,D(B)
Subtract Character Register	RR	SCR	R1(N1),R2(N2)
Subtract Halfword Register	RR	SHR	R1,R2
Subtract Register	RR	SR	R1,R2
Subtract Register Immediate	RI	SRI	R(N),I
Test Register Under Mask	RI	TRM	R(N),I

Figure 2-1. Instruction Set Summary

INSTRUCTION SET BY TYPE OF INSTRUCTION

Figure 2-2 below shows the operation code bit structure and the operand fields for each instruction. In this section, the instructions are grouped by type of instruction.

				Format																
Name	Instruction	CZ	N	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
LRI	Load Reg Immediate	Y	1	1	0	0	0	0	R		N	Immediate Data								(1,2)
ARI	Add Reg Immediate	Y	1	1	0	0	1	0	R		N	Immediate Data								(1,2)
SRI	Subt Reg Immediate	Y	1	1	0	1	0	0	R		N	Immediate Data								(1,2)
CRI	Comp Reg Immediate	Y	1	1	0	1	1	0	R		N	Immediate Data								(1,2)
XRI	XOR Reg Immediate	Y	1	1	1	0	0	0	R		N	Immediate Data								(1,2)
ORI	OR Reg Immediate	Y	1	1	1	0	1	0	R		N	Immediate Data								(1,2)
NRI	AND Reg Immediate	Y	1	1	1	1	0	0	R		N	Immediate Data								(1,2)
TRM	Test R under Mask	Y	1	1	1	1	1	0	R		N	Mask Bits								(1,2)
BALR	Branch & Link Reg	N	4	0	R2			0	R1			0	1	0	0	0	0	0	0	(3)
LHR	Load Halfword Reg	Y	1	0	R2			0	R1			1	0	0	0	0	0	0	0	(3)
LR	Load Register	Y	1	0	R2			0	R1			1	0	0	0	1	0	0	0	(3)
AHR	Add Halfword Reg	Y	1	0	R2			0	R1			1	0	0	1	0	0	0	0	(3)
AR	Add Register	Y	1	0	R2			0	R1			1	0	0	1	1	0	0	0	(3)
SHR	Subt Halfword Reg	Y	1	0	R2			0	R1			1	0	1	0	0	0	0	0	(3)
SR	Subtract Register	Y	1	0	R2			0	R1			1	0	1	0	1	0	0	0	(3)
CHR	Comp Halfword Reg	Y	1	0	R2			0	R1			1	0	1	1	0	0	0	0	(3)
CR	Compare Register	Y	1	0	R2			0	R1			1	0	1	1	1	0	0	0	(3)
XHR	XOR Halfword Reg	Y	1	0	R2			0	R1			1	1	0	0	0	0	0	0	(3)
XR	XOR Register	Y	1	0	R2			0	R1			1	1	0	0	1	0	0	0	(3)
OHR	OR Halfword Reg	Y	1	0	R2			0	R1			1	1	0	1	0	0	0	0	(3)
OR	OR Register	Y	1	0	R2			0	R1			1	1	0	1	1	0	0	0	(3)
NHR	AND Halfword Reg	Y	1	0	R2			0	R1			1	1	1	0	0	0	0	0	(3)
NR	AND Register	Y	1	0	R2			0	R1			1	1	1	0	1	0	0	0	(3)
LHOR	Load HW w. Offset	Y	1	0	R2			0	R1			1	1	1	1	0	0	0	0	(3)
LOR	Load w. Offset	Y	1	0	R2			0	R1			1	1	1	1	1	0	0	0	(3)
LCR	Load Character Reg	Y	1	0	R2		N	0	R1		N	0	0	0	0	1	0	0	0	(4,5)
ACR	Add Character Reg	Y	1	0	R2		N	0	R1		N	0	0	0	1	1	0	0	0	(4,5)
SCR	Subt Character Reg	Y	1	0	R2		N	0	R1		N	0	0	1	0	1	0	0	0	(4,5)
CCR	Comp Character Reg	Y	1	0	R2		N	0	R1		N	0	0	1	1	1	0	0	0	(4,5)
XCR	XOR Character Reg	Y	1	0	R2		N	0	R1		N	0	1	0	0	1	0	0	0	(4,5)
OCR	OR Character Reg	Y	1	0	R2		N	0	R1		N	0	1	0	1	1	0	0	0	(4,5)
NCR	Add Character Reg	Y	1	0	R2		N	0	R1		N	0	1	1	0	1	0	0	0	(4,5)
LCOR	Load Char w Offset	Y	1	0	R2		N	0	R1		N	0	1	1	1	1	0	0	0	(4,5)
L	Load	Y	4	0	Base Reg			0	R			0	Displacement					1	0	(6,7)
ST	Store	N	2	0	Base Reg			0	R			1	Displacement					1	0	(6,7)
LH	Load Halfword	Y	3	0	Base Reg			0	R			0	Displacement						1	(6,7)
STH	Store Halfword	N	1	0	Base Reg			0	R			1	Displacement						1	(6,7)
IC	Insert Character	Y	3	0	Base Reg			1	R		N	0	Displacement							(1,2,7)
STC	Store Character	N	1	0	Base Reg			1	R		N	1	Displacement							(1,2,7)
ICT	Insert Char & Ct	N	4	0	Base Reg			0	R		N	0	0	0	1	0	0	0	0	(1,2,8)
STCT	Store Char & Ct	N	3	0	Base Reg			0	R		N	0	0	1	1	0	0	0	0	(1,2,8)
B	Branch	N	4	1	0	1	0	1	Displacement											(9)
BZL	Branch on Z latch	N	2	1	0	0	0	1	Displacement											(9)
BCL	Branch on C latch	N	2	1	0	0	1	1	Displacement											(9)
BCT	Branch on Count	N	3	1	0	1	1	1	R		N	1	Displacement							(1,2,9)
BB	Branch on Bit	N	2	1	1	M	M	1	R		N	M	Displacement							(1,2,9,10)
BAL	Branch and Link	N	4	1	0	1	1	1	R			0	1	Addr Byte Ext						Addr Bytes 0 and 1 (6,11)
LA	Load Address	N	1	1	0	1	1	1	R			0	0	Addr Byte Ext						Addr Bytes 0 and 1 (6,11)
EXIT	Exit	N	9	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	
IN	CCU Reg Input	N	1	0	E			0	R			E				1	1	0	0	(6,12,13)
OUT	CCU Reg Output	N	1	0	E			0	R			E				0	1	0	0	(6,12,13)
IOH	Adapter I/O	Y	7	0	R2			0	R1			0	1	0	1	0	0	0	0	(3,12,14)
IOHI	Adapter I/O Immed	Y	6	0	0	0	0	0	R			0	1	1	1	0	0	0	0	Depends on adapter (3,12,15)

The column C,Z indicates whether the C and Z registers are changed by the execution of the instruction (Y = yes; N = no).

The column N indicates the minimum number of 200ns CCU cycles required to execute the instruction; in practice there may be more. In the case of the IOH and IOH instructions, the number of execution cycles is increased by the adapter response time.

Figure 2-2. Instruction Set by Type of Instruction

1. The R field addresses the general registers. As the R field is only two bits long, these bits form the two high order bits of the register address. The low order bit of the address is created by hardware, and is always 1. This means that only odd numbered general registers (1, 3, 5, 7) can be addressed.

2. The bit marked N is used to select one (or sometimes both) of the bytes of the general register selected by the associated R field.

3. The R1 and R2 fields address the general registers. As the R1 and R2 fields are three bits long, these bits can take any value from 0 to 7, and all three bytes of the register are used in the operation.

4. The R1 and R2 fields address the general registers. As the R1 and R2 fields are only two bits long, these bits form the two high order bits of the register address. The low order bit of the address is created by hardware, and is always 1. This means that only odd numbered general registers (1, 3, 5, 7) can be addressed.

5. The bit marked N is used to select one (or sometimes both) of the bytes of the general register selected by the associated R1/R2 field.

6. The R field addresses the general registers. As the R field is three bits long, these bits can take any value from 0 to 7, and all three bytes of the register are used in the operation.

7. The effective storage address is formed by adding the displacement to the contents of the base register selected by bits 1-3.

8. The contents of the base register specified are incremented by 1 after storage access.

9. The displacement field is added to the address of the next sequential instruction (contained in general register 0) to form the branch address.

10. The three bits of the M(ask) field specify the bit to be tested.

11. The 20 bits contained in the extension byte and in bytes 0 and 1 form an address. In the case of the branch and link instruction, these 20 bits form the branch address. In the case of the load address instruction, the 20 bits are treated as immediate data and loaded into the register specified by R.

12. The IN and OUT instructions can only address the **CCU** external registers. The IOH and IOHI instructions can only address the **adapter** external registers. See below for lists of these registers.

13. The E field consists of 7 bits and addresses one of the 128 external CCU registers.

14. The contents of R2 includes the address of the adapter external register.

15. The second half word contains the address of the adapter external register.

INSTRUCTION SET DETAILED BIT STRUCTURE

In this section, the instructions are grouped logically, that is, it starts with all the LOAD instructions, and then continues with all the STORE instructions, etc.

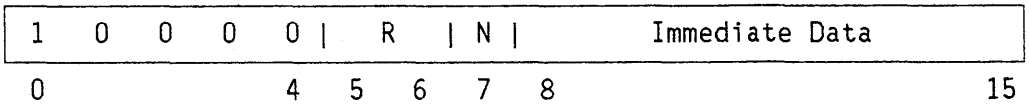
Each table is followed by a detailed description of the individual instructions of that type.

Load Instructions

			Format														
Name	Instruction	Type	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
LRI	Load Reg Immediate	RI	1	0	0	0	0	R		N	0	Immediate Data						
LR	Load Register	RR	0	R2			0	R1			1	0	0	0	1	0	0	0
LOR	Load Reg w. Offset	RR	0	R2			0	R1			1	1	1	1	1	0	0	0
LHR	Load Halfword Reg	RR	0	R2			0	R1			1	0	0	0	0	0	0	0
LHOR	Load H/W w. Offset	RR	0	R2			0	R1			1	1	1	1	0	0	0	0
LCR	Load Character Reg	RR	0	R2		N	0	R1		N	0	0	0	0	1	0	0	0
LCOR	Load Char w Offset	RR	0	R2		N	0	R1		N	0	1	1	1	1	0	0	0
L	Load	RS	0	Base Reg			0	R			0	Displacement					1	0
LH	Load Halfword	RS	0	Base Reg			0	R			0	Displacement						1
IC	Insert Character	RS	0	Base Reg			1	R		N	0	Displacement						
ICT	Insert Char & Ct	RSA	0	Base Reg			0	R		N	0	0	0	1	0	0	0	0
LA	Load Address	RA	1	0	1	1	1	R			0	0	0	0	Ad Byte Ext			
			Addr Byte 0									Addr Byte 1						

Load Register Immediate

LRI R(N),I RI



The second operand (Immediate Data field) is loaded into the first operand (byte 0 if N = 0, or byte 1 if N = 1, of the register specified by R). The non-selected bytes of the register remain unchanged. The two bits of R form the two high-order bits of the register address; the low order bit is forced to 1 by hardware. For this reason, R is always an odd-numbered register (1, 3, 5, 7).

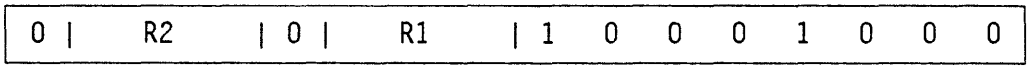
Resulting Condition Latches:

C: the result in the selected byte of R ≠ 0.

Z: the result in the selected byte of R = 0

Load Register

LR R1,R2 RR



The second operand (bytes 0 and 1, **and** byte X of R2) is loaded into the first operand (bytes 0 and 1, **and** byte X of the register specified by R1). The second operand remains unchanged, and the condition latches are set according to the result in the first operand.

Resulting Condition Latches:

C: the result in R1 ≠ 0.

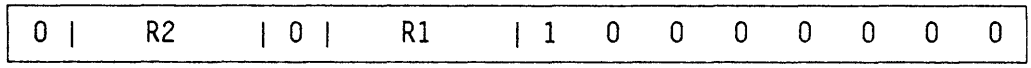
Z: the result in R1 = 0

Programming Note

If register 0 is specified as R1, a branch results to the address formed in register 0, and the condition latches remain unchanged.

Load Halfword Register

LHR R1,R2 RR



The second operand (bytes 0 and 1, but **not** byte X of R2) is loaded into the first operand (bytes 0 and 1, but **not** byte X) of the register specified by R1). The second operand remains unchanged, and the condition latches are set according to the result in the first operand.

Resulting Condition Latches:

C: the result in bytes 0 and 1 of R ≠ 0.

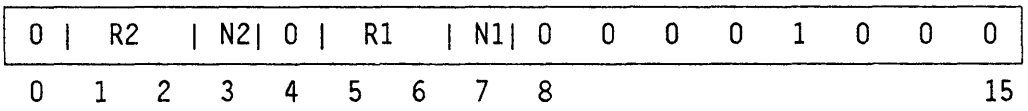
Z: the result in bytes 0 and 1 of R = 0

Programming Note

If register 0 is specified as R1, a branch results to the address formed in register 0, and the condition latches remain unchanged.

Load Character Register

LCR R1(N1),R2(N2) RR



The second operand (R2, byte 0 if N2 = 0, or byte 1 if N2 = 1) is loaded into the first operand (R1, byte 0 if N1 = 0, or byte 1 if N1 = 1). The non-selected bytes of R1 remain unchanged. The two bits of R2 and R1 form the two high-order bits of the register addresses; the low order bits are forced to 1 by hardware. For this reason, R2 and R1 are always odd-numbered registers (1, 3, 5, 7).

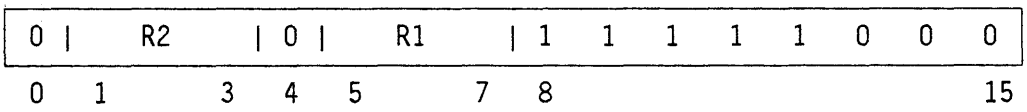
Resulting Condition Latches:

C: the selected byte of R1 contains an even number of 1 bits, or is zero.

Z: the selected byte of R1 = 0

Load Register with Offset

LOR R1,R2 RR



The second operand (bytes 0 and 1, **and** byte X of R2) is shifted right one bit position, and the result loaded into the first operand (bytes 0 and 1, **and** byte X, of the register specified by R1). A 0 bit is inserted into the high-order bit position of R1, byte X.

Resulting Condition Latches:

C: a 1 bit was shifted out of byte 1, bit 7 of R2.

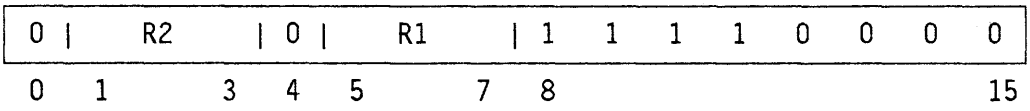
Z: the result in R1 = 0.

Programming Note

If register 0 is specified as R1, a branch results to the address formed in register 0, and the condition latches remain unchanged.

Load Halfword Register with Offset

LHOR R1,R2 RR



The second operand (bytes 0 and 1, but **not** byte X of R2) is shifted right one bit position, and the result loaded into the first operand (bytes 0 and 1, but **not** byte X, of the register specified by R1). A 0 bit is inserted into the high-order bit position of R1, byte 0.

Resulting Condition Latches:

C: a 1 bit was shifted out of byte 1, bit 7 of R2.

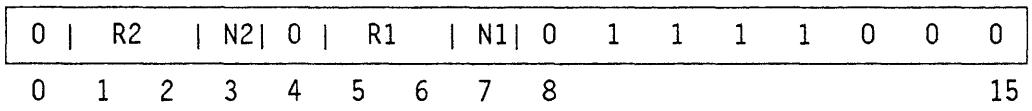
Z: the result in bytes 0 and 1 of R1 = 0.

Programming Note

If register 0 is specified as R1, a branch results to the address formed in register 0, and the condition latches remain unchanged.

Load Character Register with Offset

LCOR R1(N1),R2(N2) RR



The second operand (R2, byte 0 if N2 = 0, or byte 1 if N2 = 1) is shifted right one bit position, and the result is loaded into the first operand (R1, byte 0 if N1 = 0, or byte 1 if N1 = 1). A 0 bit is inserted into the high-order bit position of the selected byte of R1. The non-selected bytes of R1 remain unchanged. The two bits of R1 and R2 form the two high-order bits of the register addresses; the low order bits are forced to 1 by hardware. For this reason, R1 and R2 are always odd-numbered registers (1, 3, 5, 7).

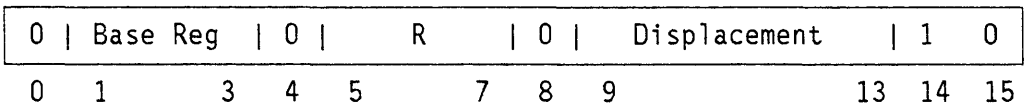
Resulting Condition Latches:

C: a 1 bit was shifted out of bit 7 of the selected byte of R2.

Z: the result in the selected byte of R1 = 0.

Load

L R,D(B) RS



The load instruction loads the data (the second operand) from a four-byte field in storage into R, the first operand register. The four byte field containing the second operand must be on a halfword boundary. As the general registers are only 3 bytes long (bytes X, 0, and 1), only the three low-order bytes (22 bits) of the storage location are used.

The storage address is formed by adding the displacement value to the contents of the base register specified by B. The displacement field allows for a displacement of between 0 and 124 bytes in multiples of 4 (32 fullwords).

Resulting Condition Latches:

C: the result in R ≠ 0

Z: the result in R = 0.

Programming Notes

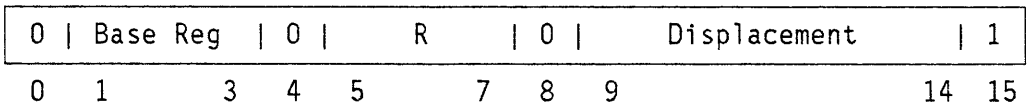
1. The low order-bit of the address is ignored since storage is addressed on halfword boundaries with this instruction.

2. If register 0 (instruction address register) is specified in the R field, the instruction causes an unconditional branch to the address loaded into register 0; the condition latches are unchanged.

3. If the base register specified is R0, the contents of R0 are **not** used as the address. Instead, a program settable address located in CCU external register X'46' is used as a base address instead of the contents of register 0. This permits direct addressing of the 32 fullwords starting at the address contained in register X'46' without having to load a base register. After program loading, the contents of register X'46' are unpredictable and must be initialized before use.

Load Halfword

LH R,D(B) RS



The load halfword instruction loads the data (the second operand) from a halfword field in storage into bytes 0 and 1 of R, the first operand register. Byte X of R is set to 0.

The storage address is formed by adding the displacement value to the contents of the base register specified by B. The displacement field allows for a displacement of between 0 and 126 bytes in multiples of 2 (64 halfwords).

Resulting Condition Latches:

C: the result in bytes 0 and 1 of R ≠ 0

Z: the result in bytes 0 and 1 of R = 0.

Programming Notes

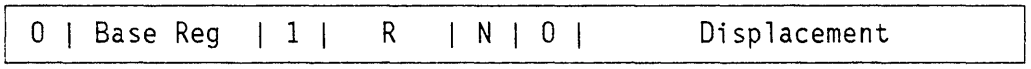
1. The low order-bit of the address is ignored since storage is addressed on halfword boundaries with this instruction.

2. If register 0 (instruction address register) is specified in the R field, the instruction causes an unconditional branch to the address loaded into register 0; the condition latches are unchanged.

3. If the base register specified is R0, the contents of R0 are **not** used as the address. Instead, a program settable address located in CCU external register X'45' is used as a base address instead of the contents of register 0. This permits direct addressing of the 64 halfwords starting at the address contained in register X'45' without having to load a base register. After program loading, the contents of register X'45' are unpredictable and must be initialized before use.

Insert Character

IC R(N),D(B) RS



The insert character instruction loads the 8-bit character at the second operand address into byte 0 (N = 0) or byte 1 (N = 1) of the register specified by R. The remaining bits of the register are unchanged.

The storage address is formed by adding the displacement value to the contents of the base register specified by B. The displacement field allows for a displacement of between 0 and 127 bytes. The two bits of R form the two high-order bits of the register address; the low order bits are forced to 1 by hardware. For this reason, R is always an odd-numbered register (1, 3, 5, 7).

Resulting Condition Latches:

C: the selected byte of R contains an even number of 1 bits, or is zero.

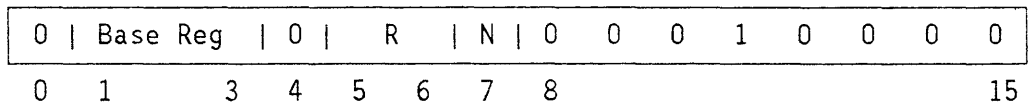
Z: the selected byte of R is equal to 0.

Programming Note

If the base register specified is R0, the contents of R0 are **not** used as the address. Instead, a program settable address located in CCU external register X'44' is used as a base address instead of the contents of register 0. This permits direct addressing of the 128 bytes starting at the address contained in register X'44' without having to load a base register. After program loading, the contents of register X'44' are unpredictable and must be initialized before use.

Insert Character and Count

ICT R(N),B RS



The eight bit character at the second operand address (contained in the base register designated by the B field) is loaded into byte 0 (N = 0) or byte 1 (N = 1) of the register specified by R. After the storage address has been obtained from the base register B, the contents of the base register is incremented by 1. After execution of the instruction, the base register normally contains an address 1 byte greater than before execution (see however programming note 2 below). The two bits of R form the two high-order bits of the register address; the low order bit is forced to 1 by hardware. For this reason, R is always an odd-numbered register (1, 3, 5, 7).

Resulting Condition Latches: unchanged

Programming Notes

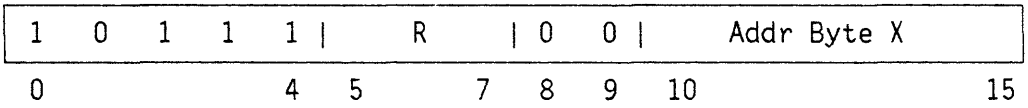
1. If register 0 is specified in the B field, it causes an invalid operation check.

2. If R and B specify the same (odd) register, its contents are incremented by 1 **before** the character is inserted into the selected byte of the register.

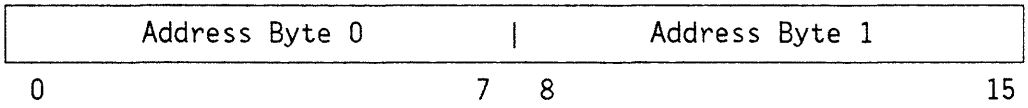
Load Address

LA R,A RA

First halfword



Second halfword



The load address instruction is a 4-byte instruction. The second operand (address field) is treated as an immediate operand, and is loaded into the first operand specified by R.

Resulting Condition Latches: unchanged

Programming Note

If register 0 is specified by R, a branch results to the address contained in the address field.

Store Instructions

Name	Instruction	Type	Format 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
ST	Store	RS	0 Base Reg 0 R 1 Displacement 1 0
STH	Store Halfword	RS	0 Base Reg 0 R 1 Displacement 1
STC	Store Character	RS	0 Base Reg 1 R N 1 Displacement
STCT	Store Char & Ct	RSA	0 Base Reg 0 R N 0 0 1 1 0 0 0 0

Store

ST R,D(B) RS

0	Base Reg	0	R	1	Displacement	1	0
0	1 3	4	5 7	8	9 13	14	15

The store instruction stores the contents of the first operand (the register specified by R) into the second operand in storage. The address of the second operand must be on a halfword boundary. As the general registers are only 3 bytes long (bytes X, 0, and 1), only the three low-order bytes (22 bits) of the storage location are affected.

The storage address is formed by adding the displacement value to the contents of the base register specified by B. The displacement field allows for a displacement of between 0 and 124 bytes in multiples of 4 (32 fullwords).

Resulting Condition Latches: unchanged

Programming Notes

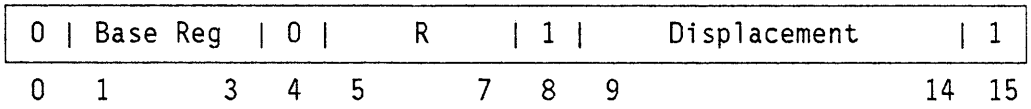
1. The low order-bit of the address is ignored since storage is addressed on halfword boundaries with this instruction.

2. If the base register specified is R0, the contents of R0 are **not** used as the address. Instead, a program settable address located in CCU external register X'46' is used as a base address instead of the contents of register 0. This permits direct addressing of the 32 fullwords starting at the address contained in register X'46' without having to load a base register. After program loading, the contents of register X'46' are unpredictable and must be initialized before use.

3. If the general register specified by R is R0, all zeros are stored at the storage location, instead of the contents of R0.

Store Halfword

STH R,D(B) RS



The store halfword instruction stores bytes 0 and 1 of the first operand (the register specified by R) into the second operand in storage.

The storage address is formed by adding the displacement value to the contents of the base register specified by B. The displacement field allows for a displacement of between 0 and 126 bytes in multiples of 2 (64 halfwords).

Resulting Condition Latches: unchanged

Programming Notes

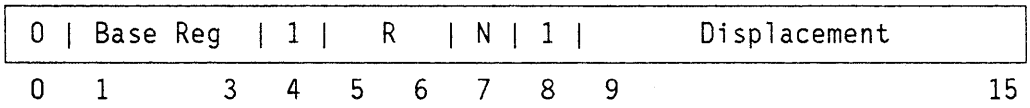
1. The low order-bit of the address is ignored since storage is addressed on halfword boundaries with this instruction.

2. If the general register specified by R is R0, all zeros are stored at the storage location, instead of the contents of R0.

3. If the base register specified is R0, the contents of R0 are **not** used as the base address. Instead, a program settable address located in CCU external register X'45' is used as a base address instead of the contents of register 0. This permits direct addressing of the 64 halfwords starting at the address contained in register X'45' without having to load a base register. After program loading, the contents of register X'45' are unpredictable and must be initialized before use.

Store Character

STC R(N),D(B) RS



The store character instruction stores byte 0 (N = 0) or byte 1 (N = 1) of the register specified by R into the second operand address in storage.

The storage address is formed by adding the displacement value to the contents of the base register specified by B. The displacement field allows for a displacement of between 0 and 127 bytes. The two bits of R form the two high-order bits of the register address; the low order bits are forced to 1 by hardware. For this reason, R is always an odd-numbered register (1, 3, 5, 7).

Resulting Condition Latches: unchanged

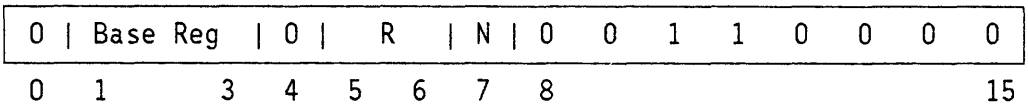
Programming Note

If the base register specified is R0, the contents of R0 are **not** used as the address. Instead, a program settable address located in CCU external register X'44' is used as a base address instead of the contents of register 0. This permits direct addressing of the 128 bytes starting at the address contained in register X'44' without having to load a base register. After program

loading, the contents of register X'44' are unpredictable and must be initialized before use.

Store Character and Count

STCT R(N),B RS



The eight bit character in byte 0 (N = 0) or byte 1 (N = 1) of the register specified by R is loaded into the location specified by the second operand address (contained in the base register designated by the B field). After the storage address has been obtained from the base register B, the contents of the base register is incremented by 1. After execution of the instruction, the base register normally contains an address 1 byte greater than before execution (see however programming note 2 below). The two bits of R form the two high-order bits of the register address; the low order bit is forced to 1 by hardware. For this reason, R is always an odd-numbered register (1, 3, 5, 7).

Resulting Condition Latches: unchanged

Programming Notes

1. If register 0 is specified in the B field, it causes an invalid operation check.

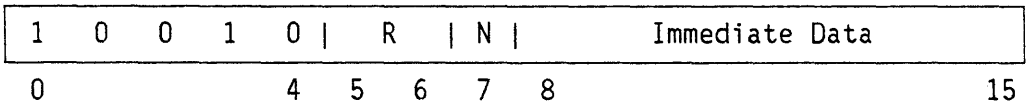
2. If R and B specify the same (odd) register, its contents are incremented by 1 **before** the selected byte of that register is stored.

Add Instructions

Name	Instruction	Type	Format 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
ARI	Add Reg Immediate	RI	1	0	0	1	0	R		N	0	Immediate Data						
AR	Add Register	RR	0		R2		0		R1		1	0	0	1	1	0	0	0
AHR	Add Halfword Reg	RR	0		R2		0		R1		1	0	0	1	0	0	0	0
ACR	Add Character Reg	RR	0	R2		N	0	R1		N	0	0	0	1	1	0	0	0

Add Register Immediate

ARI R(N),I RI



The second operand (Immediate Data field) is added to the first operand (byte 0 if N = 0, or bytes 0 and 1 if N = 1, of the register specified by R). The sum is then placed in the first operand location. If N = 0, byte 1 of the register remains unchanged. The two bits of R form the two high-order bits of the register address; the low order bit is forced to 1 by hardware. For this reason, R is always an odd-numbered register (1, 3, 5, 7).

Resulting Condition Latches:

1. N = 0

   C: an overflow occurred from byte 0 of R.

   Z: the result in byte 0 of R = 0

2. N = 1

   C: an overflow occurred from bytes 0 and 1 of R.

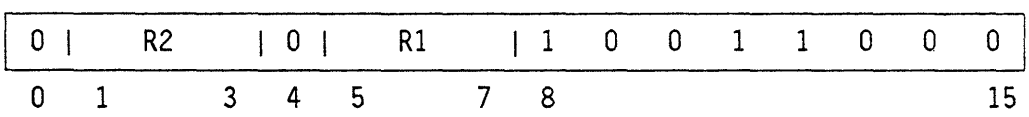
   Z: the result in bytes 0 and 1 of R = 0

Programming Note

The first operand includes byte X of the register specified by R. However, byte X does not affect the setting of the condition latches.

Add Register

AR R1,R2 RR



The second operand (bytes 0 and 1, **and** byte X of the register specified by R2) is added to the first operand (bytes 0 and 1, **and** byte X of the register specified by R1. Addition of the register operands is performed logically without regard to sign. The second operand remains unchanged, and the condition latches are set according to the result in the first operand.

Resulting Condition Latches:

C: an overflow occurred from R1.

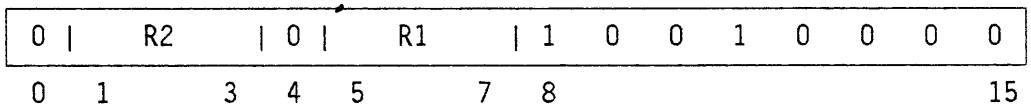
Z: the result in R1 = 0

Programming Note

If register 0 is specified by R1, a branch results to the address formed in register 0, and the condition latches remain unchanged.

Add Halfword Register

AHR R1,R2 RR



The second operand (bytes 0 and 1, but **not** byte X of R2) is added to the first operand (bytes 0 and 1, but **not** byte X, of the register specified by R1). Addition of the operands is performed logically without regard to sign. The second operand remains unchanged, and the condition latches are set according to the result in the first operand.

Resulting Condition Latches:

C: an overflow occurred from byte 0 of R1.

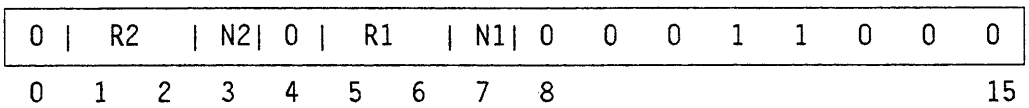
Z: the result in bytes 0 and 1 of R1 = 0

Programming Note

If register 0 is specified as R1, a branch results to the address formed in register 0, and the condition latches remain unchanged.

Add Character Register

ACR R1(N1),R2(N2) RR



The second operand (R2, byte 0 if N2 = 0, or byte 1 if N2 =1) is added to the first operand (R1, bytes X and 0 if N1 = 0, or bytes X, 0, and 1 if N1 = 1). The sum is then placed in the first operand location. If N1 = 0, byte 1 of the register remains unchanged.

The two bits of R1 and R2 form the two high-order bits of the register addresses; the low order bits are forced to 1 by hardware. For this reason, R is always an odd-numbered register (1, 3, 5, 7).

Resulting Condition Latches:

1. N = 0

   C: an overflow occurred from byte 0 of R1.

   Z: the result in byte 0 of R1 = 0

2. N = 1

   C: an overflow occurred from byte 0 of R1.

   Z: the result in bytes 0 and 1 of R1 = 0

Programming Note

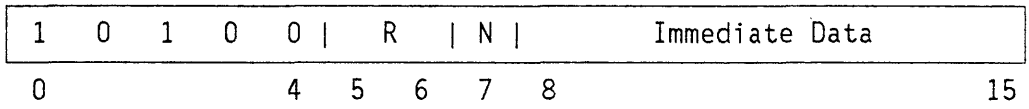
The first operand includes byte X of the register specified by R1. However, byte X does not affect the setting of the condition latches.

Subtract Instructions

Name	Instruction	Type	Format 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
SRI	Subt Reg Immediate	RI	1	0	1	0	0	R		N	0	Immediate Data						
SR	Subtract Register	RR	0		R2		0		R1		1	0	1	0	1	0	0	0
SHR	Subt Halfword Reg	RR	0		R2		0		R1		1	0	1	0	0	0	0	0
SCR	Subt Character Reg	RR	0	R2		N	0	R1		N	0	0	1	0	1	0	0	0

Subtract Register Immediate

SRI R(N),I RI



The second operand (Immediate Data field) is subtracted from the first operand (byte X and 0 if N = 0, or bytes X, 0 and 1 if N = 1, of the register specified by R). The result is then placed in the first operand location. The two bits of R form the two high-order bits of the register address; the low order bit is forced to 1 by hardware. For this reason, R is always an odd-numbered register (1, 3, 5, 7).

Before the subtraction is performed, the second operand is expanded with high-order zeros to equal the size of the first operand. Subtraction is performed by adding the two's complement of the second operand to the first operand and setting the appropriate condition latch. If the difference is less than zero, the result of the subtraction is in the two's complement form.

Resulting Condition Latches:

1. N = 0

   C: the result in byte 0 of R < 0

   Z: the result in byte 0 of R = 0

2. N = 1

   C: the result in bytes 0 and 1 of R < 0

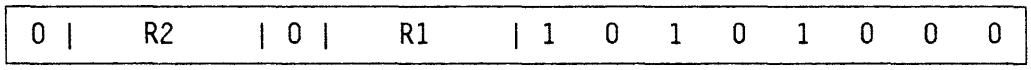
   Z: the result in bytes 0 and 1 of R = 0

Programming Note

The first operand includes byte X of the register specified by R. However, byte X does not affect the setting of the condition latches.

Subtract Register

SR R1,R2 RR



The second operand (bytes 0 and 1, **and** byte X of the register specified by R2) is subtracted from the first operand (bytes 0 and 1, **and** byte X, of the register specified by R1). Subtraction is performed by adding the two's complement of the second operand to the first operand. If the difference is less than zero, the result is in two's complement form. The second operand remains unchanged, and the condition latches are set according to the result in the first operand.

Resulting Condition Latches:

C: the result in R1 < 0.

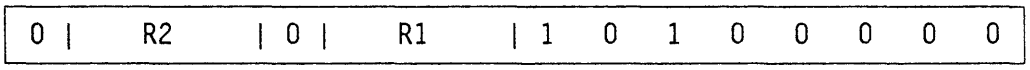
Z: the result in R1 = 0

Programming Note

If register 0 is specified as R1, a branch results to the address formed in register 0, and the condition latches remain unchanged.

Subtract Halfword Register

SHR R1,R2 RR



The second operand (bytes 0 and 1, but **not** byte X of R2) is subtracted from the first operand (bytes 0 and 1, but **not** byte X, of the register specified by R1). Subtraction is performed by adding the two's complement of the second operand to the first operand. If the difference is less than zero, the result is in two's complement form. The second operand remains unchanged, and the condition latches are set according to the result in the first operand.

Resulting Condition Latches:

C: the result in bytes 0 and 1 of R1 < 0.

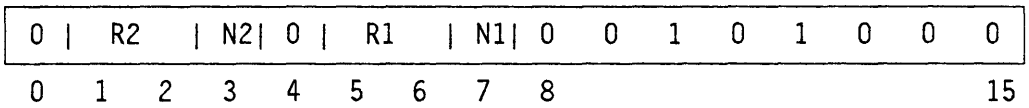
Z: the result in bytes 0 and 1 of R1 = 0

Programming Note

If register 0 is specified as R1, a branch results to the address formed in register 0, and the condition latches remain unchanged.

Subtract Character Register

SCR R1(N1),R2(N2) RR



The second operand (R2, byte 0 if N2 = 0, or byte 1 if N2 = 1) is subtracted from the first operand (R1, bytes X and 0 if N1 = 0, or bytes X, 0, and 1 if N1 = 1). The result is then placed in the first operand location. The two bits of R1 and R2 form the two high-order bits of the register addresses; the low order bit is forced to 1 by hardware. For this reason, R1 and R2 are always odd-numbered registers (1, 3, 5, 7).

Before the subtraction is performed, the second operand is expanded with high-order zeros to equal the size of the first operand. Subtraction is performed by adding the two's complement of the second operand to the first operand and setting the appropriate condition latch. If the difference is less than zero, the result of the subtraction is in the two's complement form.

Resulting Condition Latches:

1. N = 0

   C: the result in byte 0 of R1 < 0

   Z: the result in byte 0 of R1 = 0

2. N = 1

   C: the result in bytes 0 and 1 of R1 < 0

   Z: the result in bytes 0 and 1 of R1 = 0

Programming Note

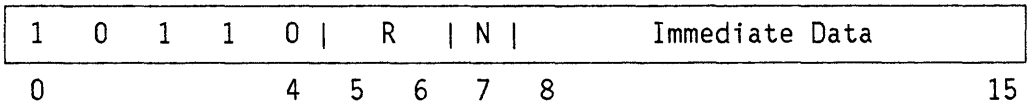
The first operand includes byte X of the register specified by R1. However, byte X does not affect the setting of the condition latches.

Compare Instructions

Name	Instruction	Type	Format 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CRI	Comp Reg Immediate	RI	1	0	1	1	0	R		N	Immediate Data							
CR	Compare Register	RR	0		R2		0		R1		1	0	1	1	1	0	0	0
CHR	Comp Halfword Reg	RR	0		R2		0		R1		1	0	1	1	0	0	0	0
CCR	Comp Character Reg	RR	0	R2		N	0	R1		N	0	0	1	1	1	0	0	0
TRM	Test R under Mask	RI	1	1	1	1	0	R		N	Mask Bits							

Compare Register Immediate

CRI R(N),I RI



The second operand (Immediate Data field) is compared with the first operand (byte 0 if N = 0, or byte 1 if N = 1, of the register specified by R), and the appropriate condition latch is set. The instruction performs a logical compare without regard to an eventual sign bit; all bits of each operand participate in the comparison. The two bits of R form the two high-order bits of the register address; the low order bit is forced to 1 by hardware. For this reason, R is always an odd-numbered register (1, 3, 5, 7).

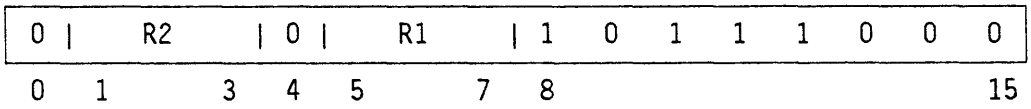
Resulting Condition Latches:

C: the value in the selected byte of R < I

Z: the value in the selected byte of R = I

Compare Register

CR R1,R2 RR



The second operand (bytes 0 and 1, **and** byte X of R2) is compared with the first operand (bytes 0 and 1, **and** byte X, of the register specified by R1). This instruction performs a logical compare without regard to the sign bit. The condition latches are set according to the result of the comparison.

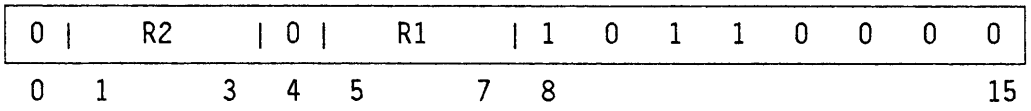
Resulting Condition Latches:

C: value in R1 < value in R2.

Z: value in R1 = value in R2.

Compare Halfword Register

CHR R1,R2 RR



The second operand (bytes 0 and 1, but **not** byte X of the register specified by R2) is compared with the first operand (bytes 0 and 1, but **not** byte X, of the register specified by R1). This instruction performs a logical compare without regard to the sign bit. The condition latches are set according to the result of the comparison.

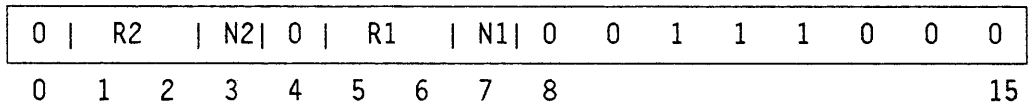
Resulting Condition Latches:

C: bytes 0 and 1 of R1 < bytes 0 and 1 of R2.

Z: bytes 0 and 1 of R1 = bytes 0 and 1 of R2.

Compare Character Register

CCR R1(N1),R2(N2) RR



The second operand (R2, byte 0 if N2 = 0, or byte 1 if N2 = 1) is compared with the first operand (R1, byte 0 if N1 = 0, or byte 1 if N1 = 1), and the appropriate condition latch is set. The instruction performs a logical compare without regard to an eventual sign bit; all bits of each operand participate in the comparison. The two bits of R1 and R2 form the two high-order bits of the register addresses; the low order bits are forced to 1 by hardware. For this reason, R1 and R2 are always odd-numbered registers (1, 3, 5, 7).

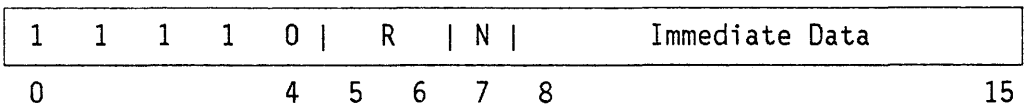
Resulting Condition Latches:

C: the value in the selected byte of R1 < selected byte of R2.

Z: the value in the selected byte of R1 = selected byte of R2.

Test Register Under Mask

TRM R(N),I RI



The second operand (Immediate Data field) is used as a mask to test the bits of the first operand (byte 0 if N = 0, or byte 1 if N = 1, of the register specified by R). A mask bit of 1 indicates that the corresponding register bit is to be tested; when the mask bit is 0, the register bit is ignored. Testing is done by ANDing the selected byte of the register with the immediate operand. The contents of the register remain unchanged. The two bits of R form the two high-order bits of the register address; the low order bit is forced to 1 by hardware. For this reason, R is always an odd-numbered register (1, 3, 5, 7).

Resulting Condition Latches:

C: the result of testing the selected byte of R ≠ 0.

Z: the result of testing the selected byte of R = 0.

XOR Instructions

Name	Instruction	Type	Format 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
XRI	XOR Reg Immediate	RI	1	1	0	0	0	R		N	Immediate Data							
XR	XOR Register	RR	0		R2		0		R1		1	1	0	0	1	0	0	0
XHR	XOR Halfword Reg	RR	0		R2		0		R1		1	1	0	0	0	0	0	0
XCR	XOR Character Reg	RR	0	R2		N	0	R1		N	0	1	0	0	1	0	0	0

XOR Register Immediate

XRI R(N),I RI

The second operand (Immediate Data field) is exclusive ORed with the first operand (byte 0 if N = 0, or byte 1 if N = 1, of the register specified by R). The resulting byte is placed in the first operand location, and the appropriate condition latch is set. The remaining bytes of R are unchanged.

Operands are treated as unstructured logical quantities, and the connective exclusive OR is applied bit by bit. A bit position in the result is set to 1 if the corresponding bit positions in the two operands are unlike; otherwise, the resulting bit position is set to 0.

The two bits of R form the two high-order bits of the register address; the low order bit is forced to 1 by hardware. For this reason, R is always an odd-numbered register (1, 3, 5, 7).

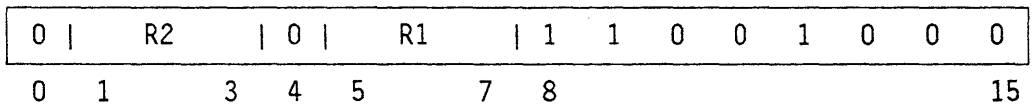
Resulting Condition Latches:

C: the result in the selected byte of R ≠ 0.

Z: the result in the selected byte of R = 0

XOR Register

XR R1,R2 RR



The second operand (bytes 0 and 1, **and** byte X of the register specified by R2) is exclusive ORed with the first operand (bytes 0 and 1, **and** byte X, of the register specified by R1). Operands are treated as unstructured logical quantities, and the connective exclusive OR is applied bit by bit. A bit position in the result is set to 1 if the corresponding bit positions in the two operands are unlike; otherwise, the resulting bit is set to 0. The second

operand is unchanged, and the condition latches are set according to the result of the operation.

Resulting Condition Latches:

C: the result in R1 ≠ 0.

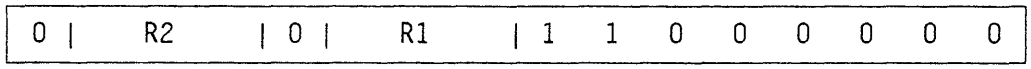
Z: the result in R1 = 0.

Programming Note

If register 0 is specified as R1, a branch results to the address formed in register 0, and the condition latches remain unchanged.

XOR Halfword Register

XHR R1,R2 RR



The second operand (bytes 0 and 1, but **not** byte X of the register specified by R2) is exclusive ORed with the first operand (bytes 0 and 1, but **not** byte X, of the register specified by R1). Operands are treated as unstructured logical quantities, and the connective exclusive OR is applied bit by bit. A bit position in the result is set to 1 if the corresponding bit positions in the two operands are unlike; otherwise, the resulting bit is set to 0. The second operand is unchanged, and the condition latches are set according to the result of the operation.

Resulting Condition Latches:

C: the result in bytes 0 and 1 of R1 ≠ 0.

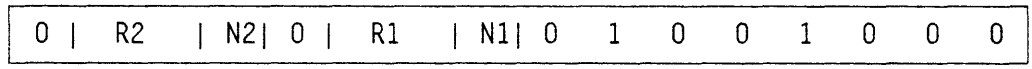
Z: the result in bytes 0 and 1 of R1 = 0.

Programming Note

If register 0 is specified as R1, a branch results to the address formed in register 0, and the condition latches remain unchanged.

XOR Character Register

XCR R1(N1),R2(N2) RR



The second operand (R2, byte 0 if N2 = 0, or byte 1 if N2 = 1) is exclusive ORed with the first operand (R1, byte 0 if N1 = 0, or byte 1 if N1 = 1). The result is placed in the first operand location, and the appropriate condition latch is set. The remaining bytes of R are unchanged.

Operands are treated as unstructured logical quantities, and the connective exclusive OR is applied bit by bit. A bit position in the result is set to 1

if the corresponding bit positions in the two operands are unlike; otherwise, the resulting bit position is set to 0.

The two bits of R1 and R2 form the two high-order bits of the register addresses; the low order bits are forced to 1 by hardware. For this reason, R1 and R2 are always odd-numbered registers (1, 3, 5, 7).

Resulting Condition Latches:

C: the result in the selected byte of R ≠ 0.

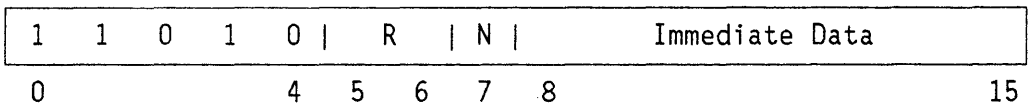
Z: the result in the selected byte of R = 0

OR Instructions

Name	Instruction	Type	Format 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
ORI	OR Reg Immediate	RI	1	1	0	1	0	R		N	Immediate Data							
OR	OR Register	RR	0		R2		0		R1		1	1	0	1	1	0	0	0
OHR	OR Halfword Reg	RR	0		R2		0		R1		1	1	0	1	0	0	0	0
OCR	OR Character Reg	RR	0	R2		N	0	R1		N	0	1	0	1	1	0	0	0

OR Register Immediate

ORI R(N),I RI



The second operand (Immediate Data field) is ORed with the first operand (byte 0 if N = 0, or byte 1 if N = 1, of the register specified by R). The result of the operation is placed in the first operand location; the remaining bytes of the register remain unchanged.

Operands are treated as unstructured logical quantities, and the connective inclusive OR is applied bit by bit. A bit position in the result is set to one if the corresponding bit in either one or both of the operands contains a 1; otherwise, the result bit remains at zero.

The two bits of R form the two high-order bits of the register address; the low order bit is forced to 1 by hardware. For this reason, R is always an odd-numbered register (1, 3, 5, 7).

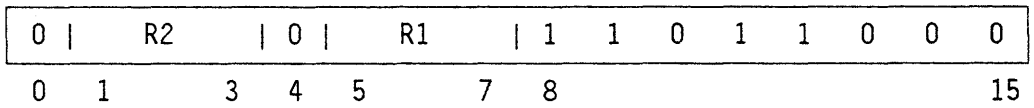
Resulting Condition Latches:

C: the result in the selected byte of R ≠ 0.

Z: the result in the selected byte of R = 0

OR Register

OR R1,R2 RR



The second operand (bytes 0 and 1, **and** byte X of the register specified by R2) is ORed with the first operand (bytes 0 and 1, **and** byte X, of the register specified by R1). Operands are treated as unstructured logical quantities, and the connective inclusive OR is applied bit by bit. A bit position in the result is set to 1 if the corresponding bit position in either one or both of the operands contains a 1. Otherwise the result bit contains 0. Any value in the operands or in the result is valid. The second operand

is unchanged, and the condition latches are set according to the result of the operation.

Resulting Condition Latches:

C: the result in R1 ≠ 0.

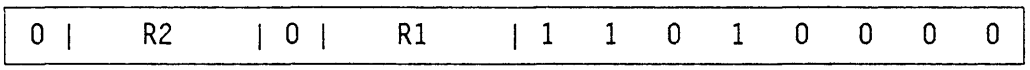
Z: the result in R1 = 0.

Programming Note

If register 0 is specified as R1, a branch results to the address formed in register 0, and the condition latches remain unchanged.

OR Halfword Register

OHR R1,R2 RR



The second operand (bytes 0 and 1, but **not** byte X of the register specified by R2) is ORed with the first operand (bytes 0 and 1, but **not** byte X, of the register specified by R1). Operands are treated as unstructured logical quantities, and the connective inclusive OR is applied bit by bit. A bit position in the result is set to 1 if the corresponding bit position in either one or both of the operands contains a 1. Otherwise the result bit contains 0. Any value in the operands or in the result is valid. The second operand is unchanged, and the condition latches are set according to the result of the operation.

Resulting Condition Latches:

C: the result in bytes 0 and 1 of R1 ≠ 0.

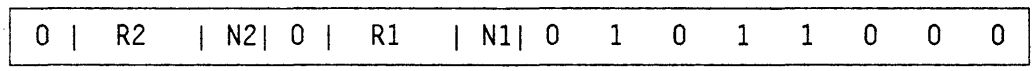
Z: the result in bytes 0 and 1 of R1 = 0.

Programming Note

If register 0 is specified as R1, a branch results to the address formed in register 0, and the condition latches remain unchanged.

OR Character Register

OCR R1(N1),R2(N2) RR



The second operand (R2, byte 0 if N2 = 0, or byte 1 if N2 = 1) is ORed with the first operand (R1, byte 0 if N1 = 0, or byte 1 if N1 = 1). The result of the operation is placed in the first operand location; the remaining bytes of the register remain unchanged.

Operands are treated as unstructured logical quantities, and the connective inclusive OR is applied bit by bit. A bit position in the result is set to

one if the corresponding bit in either one or both of the operands contains a 1; otherwise, the result bit remains at zero.

The two bits of R1 and R2 form the two high-order bits of the register addresses; the low order bits are forced to 1 by hardware. For this reason, R1 and R2 are always odd-numbered registers (1, 3, 5, 7).

Resulting Condition Latches:

C: the result in the selected byte of R ≠ 0.

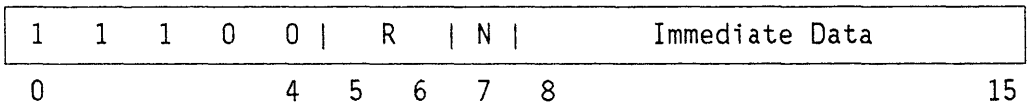
Z: the result in the selected byte of R = 0

AND Instructions

			Format													
Name	Instruction	Type	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
NRI	AND Reg Immediate	RI	1	1	1	0	0	R		N	Immediate Data							
NR	AND Register	RR	0		R2		0		R1		1	1	1	0	1	0	0	0
NHR	AND Halfword Reg	RR	0		R2		0		R1		1	1	1	0	0	0	0	0
NCR	AND Character Reg	RR	0	R2		N	0	R1		N	0	1	1	0	1	0	0	0

AND Register Immediate

NRI R(N),I RI



The second operand (Immediate Data field) is ANDed with the first operand (byte 0 if N = 0, or byte 1 if N = 1, of the register specified by R). The result of the operation is placed in the first operand location; the remaining bytes of the register remain unchanged.

Operands are treated as unstructured logical quantities, and the connective AND is applied bit by bit. A bit position in the result is set to one if the corresponding bit position in both operands contains a 1; otherwise the result bit is set to zero. All bits of each operand participate in the operation. All values of operands and result are valid.

The two bits of R form the two high-order bits of the register address; the low order bit is forced to 1 by hardware. For this reason, R is always an odd-numbered register (1, 3, 5, 7).

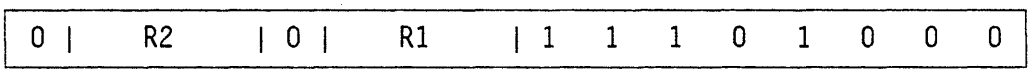
Resulting Condition Latches:

C: the result in the selected byte of R ≠ 0.

Z: the result in the selected byte of R = 0

AND Register

NR R1,R2 RR



The second operand (bytes 0 and 1, **and** byte X of the register specified by R2) is ANDed with the first operand (bytes 0 and 1, **and** byte X, of the register specified by R1). Operands are treated as unstructured logical quantities, and the connective AND is applied bit by bit. A bit position in the result is set to 1 if the corresponding bit positions in both operands contains a 1; otherwise the result bit is set to 0. All bits of each operand participate in the operation. Any value in the operands or in the result is valid. The second operand is unchanged, and the condition latches are set according to the result of the operation.

Resulting Condition Latches:

C: the result in R1 ≠ 0.

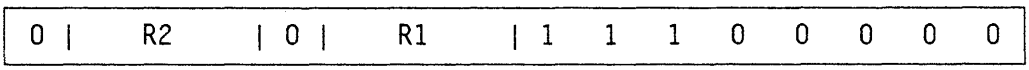
Z: the result in R1 = 0.

Programming Note

If register 0 is specified as R1, a branch results to the address formed in register 0, and the condition latches remain unchanged.

AND Halfword Register

NHR R1,R2 RR



The second operand (bytes 0 and 1, but **not** byte X of the register specified by R2) is ANDed with the first operand (bytes 0 and 1, but **not** byte X, of the register specified by R1). Operands are treated as unstructured logical quantities, and the connective AND is applied bit by bit. A bit position in the result is set to 1 if the corresponding bit positions in both operands contains a 1; otherwise the result bit is set to 0. All bits of each operand participate in the operation. Any value in the operands or in the result is valid. The second operand is unchanged, and the condition latches are set according to the result of the operation.

Resulting Condition Latches:

C: the result in bytes 0 and 1 of R1 ≠ 0.

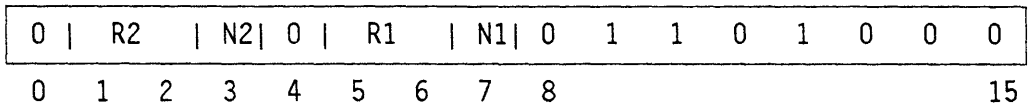
Z: the result in bytes 0 and 1 of R1 = 0.

Programming Note

If register 0 is specified as R1, a branch results to the address formed in register 0, and the condition latches remain unchanged.

AND Character Register

NCR R1(N1),R2(N2) RR



The second operand (R2, byte 0 if N2 = 0, or byte 1 if N2 = 1) is ANDed with the first operand (R1, byte 0 if N1 = 0, or byte 1 if N1 = 1). The result of the operation is placed in the first operand location; the remaining bytes of the register remain unchanged.

Operands are treated as unstructured logical quantities, and the connective AND is applied bit by bit. A bit position in the result is set to one if the corresponding bit position in both operands contains a 1; otherwise the result bit is set to zero. All bits of each operand participate in the operation. All values of operands and result are valid.

The two bits of R1 and R2 form the two high-order bits of the register addresses; the low order bits are forced to 1 by hardware. For this reason, R1 and R2 are always odd-numbered registers (1, 3, 5, 7).

Resulting Condition Latches:

C: the result in the selected byte of R ≠ 0.

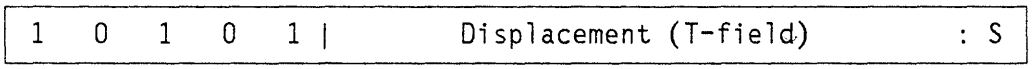
Z: the result in the selected byte of R = 0

Branch Operations

Name	Instruction	Type	Format 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
B	Branch	RT	1 0 1 0 1 Displacement
BCL	Branch on C latch	RT	1 0 0 1 1 Displacement
BZL	Branch on Z latch	RT	1 0 0 0 1 Displacement
BCT	Branch on Count	RT	1 0 1 1 1 R N 1 Displacement
BB	Branch on Bit	RT	1 1 M M 1 R N M Displacement
BALR	Branch & Link Reg	RR	0 R2 0 R1 0 1 0 0 0 0 0 0
BAL	Branch and Link	RA	1 0 1 1 1 R 0 1 |0 0|Ad Byte Ext
			Addr Byte 0 Addr Byte 1

Branch

B T RT

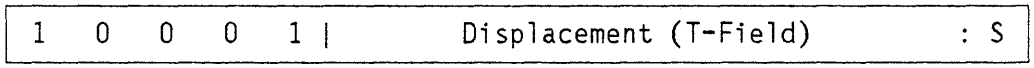


This instruction causes an unconditional branch to the branch address. The branch address is formed by adding the displacement value in the T field to the address of the next sequential instruction. Therefore, the branch address is always calculated relative to the next sequential instruction. The T-field allows a displacement of between +1023 and -1023 halfwords. The low order bit position (bit 15: marked S for sign) indicates whether the displacement is positive (0), or negative (1).

Resulting Condition Latches: unchanged

Branch on Z Latch

BZL T RT

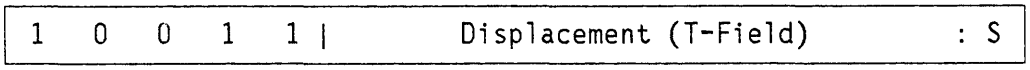


This instruction tests the state of the Z condition latch associated with the active group of general registers. If the tested latch is not set (0), the next sequential instruction is executed. If the tested latch is set (1), the instruction located at the branch address is executed. The branch address is formed by adding the displacement value in the T field to the address of the next sequential instruction. Therefore, the branch address is always calculated relative to the next sequential instruction. The T-field allows a displacement of between +1023 and -1023 halfwords. The low order bit position (bit 15: marked S for sign) indicates whether the displacement is positive (0), or negative (1).

Resulting Condition Latches: unchanged

Branch on C Latch

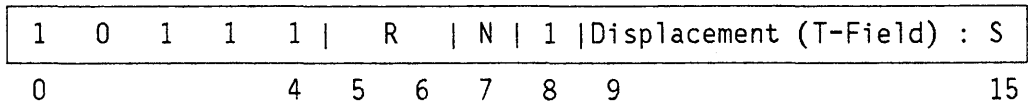
BCL T RT



This instruction behaves in exactly the same way as the 'Branch on Z Latch' instruction, except that it is the C latch which is tested.

Branch on Count

BCT R(N),T RT



The count value contained in the register specified by R is decremented by one and then tested for zero. If the result is zero, the next sequential instruction is executed. If the result is not zero, the instruction located at the branch address is executed. The count is contained in byte 0 only (if N = 0) or in both bytes 0 and 1 (if N = 1) of the register. The branch address is formed by adding the displacement value in the T field to the address of the next sequential instruction. Therefore, the branch address is always calculated relative to the next sequential instruction. The T-field allows a displacement of between +63 and -63 halfwords. The low order bit position (bit 15: marked S for sign) indicates whether the displacement is positive (0), or negative (1).

The two bits of R form the two high-order bits of the register address; the low order bit is forced to 1 by hardware. For this reason, R is always an odd-numbered register (1, 3, 5, 7).

Resulting Condition Latches: unchanged

Programming Notes

1. If N = 0, and before execution the count in byte 0 was zero, the effective count is 256.
2. If N = 1, and before execution the count in bytes 0 and 1 was zero, the effective count is 65536.
3. Byte X is ignored.

Branch on Bit

BB R(N,M),T RT

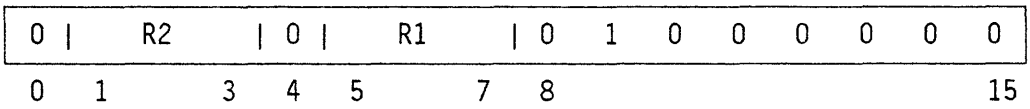
This instruction tests the state of a specified bit in a general register. The three bits of the M (mask) field specify which one of the 8 bits of byte 0 (N = 0) or byte 1 (N = 1) of the register is to be tested. If the bit tested is zero, the next sequential instruction is executed. If the bit tested is a one, the instruction located at the branch address is executed. The branch address is formed by adding the displacement value in the T field to the address of the next sequential instruction. Therefore, the branch address is always calculated relative to the next sequential instruction. The T-field allows a displacement of between +63 and -63 halfwords. The low order bit position (bit 15: marked S for sign) indicates whether the displacement is positive (0), or negative (1).

The two bits of R form the two high-order bits of the register address; the low order bit is forced to 1 by hardware. For this reason, R is always an odd-numbered register (1, 3, 5, 7).

Resulting Condition Latches: unchanged

Branch and Link Register

BALR R1,R2 RR



The address of the next sequential instruction (from the IAR, R0) is stored as link information in the register specified by R1. Subsequently, the instruction address in register 0 is replaced by the branch address (the contents of the register specified by R2), and the branch is executed. The branch address is obtained from R2 before the link information is stored in R1.

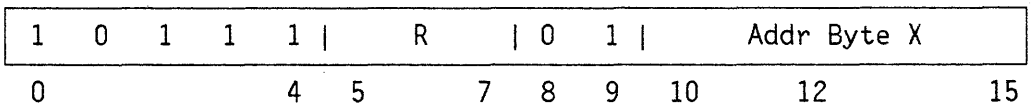
Programming Note

Since register 0 is the instruction address register (IAR), no linkage is provided if it is specified in the R1 field, and no branch occurs if it is specified in the R2 field.

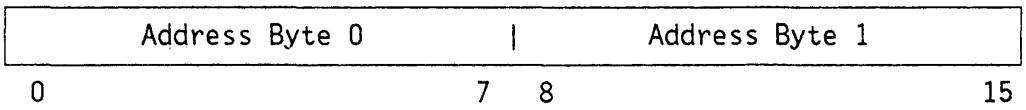
Branch and Link

BAL R,A RA

First halfword



Second halfword



The branch and link instruction is a 4-byte instruction which causes an unconditional branch. The address of the next sequential instruction is stored as link information in the register specified by R. The instruction address in register 0 is then replaced by the branch address taken from the 22-bit address field, and the branch is executed.

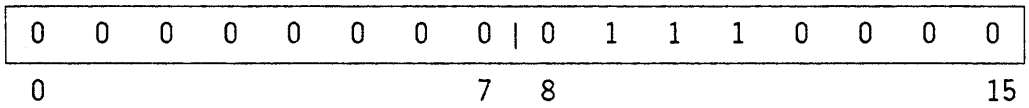
Resulting Condition Latches: unchanged

Programming Note

Since register 0 is the instruction address register, no linkage is provided if it is specified in the R field.

Exit

EXIT - EXIT



The exit instruction is used to exit from the active program level. The interrupt priority logic then determines which group of general registers to select as the active group for the next program operation, and automatically loads the work registers with the contents of the active group. This instruction also resets the 'interrupt entered' latch for the program level that executes it.

If the exit instruction is executed at program level 5, the level 4 supervisor call interrupt request (SVC L4) is set. The next instruction is then normally the instruction at the starting address for program level 4. However, if other interrupt requests of higher priority are present, the next instruction executed is the instruction at the starting address of the highest priority program level requesting an interrupt.

Resulting Condition Latches: unchanged

Programming Notes

1. If program level 4 is masked, and no unmasked higher level interrupts are pending, program execution continues at level 5 after the SVC level 4 interrupt request is set.

2. If this instruction follows an IOH or IOHI instruction (used to clear an adapter interrupt), at least 8 CCU cycles must separate the execution of the IOH/IOHI and the Exit instructions.

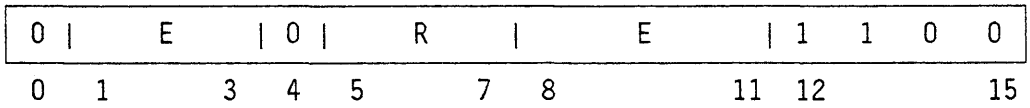
3. When executing at any interrupt level, if another interrupt is pending when the EXIT instruction is executed, the CCU momentarily returns to the original interrupted level **without instruction execution,** and then goes to the new interrupt level. This action is invisible, except if branch tracing is done; in this case, the passage through the original interrupted level is recorded in the branch trace list.

Input/Output Instructions (RE, RR, and RA)

			Format															
Name	Instruction	Type	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
IN	CCU Reg Input	RE	0		E		0		R			E			1	1	0	0
OUT	CCU Reg Output	RE	0		E		0		R			E			0	1	0	0
IOH	Adapter I/O	RR	0		R2		0		R1		0	1	0	1	0	0	0	0
IOHI	Adapter I/O Immed	RI	0	0	0	0	0		R		0	1	1	1	0	0	0	0
			(Second half word depends on the adapter)															

CCU Register Input

IN R,E RE



The input instruction loads the general register specified by R with the contents of one of the 128 CCU registers, as specified by the E field. Throughout this manual, CCU Register Input instructions are referred to in the form: Input X'nn' where 'nn' is the hexadecimal address of the CCU register. See the Appendixes to this manual for a list of the external registers and for their detailed bit structure.

Resulting Condition Latches: unchanged

Programming Notes

1. If register 0 of the active group of general registers is specified by R, this instruction causes a branch to the address formed in register 0.

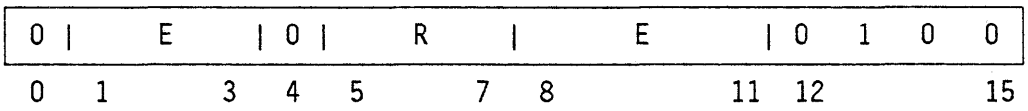
2. This instruction is a **privileged** instruction, executable only at program levels 1, 2, 3, or 4. Any attempt to execute the instruction at program level 5 causes a level 1 interrupt request (Level 5 I/O Error) to be set.

3. The following input external register addresses are invalid: X'28' through X'2F', X'49' through X'4F', X'60' through X'6F', and X'78'. If the control program tries to read one of these registers, an invalid operation condition is detected, and a level 1 interrupt is set. The instruction is not executed.

4. If the control program uses an IOH or IOHI instruction to clear an adapter interrupt, at least 8 CCU cycles must be performed before executing an Input X'77' or an Input X'7E' instruction.

CCU Register Output

OUT R,E RE



The output instruction loads one of the 128 CCU registers specified by the E field with the contents of the general register specified by R. Throughout this manual, CCU Register Output instructions are referred to in the form: Output X'nn' where 'nn' is the hexadecimal address of the CCU register. All registers are three bytes long. See the Appendixes to this manual for a list of the CCU registers and for their detailed bit structure.

Resulting Condition Latches: unchanged

Programming Notes

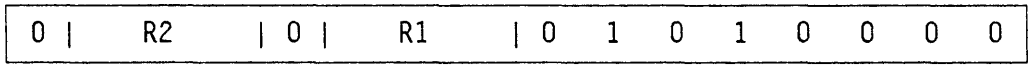
1. If register 0 of the active group of general registers is specified by E, this instruction causes a branch to the address formed in register 0.

2. This instruction is a **privileged** instruction, executable only at program levels 1, 2, 3, or 4. Any attempt to execute the instruction at program level 5 causes a level 1 interrupt request (Level 5 I/O Error) to be set.

3. The following output external register addresses are invalid: X'28' through X'2F', X'49' through X'4F', X'60' through X'6F', and X'75'. If the control program tries to write into one of these registers, an invalid operation condition is detected, and a level 1 interrupt is set. The instruction is not executed.

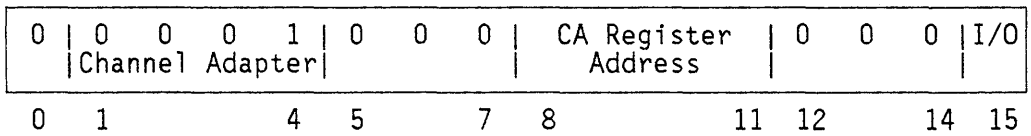
Adapter Input/Output

IOH R1,R2 RR



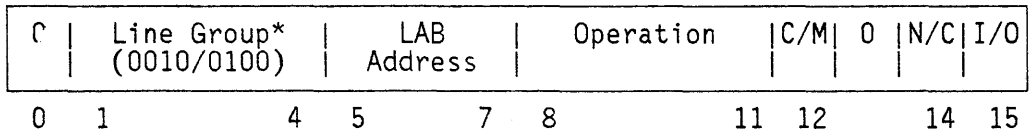
This instruction transfers the contents of the register specified by R1 to the channel adapter/communication scanner, or places information coming from the channel adapter/communication scanner into the register specified by R1. The adapter, the adapter command or register, and the direction of data movement are all specified by the contents of R2. This instruction cannot be used to address registers within the CCU. For the instruction to execute correctly, R2 must be loaded as follows:

Channel Adapter



I/O = input/output bit: 0 = output, 1 = input

Communication Scanners



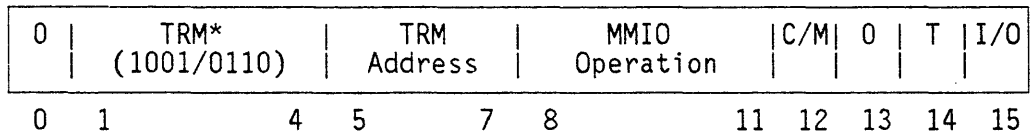
* 0010 selects the first group of 16 lines; 0100 selects the second group. In the case of LAB 2, this division corresponds to the two communication scanners.

C/M = CCU/MOSS bit: 0 = initiated by CCU, 1 = initiated by MOSS

N/C: 0 = normal mode, 1 = character mode

I/O = input/output bit: 0 = output, 1 = input

Token-Ring Multiplexor



*1001 indicates that the operation is for the TRM. 0110 gets the address for the Get Line instruction.

C/M = CCU/MOSS bit: 0 = initiated by CCU, 1 = initiated by MOSS

T = TIC bit: 0 = TIC 1, 1 = TIC 2.

I/O = input/output bit: 0 = output, 1 = input.

Redrive Logic

Bits 1 through 4 contain the group address (always 1 0 0 0)

Bits 5 through 7 contain the primary redrive address (PRA)

Bits 8 through 11 indicate the redrive command

Bits 12 through 14 contain the secondary redrive address (SRA)

I/O = input/output bit: 0 = output, 1 = input

Resulting Condition Latches:

C: the exception line was raised by the adapter

Z: the exception line was not raised by the adapter

Programming Notes

1. This instruction is a **privileged** instruction, executable only at program levels 1, 2, 3, or 4. Any attempt to execute it at program level 5 causes a level 1 input/output check interrupt request (level 5 I/O error) to be set.

2. A time out level 1 interrupt request occurs if no valid response is received from the adapter within a specified time.

3. Byte X of register R2 is not used.

4. Byte X of register R1 is set to all zeros if the operation is read.

5. If register 0 is specified in the R1 field, an Invalid OP Check L1 interrupt request occurs; the instruction is not executed.

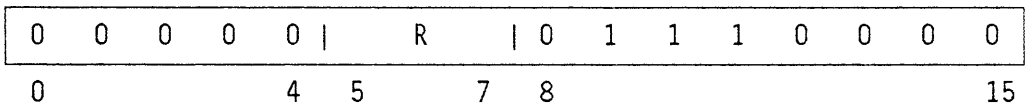
6. If the R2 field is 0, CCU external register X'48' (IOH Address Substitution) is used in place of register 0 to specify the external adapter register.

7. If this instruction is used to clear an interrupt and is followed by an Exit instruction, at least 8 CCU cycles must separate the execution of the IOH and the Exit instructions.

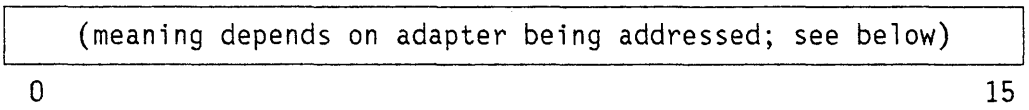
Adapter Input/Output Immediate

IOHI R,A RA

First halfword

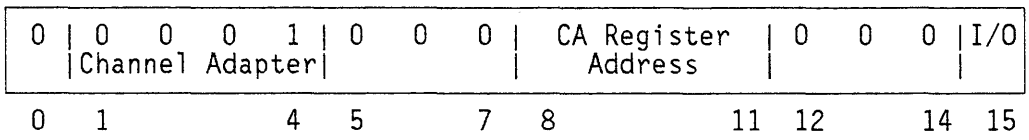


Second halfword

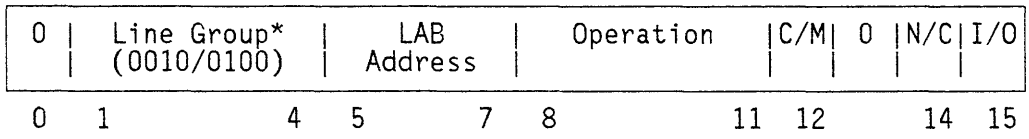


This instruction transfers the contents of the register specified by R to the channel adapter/communication scanner, or places information coming from the channel adapter/communication scanner into the register specified by R. The adapter, the adapter command or register, and the direction of data movement are all specified by the contents of the second halfword. This instruction cannot be used to address registers within the CCU. For the instruction to execute correctly, the second halfword must be composed as follows:

Channel Adapter



I/O = input/output bit: 0 = output, 1 = input Communication Scanners



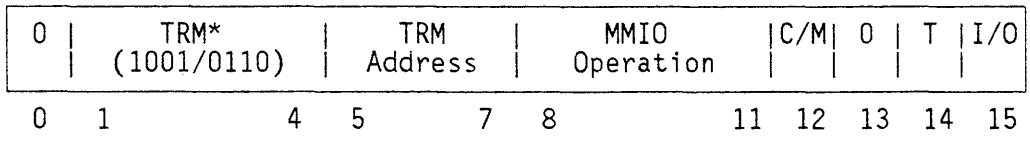
* 0010 selects the first group of 16 lines; 0100 selects the second group. In the case of LAB 2, this division corresponds to the two communication scanners.

C/M = CCU/MOSS bit: 0 = initiated by CCU, 1 = initiated by MOSS

N/C: 0 = normal mode, 1 = character mode

I/O = input/output bit: 0 = output, 1 = input

Token-Ring Multiplexor



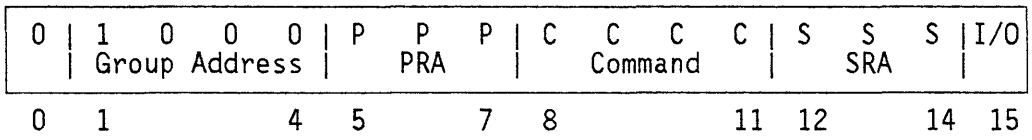
*1001 indicates that the operation is for the TRM. 0110 gets the address for the Get Line instruction.

C/M = CCU/MOSS bit: 0 = initiated by CCU, 1 = initiated by MOSS

T = TIC bit: 0 = TIC 1, 1 = TIC 2.

I/O = input/output bit: 0 = output, 1 = input.

Redrive Logic



Bits 1 through 4 contain the group address (always 1 0 0 0)

Bits 5 through 7 contain the primary redrive address (PRA)

Bits 8 through 11 indicate the redrive command

Bits 12 through 14 contain the secondary redrive address (SRA)

I/O = input/output bit: 0 = output, 1 = input

Resulting Condition Latches:

C: the exception line was raised by the adapter

Z: the exception line was not raised by the adapter

Programming Notes

1. This instruction is a **privileged** instruction, executable only at program levels 1, 2, 3, or 4. Any attempt to execute it at program level 5 causes a level 1 input/output check interrupt request (level 5 I/O error) to be set.

2. A time out level 1 interrupt request occurs if no valid response is received from the adapter within a specified time.

3. Byte X of register R is not used, and is set to all zeros if the operation is read.

4. If register 0 is specified in the R field, an EXIT operation occurs.

5. If this instruction is used to clear an interrupt and is followed by an Exit instruction, at least 8 CCU cycles must separate the execution of the IOHI and the Exit instructions.

CHAPTER 3. CENTRAL CONTROL UNIT (CCU)

This chapter gives the reader a basic understanding of the operation of the Central Control Unit (CCU) and the requirements necessary to control its operation.

The CCU is interrupt driven, operating in response to interrupts coming from the channel adapters, the communication scanners, the MOSS, and from other program levels via program controlled interrupts (PCIs). However, the interrupt system is not described here; for an overview, refer to the sections 'Program Levels' and 'Interrupts' in Chapter 1. For details of the channel adapter and communication scanner interrupts, refer to chapter 5. For details of the token-ring subsystem interrupts (3725 only), refer to chapter 6.

The CCU contains the circuits and data flow paths needed for the following operations:

- Accept interrupts
- Execute the instruction set
- Address storage
- Perform arithmetic and logical processing of data
- Control the attached adapters

Operation of the CCU is under the control of the programs in storage.

The data flow in the CCU is implemented in hardware in the 3725, in microcode in the 3720/3721, and controlled by the control program. The data flow for a particular operation is determined by the instruction, cycle steal, or control operation that is being executed.

CCU REGISTERS

The CCU contains a number of registers, most of which are accessible to the program. The registers are implemented in hardware in the 3725, and mapped into local storage in the 3720/21. Their use, however, is the same for both controllers. The bits of these registers are described in detail under the heading **CCU Input/Output Instructions** in this chapter.

Operation Register

The operation register holds the first 16 bits of the instruction currently being executed. It is not available to the program, but may be displayed on the screen.

Storage Address Register (SAR)

The storage address register holds the storage address that is being used, or that was last used. It is not available to the program, but may be displayed on the screen.

CCU Work Registers

The CCU work registers are a group of eight hardware registers for the immediate use of the CCU. They may be used by the program without restriction, with the exception of register 0 which is used as the Instruction Address Register.

Instruction Address Register (IAR)

The IAR is an implied base register, and always contains the address of the next instruction to be executed. It is always incremented to point to the next sequential instruction before the current instruction is executed. In most cases, the next halfword in storage contains the next instruction to be executed. Sometimes, however, the contents of the IAR are changed as the result of the instruction being executed. Execution of a branch instruction, for example, can cause the IAR to be loaded with a storage address other than the next sequential instruction. Refer to the descriptions of the individual instructions in Chapter 3 for the results of using register 0, and the precautions to be taken.

CCU External Registers

The CCU register addressing scheme can address up to 128 registers; however, not all of these register addresses are used.

Note that some instructions can address two different registers, one of which is only used for input, the other being used only for output. For example, the instruction Input X'71' reads the contents of local storage position X'71', but Output X'71' sets hardware display register 1.

Some instructions do not address a register at all. For example, Output X'7B', which forces a program controlled interrupt at level 2.

The table below shows the name and address of the CCU external registers:

Code	Meaning	Ext reg addr	Type
00-27	General Registers	LS X'00'-X'27'	I/O
30-35	Channel Adapter CS Address Pointers	LS X'30'-X'35'	I/O
36-3E	Reserved Pointer Registers	LS X'36'-X'3E'	I/O
3F	Communication Scanner CS Address Pointer	LS X'3F'	I/O
40-43	Interrupt Start Addresses	LS X'40'-X'43'	I/O
44	Byte Operations Base Register	LS X'44'	I/O
45	Halfword Operations Base Register	LS X'45'	I/O
46	Fullword Operations Base Register	LS X'46'	I/O
48	IOH Address Substitution Register	LS X'48'	I/O
50-5F	Programmable Registers	LS X'50'-X'5F'	I/O
70	Storage Size Installed	HW register	In
71	Operator Address/Data Entry Register	LS X'71'	In
71	Display Register 1	HW register	Out
72	Operator Display/Function Select Control	LS X'72'	In
72	Display Register 2	HW register	Out
73	Insert Storage Protect/Address Exception Key	HW register	In
73	Set Storage Protect/Address Exception Key	HW register	Out
74	Lagging Address Register	HW register	In
75	CCW for AIO Operations	HW register	In
76	CCU Level 1 Interrupt Requests on I/O ops.	HW register	In
76	Miscellaneous Control 1	HW register	Out
77	Adapter Levels 2 and 3 Interrupt Requests	HW register	In
77	Miscellaneous Control 2	HW register	Out
79	Utility	HW register	In
79	Utility	HW register	Out
7A	High Res. Timer/Utilization Counter	HW register	In
7A	High Res. Timer/Utilization Counter Control	HW register	Out
7B	Branch Trace Address Pointer	LS X'7B'	In
7C	Branch Trace Buffer Count	LS X'7C'	In
7D	CCU Hardware Check Register	HW register	In
7E	CCU Level 1 Interrupt Requests	HW register	In
7E	Set Program Interrupt Mask Bits	HW register	Out
7F	CCU L2, 3, or 4 Interrupt Requests	HW register	In
7F	Reset Program Interrupt Mask Bits	HW register	Out

Note: Output instructions corresponding to codes X'70', X'78', X'7B', X'7C', and X'7D' exist, but do not correspond to any register; instead, they perform a function.

General Registers

The first 40 external registers (00 through 27) are called the **General Registers.** They are divided into five groups, numbered 0 through 4; each group contains eight registers numbered 0 through 7 and is assigned to a specific program level.

The first general register of each group (register 0) is used as the instruction address register (IAR) of the corresponding program level. As the program is executed, the eight local storage registers (with the exception of the IAR) are updated with the contents of the work registers when the register is used in the execution of an instruction. Refer to Figure 1-3 on page 1-7 for the relationship between the work registers and the general registers.

Programming Notes:

1. When a program interrupt level is entered, the work registers must be initialized as required, with the exception of register 0. Register 0 contains the IAR, and is, of course, always correct at the entry to the interrupt.

2. In general, when a level is exited, the work registers are restored automatically to the values they contained before the interrupt.

3. The other groups of registers (corresponding to the non-active program levels) cannot be accessed via the usual instructions. They must be accessed via the CCU Register In/Out instructions using the following addresses:

General register group (in local storage)	External (and LS) hexadecimal addresses
0 (program level 2)	00 through 07
1 (program level 3)	08 through 0F
2 (program level 4)	10 through 17
3 (program level 5)	18 through 1F
4 (program level 1)	20 through 27

Local Storage Map

LS Addr	Register Functions	Accessed by:
00 - 07	General Register Group 0 (Interrupt Level 2)	IN/OUT 00-07
08 - 0F	General Register Group 1 (Interrupt Level 3)	IN/OUT 08-0F
10 - 17	General Register Group 2 (Interrupt Level 4)	IN/OUT 10-17
18 - 1F	General Register Group 3 (Interrupt Level 5)	IN/OUT 18-1F
20 - 27	General Register Group 4 (Interrupt Level 1)	IN/OUT 20-27
28 - 2F	Invalid	
30	Cycle Steal Address Pointer Register - CA 1	IN/OUT 30
31	" 2	IN/OUT 31
32	" 3	IN/OUT 32
33	" 4	IN/OUT 33
34	" 5	IN/OUT 34
35	" 6	IN/OUT 35
36 - 3E	Pointer Registers 6 through E	IN/OUT 36-3E
3F	Communication Scanner CS Address Pointer Reg.	IN/OUT 3F
40	Interrupt start address - Level 1	IN/OUT 40
41	" " 2	IN/OUT 41
42	" " 3	IN/OUT 42
43	" " 4	IN/OUT 43
44	Byte Operations Base Register	IN/OUT 44
45	Halfword Operations Base Register	IN/OUT 45
46	Fullword Operations Base Register	IN/OUT 46
47	(reserved)	
48	IOH TA substitution register	IN/OUT 48
49 - 4F	Invalid	
50 - 5F	Programmable registers	IN/OUT 50-5F
60 - 6F	Invalid	
70	Not used - address corresponds to a H/W register	
71	Operator address/data entry	IN 71
72	Operator function select control	IN 72
73	Not used - address corresponds to a H/W register	
74	(reserved)	
75 - 7A	Not used - addresses correspond to H/W registers	
7B	Branch trace address pointer	IN 7B
7C	Branch trace buffer count	IN 7C
7D - 7F	Not used - addresses correspond to H/W registers	

CCU INPUT/OUTPUT INSTRUCTIONS

Most CCU input/output instructions address specific registers, although some only execute a function without using the contents of a register. In the text below, to avoid dispersion, the contents of the register is described in detail after the instruction which addresses it.

Type	Code	Meaning	Ext reg addr
I/O	00-27	General Registers	LS X'00'-X'27'
I/O	30-35	Channel Adapter CS Address Pointers	LS X'30'-X'35'
I/O	36-3E	Reserved Pointer Registers	LS X'36'-X'3E'
I/O	3F	Communication Scanner CS Address Pointer	LS X'3F'
I/O	40-43	Interrupt Start Addresses	LS X'40'-X'43'
I/O	44	Byte Operations Base Register	LS X'44'
I/O	45	Halfword Operations Base Register	LS X'45'
I/O	46	Fullword Operations Base Register	LS X'46'
I/O	48	IOH Address Substitution Register	LS X'48'
I/O	50-5F	Programmable Registers	LS X'50'-X'5F'
In	70	Storage Size Installed	HW IN X'70'
Out	70	Hardstop	Function (1)
In	71	Operator Address/Data Entry Register	LS X'71'
Out	71	Display Register 1	HW OUT X'71'
In	72	Operator Display/Function Select Control	LS X'72'
Out	72	Display Register 2	HW OUT X'72'
In	73	Insert Storage Protect/Address Exception Key	HW IN X'73'
Out	73	Set Storage Protect/Address Exception Key	HW OUT X'73'
In	74	Lagging Address Register	HW IN X'74'
In	75	CCW for AIO Operations	HW IN X'75'
In	76	CCU Level 1 Interrupt Requests on I/O ops.	HW IN X'76'
Out	76	Miscellaneous Control 1	HW OUT X'76'
In	77	Adapter Levels 2 and 3 Interrupt Requests	HW IN X'77'
Out	77	Miscellaneous Control 2	HW OUT X'77'
Out	78	Force ALU Checks (not available for 3720)	Function (1)
In	79	Utility	HW IN X'79'
Out	79	Utility	HW OUT X'79'
In	7A	High Res. Timer/Utilization Counter	HW IN X'7A'
Out	7A	High Res. Timer/Utilization Counter Control	HW OUT X'7A'
In	7B	Branch Trace Address Pointer	LS X'7B'
Out	7B	Set PCI Level 2	Function (1)
In	7C	Branch Trace Buffer Count	LS X'7C'
Out	7C	Set PCI Level 3	Function (1)
In	7D	CCU Hardware Check Register	HW IN X'7D'
Out	7D	Set PCI Level 4 (not available for 3720)	Function (1)
In	7E	CCU Level 1 Interrupt Requests	HW IN X'7E'
Out	7E	Set Program Interrupt Mask Bits	HW OUT X'7E'
In	7F	CCU L2, 3, or 4 Interrupt Requests	HW IN X'7F'
Out	7F	Reset Program Interrupt Mask Bits	HW OUT X'7F'

Notes:

1. These instructions perform a function.
2. Input/Output instructions not listed in this table cause the Invalid Operation bit (X'7E', byte 0, bit 4) to be set.

Input/Output X'00' through X'27' (General Registers)

A program level can access only its own group of general registers directly. It can access the general registers associated with the other program levels by means of the Input/Output X'00' through X'27' instructions. The bit assignments of these registers are, in general, not fixed, but vary with the use of the register. The exception is the first register of each group which always contains the address of the next sequential instruction in that level.

Input/Output X'28' through X'2F' (Reserved)

The eight registers addressed by these instructions are reserved. If the control program tries to read or write one of these registers, an invalid operation condition is detected, and a level 1 interrupt request is set.

Input/Output X'30' through X'35' (CA CS Address Pointers)

These registers are used as channel adapter cycle steal address pointer registers (CSARs). They are assigned as follows:

Register	Channel adapter
X'30'	1
X'31'	2
X'32'	3
X'33'	4
X'34'	5
X'35'	6

Programming Note:

The contents of the cycle steal address pointer register are unpredictable at the end of any data transfer, and should not be used to determine the exact number of bytes that were transferred. The exact number of bytes transferred may be determined via an Input X'C' instruction.

Input/Output X'36' through X'3E' (Reserved Pointer Registers)

The registers addressed by these instructions are reserved, and must not be used by the program. However, they are not invalid, and if used, do not cause the Invalid Operation bit to be set in register X'7E'.

Input/Output X'3F' (Communication Scanner CS Address Pointer)

This register is used as the communication scanner cycle steal address pointer register (CSAR).

Programming Note: This register is set automatically by a **communication scanner** before it starts a cycle steal operation. It should not normally be set by the CCU.

Input/Output X'40' through X'43' (Interrupt Start Addresses)

The start addresses of the four interrupt levels are contained in the registers numbered X'40' through X'43, and may be set by the control program via the Input/Output X'40' through X'43' instructions. The Output instruction is used to load a register, and the Input instruction is used to examine it. The bit structure is of the normal 3-byte address type (bytes X, 0, and 1).

Input/Output X'44' (Byte Operations Base Register)

The register addressed by these instructions is used by the 'Insert Character' and 'Store Character' instructions when the base register field (B-field) is defined as '0'. The contents of external register X'44' is then used as the base register instead of register 0.

Input/Output X'45' (Halfword Operations Base Register)

The register addressed by these instructions is used by the 'Load Halfword' and 'Store Halfword' instructions when the base register field (B-field) is defined as '0'. The contents of external register X'45' is then used as the base register instead of register 0.

Input/Output X'46' (Fullword Operations Base Register)

The register addressed by these instructions is used by the 'Load' and 'Store' instructions when the base register field (B-field) is defined as '0'. The contents of external register X'46' is then used as the base register instead of register 0.

Input/Output X'48' (IOH Address Substitution Register)

The register addressed by these instructions is used by the 'IOH' instruction when the R2 field is defined as '0'. The contents of external register X'48' is then used to define the address of the external adapter instead of the contents of register 0.

Input/Output X'49' through X'4F' (Reserved)

The registers addressed by these instructions are reserved. If the control program tries to read or write one of these registers, an invalid operation condition is detected, and a level 1 interrupt request is set.

Input/Output X'50' through X'5F' (Programmable Registers)

The registers addressed by these instructions have no specific functions and are available for use by the program.

Input/Output X'60' through X'6F' (Reserved)

The registers addressed by these instructions are reserved. If the control program tries to read or write one of these registers, an invalid operation condition is detected, and a level 1 interrupt request is set.

Input X'70' (Storage Size Installed)

This instruction causes the register specified by R to be loaded with a bit combination that indicates the amount of storage installed in the controller.

The register bits are set as shown in the table below:

Byte	Bit	Meaning
0	0	0 (For 3725 only. For 3720/3721 see below)
	1	1 (For 3725 only. For 3720/3721 see below)
	2	0
	3	2048 K
	4	1024 K
	5	512 K
	6	256 K
	7	Storage not a multiple of 256K
1	0-7	0

The effect of bits 4 through 6 is additive, so that a machine with 768K would have byte 1, bit 4 off and bits 5 and 6 on.

Note that for 3720/3721 only, Byte 0, Bits 0 and 1 have the following significance:

Bits 0 1	Meaning
0 0	All cards 512k
0 1	Reserved
1 0	Reserved
1 1	Reserved

Byte 0, Bit 7 - Storage not a Multiple of 256K: this bit, when on, indicates that the storage configuration is non-standard.

Output X'70' (Hardstop)

This instruction causes the controller to enter the 'hardstop' state. In this state, program execution, program interrupts, and adapter cycle stealing are blocked. Since this instruction performs a function, the bit settings of the general register are ignored.

Note: There is no way to leave the hardstop state except via the MOSS, IPL, or external interrupt.

Input X'71' (Operator Address/Data Entry Register)

This instruction causes the register specified by R to be loaded with the contents (three bytes: X, 0, 1) of the operator address/data entry register. This data is used in control panel functions. The operator address/data entry register is itself set by the operator via the MOSS. The bits have the following meaning:

Byte	Bit	Meaning
X	2-7	Operator address/data register byte X, bits 2-7
0	0-7	Operator address/data register byte 0, bits 0-7
1	0-7	Operator address/data register byte 1, bits 0-7

Output X'71' (Display Register 1)

This instruction transfers the contents of the register specified by R (three bytes: X, 0, 1) to display register 1; at the same time, the MOSS is informed that the register must be displayed. The bits of this register have the following meaning:

Byte	Bit	Meaning
X	2-7	Display register 1 byte X, bits 2-7
0	0-7	Display register 1 byte 0, bits 0-7
1	0-7	Display register 1 byte 1, bits 0-7

Notes:

1. This instruction should not be executed more than once every 500 milliseconds. This is to allow the MOSS to control the controller correctly.

2. When a message is displayed on the MOSS display via the X'71' (or X'72') register, the keyboard is locked for the duration of the display function. For this reason, Output X'71' (or X'72') should be used with care (for example, only after one of the registers is updated) to avoid reducing the availability of the keyboard.

Input X'72' (Operator Display/Function Select Control)

This instruction causes the register specified by R to be loaded with the contents (two bytes: 0, 1) of the operator display/function select register. The operator function select register is itself set by the operator via the MOSS. The bits of this register have the following meaning:

Byte	Bit	Meaning
0	0	Function select 8
	1	Function select 9
	2	Function select 10
	3	Function select 11 (storage address)
	4	Function select 12 (register address)
	5	Function select 13
	6	Function select 14
	7	Function select 15
1	0	Function select 16
	1	Function select 1
	2	Function select 2
	3	Function select 3
	4	Function select 4
	5	Function select 5
	6	Function select 6
	7	Function select 7

Byte 0, bit 0: this bit indicates that function 8 has been selected by the MOSS operator.

Byte 0, bit 1: this bit indicates that function 9 has been selected by the MOSS operator.

Byte 0, bit 2: this bit indicates that function 10 has been selected by the MOSS operator.

Byte 0, bit 3: this bit indicates that function 11 (storage address) has been selected by the MOSS operator.

Byte 0, bit 4: this bit indicates that function 12 (register address) has been selected by the MOSS operator.

Byte 0, bit 5: this bit indicates that function 13 has been selected by the MOSS operator.

Byte 0, bit 6: this bit indicates that function 14 has been selected by the MOSS operator.

Byte 0, bit 7: this bit indicates that function 15 has been selected by the MOSS operator.

Byte 1, bit 0: this bit indicates that function 16 has been selected by the MOSS operator.

Byte 1, bit 1: this bit indicates that function 1 has been selected by the MOSS operator.

Byte 1, bit 2: this bit indicates that function 2 has been selected by the MOSS operator.

Byte 1, bit 3: this bit indicates that function 3 has been selected by the MOSS operator.

Byte 1, bit 4: this bit indicates that function 4 has been selected by the MOSS operator.

Byte 1, bit 5: this bit indicates that function 5 has been selected by the MOSS operator.

Byte 1, bit 6: this bit indicates that function 6 has been selected by the MOSS operator.

Byte 1, bit 7: this bit indicates that function 7 has been selected by the MOSS operator.

Output X'72' (Display Register 2)

This instruction transfers the contents of the register specified by R (three bytes: X, 0, 1) to display register 2; at the same time, the MOSS is informed that the register must be displayed. The bits of this register have the following meaning:

Byte	Bit	Meaning
X	2-7	Display register 2 byte X, bits 2-7
0	0-7	Display register 2 byte 0, bits 0-7
1	0-7	Display register 2 byte 1, bits 0-7

Notes:

1. This instruction should not be executed more than once every 500 milliseconds. This is to allow the MOSS to control the comtroller correctly.

2. When a message is displayed on the MOSS display via the X'72' (or X'71') register, the keyboard is locked for the duration of the display function. For this reason, Output X'72' (or X'71') should be used with care (for example, only after one of the registers is updated) to avoid reducing the availability of the keyboard.

Input X'73' (Insert Storage Protect/Address Exception Key)

This instruction is associated with storage protection. It causes the key that was addressed by the last Output X'73' (Set Key) instruction to be loaded into byte 1, bits 5-7 of the register specified by R as shown in the table below:

Byte	Bit	Meaning
0	0-7	(not used)
1	0	(not used)
	1	(not used)
	2	(not used)
	3	(not used)
	4	(not used)
	5	Key Bit 0
	6	Key Bit 1
	7	Key Bit 2

Programming Notes:

1. If the last Output X'73' instruction addressed a storage key, an exception key, or a read-only key, the next Input X'73' instruction will read back this key, **even though the key was not set** (byte 1, bit 4 off).

2. If the last Output X'73' instruction addressed a user protect key, the next Input X'73' instruction reads back the key that was set by the last Output X'73' instruction that **changed** the user protect key. All other Output X'73' instructions have no effect.

Output X'73' (Set Storage Protect/Address Exception Key)

This instruction is associated with storage protection. Note that the 3725 and 3720 have different storage modularity (4096 bytes for the 3725, 256K bytes for the 3720). If this modularity is not respected, then this instruction will return 'no operation'. The bits of this register have the following meaning:

Byte	Bit	Meaning
X	2	Storage key address bit 0
	3	Storage key address bit 1
	4	Storage key address bit 2
	5	Storage key address bit 3
	6	Storage key address bit 4
	7	Storage key address bit 5
0	0	Storage key address bit 6
	1	Storage key address bit 7
	2	Storage key address bit 8
	3	Storage key address bit 9/user key address bit 0
	4	Storage key address bit 10/user key address bit 1
	5	User key address bit 2
	6	User key address bit 3
	7	User key address bit 4
1	0	(not used)
	1	Enable storage protect/address exception
	2	Key type bit 0
	3	Key type bit 1
	4	Modify key value
	5	Key bit 0
	6	Key bit 1
	7	Key bit 2

Byte X, Bits 2 through 7, and Byte 0, Bits 0 through 7 - Address: these bits contain the address of the key to be set. The storage, address exception, and read-only keys require a 14-bit address, and the user key a 5-bit address. The use of these addresses is described below under the heading 'CCU Special Topics', section 'Storage Protect'.

Byte 1, Bit 1 - Enable Storage Protect/Address Exception: this bit, when on, enables the storage protect/address exception mechanism. Once enabled, there is no way to disable storage protect/address exception except via a power off, or a power on reset.

Byte 1, Bits 2 and 3 - Key Type: these bits select the type of key as shown in the following table:

Byte 1 bit		
2	3	Meaning
0	0	User protect key
0	1	Storage key
1	0	Exception key
1	1	Read-only key

Byte 1, Bit 4 - Modify Key Value: this bit, when on, indicates that the addressed key is to be set with the key bits contained in byte 1, bits 5-7. If the bit is off, the addressed key is not set.

Byte 1, Bits 5 through 7 - Key Bits: these bits indicate the key that is to be set into the addressed location.

Input X'74' (Lagging Address Register)

This instruction causes the register specified by R to be loaded with the contents (three bytes: X, 0, 1) of the lagging address register (LAR).

The LAR is essentially a 'location came from' register. When displayed by the operator or by the program, it contains the address of the last instruction that was being executed prior to the current instruction (if any). The LAR is loaded from the instruction address register (IAR) at the beginning of each instruction.

The program may load the contents of the LAR into a general register by executing this instruction. From there, the program may use the LAR directly, or it may display it on the control panel by using the general register as input to the display register.

Programming Note

The address contained in the LAR is that of the last instruction executed before the Input X'74'. To preserve the contents of the LAR after a level 1 interrupt, the first instruction executed after entering level 1 should be an Input X'74' instruction.

Contents of the LAR after an Unusual Condition

In normal operation, the LAR operates as described above. However, certain check conditions and control panel operations may give a different result in the LAR:

1. **Invalid Op Code Check:** the LAR contains the address of the last instruction executed before the one that caused the check.

2. **Input/Output Check at Level 5:** the LAR contains the address of the last instruction executed before the one that caused the check.

3. **Storage Protect Check on Instruction Fetch:** the LAR contains the address of the last instruction executed before the one that caused the check.

4. **Address Exception Check on Instruction Fetch:** the LAR contains the address of the last instruction executed before the one that caused the check.

5. **Storage Protect Check on Store or Load Instruction:** the LAR contains the address of the instruction that caused the check, or this address incremented by 2.

6. **Address Exception Check on Store or Load Instruction:** the LAR contains the address of the instruction that caused the check, or this address incremented by 2.

7. **Adapter Interconnection Check on IOH or IOHI Instruction:** the LAR contains the address of the IOH or IOHI instruction that caused the check.

8. **Adapter Check:** the contents of the LAR is not predictable.

9. **Adapter Interconnection Check during Adapter Cycle Steal:** the contents of the LAR is not predictable.

10. **I-Fetch Address Compare Stop/Interrupt:** all instructions except IOH/IOHI are executed **before** the address compare stop/interrupt occurs. For the 3725 the LAR contains the address of the instruction that was executed before the one that caused the address compare. For the 3720 the LAR contains the address of the instruction that caused the address compare.

    Note: IOH/IOHI instructions **may or may not be executed** before the address compare stop/interrupt. Because of the impossibility of knowing whether or not the IOH/IOHI instruction has been executed, it is recommended not to set the compare address to either of these two instructions.

11. **Instruction Access to Storage Address Compare Stop/Interrupt:** the LAR contains the address of the instruction that loaded from or stored into the indicated storage location.

12. **Instruction Step Mode:** the LAR contains the address of the last instruction executed.

13. **Program Stop Mode:** the LAR contains the address of the last instruction executed.

Input X'75' (CCW for AIO Operations)

This instruction causes the register specified by R to be loaded with certain bits of the CCW when the current adapter-initiated operation (AIO) operation causes an error condition which stops the AIO on the I/O bus. This allows the control program and the MOSS to determine which adapter caused the error condition. The control program cannot write to this register. If the control program tries to write to this register, an invalid operation condition is detected, and a level 1 interrupt request is set. The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	CCW Bit 5 (0 = CA AIO, 1 = scanner AIO)
	1	CCW Bit 11 (pointer no./scanner address bit 0)
	2	CCW Bit 12 (pointer no./scanner address bit 1)
	3	CCW Bit 13 (pointer no./scanner address bit 2)
	4	CCW Bit 14 (pointer no./scanner address bit 3)
	5	(not used)
	6	(not used)
	7	(not used)
1	0-7	(not used)

If byte 0, bit 0 = 0, the CCW has come from the channel adapter, and byte 0, bits 1 through 4 contain a pointer number indicating one of the six channel adapter cycle steal address pointer registers:

Register	Channel adapter
X'30'	1
X'31'	2
X'32'	3
X'33'	4
X'34'	5
X'35'	6

If byte 0, bit 0 = 1, the CCW has come from the communication scanner, and byte 0, bits 1 through 4 contain the address of the scanner. In this case, register X'3F' (Cycle Steal Address Register) is used as the communication scanner cycle steal (common) address pointer register.

Input X'76' (CCU Level 1 Interrupt Requests on I/O Operations)

The register addressed by this instruction contains information identifying the source of a level 1 interrupt request. The bits of this register have the following meaning:

Byte	Bit	Meaning
0	0	Addressing exception during I/O operations
	1	Storage protection check during I/O operations
	2	Invalid CCW during I/O operations
	3	(not used)
	4	Time out condition
	5	Bus in parity check
	6	Adapter initiated operation
	7	MOSS initiated operation
1	0-7	(not used)

Note: If either or both of bits 4 and 5 of byte 0 is on, bits 0 through 3 of byte 0 have a special meaning for maintenance purposes, and should normally be ignored. For the special meaning of these bits, refer to the maintenance documentation.

Output X'76' (Miscellaneous Control 1)

The register addressed by this instruction contains miscellaneous control information, used mainly to set/reset interrupt requests. The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	Reset errors detected during I/O operations
	1	(not used)
	2	(not used)
	3	Control program to MOSS request
	4	Control program to MOSS response
	5	(not used)
	6	(not used)
	7	(not used)
1	0-7	(not used)

Byte 0, Bit 0 - Reset Errors Detected during I/O this bit, when on, resets all CCU errors that were detected during I/O operations.

Byte 0, Bit 3 - Control Program to MOSS Request: this bit, when on, raises a MOSS interrupt to inform the MOSS that the control program requires the execution of a MOSS function as defined in the CCU to MOSS Request Control Block (Mailbox) in CCU storage.

Byte 0, Bit 4 - Control Program to MOSS Response: this bit, when on, raises a MOSS interrupt to inform the MOSS that the control program has executed a request from the MOSS, and that the result is available in the CCU to MOSS Response Control Block (Mailbox) in CCU storage.

Input X'77' (Adapter Levels 2 and 3 Interrupt Requests)

The register addressed by this instruction contains information about the source of interrupts at levels 2 and 3. The bits of this register have the following meaning:

Byte	Bit	Meaning
0	0	(not used)
	1	Scanner level 2 interrupt
	2	(not used)
	3	(not used)
	4	(not used)
	5	(not used)
	6	(not used)
	7	(not used)
1	0	Level 3 channel adapter interrupt
	1	(not used)
	2	(not used)
	3	(not used)
	4	(not used)
	5	(not used)
	6	(not used)
	7	(not used)

Byte 0, Bit 1 - Scanner Level 2 Interrupt: this bit, when on, indicates that one of the communication scanners has raised a level 2 interrupt.

Byte 1, Bit 0 - Level 3 Channel Adapter Interrupt: this bit, when on, indicates that one of the channel adapters has raised a level 3 interrupt.

Programming Note: These interrupts are cleared at the adapter by means of the appropriate IOH/IOHI instruction. After clearing the interrupt in this way, a minimum of eight CCU cycles must be allowed before executing an EXIT or an Input X'77' or X'7E' instruction.

Output X'77' (Miscellaneous Control 2)

The register addressed by this instruction contains miscellaneous control information, used mainly to set/reset interrupt requests. The bits of this register have the following meaning:

Byte	Bit	Meaning
0	0	Reset IPL level 1 interrupt
	1	Reset CCU hardware checks
	2	Reset MOSS panel interrupt request level 3
	3	Reset MOSS diagnostic interrupt request level 3
	4	Reset MOSS service interrupt request level 4
	5	Reset MOSS service interrupt response level 4
	6	(not used)
	7	Reset program controlled interrupt level 2
1	0	Reset MOSS inoperative level 1 interrupt
	1	Reset interval timer level 3 interrupt
	2	Reset program controlled interrupt level 3
	3	Reset MOSS diagnostic interrupt request level 2
	4	Reset address compare level 1 interrupt
	5	Reset program errors
	6	Reset program controlled interrupt level 4
	7	Reset supervisor call level 4 interrupt

Byte 0, Bit 0 - Reset IPL Level 1 Interrupt: this bit, when on, resets an IPL level 1 interrupt.

Byte 0, Bit 1 - Reset CCU Hardware Checks: this bit, when on, causes all CCU hardware checks to be reset.

Byte 0, Bit 2 - Reset MOSS Panel Interrupt Request Level 3: this bit, when on, resets the MOSS panel interrupt request level 3.

Byte 0, Bit 3 - Reset MOSS Diagnostic Interrupt Request Level 3: this bit, when on, resets the MOSS diagnostic interrupt request level 3.

Byte 0, Bit 4 - Reset MOSS Service Interrupt Request Level 4: this bit, when on, resets the MOSS service interrupt request level 4.

Byte 0, Bit 5 - Reset MOSS Service Interrupt Response Level 4: this bit, when on, resets the MOSS service interrupt response level 4.

Byte 0, Bit 7 - Reset Program Controlled Interrupt Level 2: this bit, when on, resets the program controlled interrupt at level 2.

Byte 1, Bit 0 - Reset MOSS Inoperative Level 1 Interrupt: this bit, when on, resets the MOSS inoperative interrupt request at level 1.

Byte 1, Bit 1 - Reset Interval Timer Level 3 Interrupt: this bit, when on, resets the interval timer level 3 interrupt.

Byte 1, Bit 2 - Reset Program Controlled Interrupt Level 3: this bit, when on, resets the program controlled interrupt at level 3.

Byte 1, Bit 3 - Reset MOSS Diagnostic Interrupt Request Level 2: this bit, when on, resets the MOSS diagnostic interrupt request level 2.

Byte 1, Bit 4 - Reset Address Compare Level 1 Interrupt: this bit, when on, resets the address compare interrupt request level 1.

Byte 1, Bit 5 - Reset Program Errors: this bit, when on, resets all program errors.

Byte 1, Bit 6 - Reset Program Controlled Interrupt Level 4: this bit, when on, resets the program controlled interrupt at level 4.

Byte 1, Bit 7 - Reset Service Level 4 Interrupt: this bit, when on, resets the supervisor call interrupt at level 4.

Output X'78' (Force ALU Checks)

This instruction provides the means of testing the ALU compare circuit under diagnostic control. It causes one of the two redundant ALUs to be taken out of circuit, thus forcing a bad parity on the data at the output of the ALU. As this instruction performs a function, the bit settings of the register are not used.

For the 3720 family this instruction does not exist and will cause 'no operation' if used.

Input X'79' (Utility)

This instruction causes the register specified by R to be loaded with information indicating:

- The state of the program level 5 C and Z condition latches.
- The last program level that was active before a level 1 interrupt.

The bits of this register have the following meaning:

Byte	Bit	Meaning
0	0	(not used)
	1	(not used)
	2	(not used)
	3	(not used)
	4	(not used)
	5	(not used)
	6	Program level 5 C latch
	7	Program level 5 Z latch
1	0	Program level 2 interrupted by level 1 (note)
	1	Program level 3 interrupted by level 1 (note)
	2	Program level 4 interrupted by level 1 (note)
	3	Program level 5 interrupted by level 1 (note)
	4	(not used)
	5	(not used)
	6	(not used)
	7	(not used)

Note: One only of these bits is set if the instruction is executed in program level 1. If the instruction is executed in any other level, or in level 1 if it is reentered immediately after a level 1 exit, these four bits are set to 0.

Byte 0, bit 6 - Program Level 5 C Latch: this bit indicates that the 'C' condition latch for program level 5 is on.

Byte 0, bit 7 - Program Level 5 Z Latch: this bit indicates that the 'Z' condition latch for program level 5 is on.

Byte 1, bit 0 - Program Level 2 Interrupted by Program Level 1: this bit indicates that program level 2 was interrupted by program level 1 (see note below).

Byte 1, bit 1 - Program Level 3 Interrupted by Program Level 1: this bit indicates that program level 3 was interrupted by program level 1 (see note below).

Byte 1, bit 2 - Program Level 4 Interrupted by Program Level 1: this bit indicates that program level 4 was interrupted by program level 1 (see note below).

Byte 1, bit 3 - Program Level 5 Interrupted by Program Level 1: this bit indicates that program level 5 was interrupted by program level 1 (see note below).

Note to byte 1, bits 0-3: When an Input X'79' is executed in program level 1, one of these bits is set to 1 to indicate the program level that was running when control was passed to level 1. The other bits are set to 0.

When an Input X'79' is executed in levels other than level 1, all 4 bits are set to 0.

Programming Note: If this instruction follows an **Output** X'79' instruction, at least one CCU cycle must separate the Output and Input instructions.

Output X'79' (Utility)

This instruction is used to set and reset various hardware latches. The bits of register X'79' have the following meaning:

Byte	Bit	Meaning
0	0	(not used)
	1	(not used)
	2	Set programmed IPL request
	3	(not used)
	4	Remote power off
	5	Inhibit program level 5 C and Z latches replacement
	6	Set program level 5 C latch
	7	Set program level 5 Z latch
1	0	(not used)
	1	(not used)
	2	Set AIO stop mode
	3	Reset AIO stop mode
	4	Set bypass CCU check stop mode
	5	Reset bypass CCU check stop mode
	6	Scope sync pulse 1
	7	Scope sync pulse 2

Byte 0, Bit 2 - Set Programmed IPL Request: this bit, when on, causes an interrupt to the MOSS to indicate that IPL is required (because the program is about to ABEND).

Programming Note:

This bit only causes an interrupt to the MOSS; the control program is not stopped. If the program must stop after the IPL request, an Output X'70' (Hardstop) instruction must also be executed by the control program.

Byte 0, Bit 4 - Remote Power Off: this bit, when on, raises a line to the power subsystem causing it to power down if in the "network" power control mode (3725 and 3720 Model 1), or if the switch "remote power off" is enabled (3720 Model 2).

Byte 0, Bit 5 - Inhibit Program Level 5 C and Z Latches Replacement: this bit, when on, prevents byte 0, bits 6 and 7 from changing the state of the 5C and 5Z latches.

Byte 0, Bit 6 - Set Program Level 5 C Latch: this bit, when on, sets the program level 5 C latch to 1. For 3720, if this bit is off then program level 5 C latch is reset.

Byte 0, Bit 7 - Set Program Level 5 Z Latch: this bit, when on, sets the program level 5 Z latch to 1. For 3720, if this bit is off then program level 5 Z latch is reset.

Byte 1, Bit 2 - Set AIO Stop Mode: this bit, when on, sets the AIO stop mode, causing all AIO transfers to stop.

Byte 1, Bit 3 - Reset AIO Stop Mode: this bit, when on, resets the AIO stop mode. For 3720, if this bit is on as well as bit 2 then only bit 2 takes effect, bit 3 is ignored.

Byte 1, bit 4 - Set Bypass CCU Check Stop Mode: when this bit is set to 1, it prevents CCU hardware checks from setting a CCU hardstop and a MOSS interrupt.

Byte 1, bit 5 - Reset Bypass CCU Check Stop Mode: when this bit is set to 1, it resets the CCU check bypass mode to allow CCU hardstops and MOSS interrupts. For 3720, if this bit is on as well as bit 4 then only bit 4 takes effect, bit 5 is ignored.

Byte 1, bit 6 - Scope Sync Pulse 1: this bit, when on, generates a scope synchronization pulse at scope sync point no. 1 when the instruction is executed.

Byte 1, bit 7 - Scope Sync Pulse 2: this bit, when on, generates a scope synchronization pulse at scope sync point no. 2 when the instruction is executed.

Programming Note: If this instruction is to be followed by an **Input** X'79' instruction, at least one CCU cycle must separate the Output and Input instructions.

Input X'7A' (High Resolution Timer/Utilization Counter)

This instruction causes the register specified by R to be loaded with the contents (three bytes: X, 0, 1) of the high resolution timer/utilization counter. The correspondence between the bits is shown in the table below: The bits of this register have the following meaning:

Byte	Bit	Meaning
X	2	Timer Bit 0
	3	Timer Bit 1
	4	Timer Bit 2
	5	Timer Bit 3
	6	Timer Bit 4
	7	Timer Bit 5
0	0	Timer Bit 6
	1	Timer Bit 7
	2	Timer Bit 8
	3	Timer Bit 9
	4	Timer Bit 10
	5	Timer Bit 11
	6	Timer Bit 12
	7	Timer Bit 13
1	0	Timer Bit 14
	1	Timer Bit 15
	2	Timer Bit 16
	3	Timer Bit 17
	4	Timer Bit 18
	5	Timer Bit 19
	6	Timer Bit 20
	7	Timer Bit 21

Output X'7A' (High Resolution Timer/Utilization Counter Control)

The bits of the register addressed by this instruction have the following meaning:

Byte	Bit	Meaning
0	0	Timer/counter (1 = reset timer/enable count)
	1	High/low resolution (1 = low resolution)
	2	Timer/utilization counter (0 = timer)
	3	(not used)
	4	(not used)
	5	(not used)
	6	(not used)
	7	(not used)
1	0-7	(not used)

Byte 0, Bit 0 - Enable/Disable Timer/Counter: this bit, when on, indicates that high/low resolution timer/counter is enabled; when the bit is off, the timer/counter is disabled. For 3720 the timer/counter can only be reset.

Byte 0, Bit 1 - High/Low Resolution: this bit, when off, indicates that the timer/counter is set in the high resolution mode; when on, it indicates that the timer/counter is in the low resolution mode.

Byte 0, Bit 2 - Timer/Utilization Counter: this bit, when off, indicates that the high/low resolution timer is selected; when the bit is on, the utilization counter is selected.

Input X'7B' (Branch Trace Address Pointer)

This instruction transfers the contents of the branch trace address pointer to the register specified by the R field. The bits of this register have the following meaning:

Byte	Bit	Meaning
X	4-7	Branch trace address pointer byte X, bits 4-7
0	0-7	Branch trace address pointer byte 0, bits 0-7
1	0-7	Branch trace address pointer byte 1, bits 0-7

Output X'7B' (Set PCI Level 2)

This instruction sets a program controlled interrupt (PCI) at level 2. This allows a program level to transfer a processing requirement to a program level of different priority. A program controlled interrupt request is immediately effective. As this instruction performs a function, the bit settings of the register are not used.

Input X'7C' (Branch Trace Buffer Count)

This instruction transfers the contents of the branch trace buffer count register to the register specified by the R field. The bits of this register have the following meaning:

Byte	Bit	Meaning
0	0	Branch trace buffer count bit 0
	1	Branch trace buffer count bit 1
	2	Branch trace buffer count bit 2
	3	Branch trace buffer count bit 3
	4	Branch trace buffer count bit 4
	5	Branch trace buffer count bit 5
	6	Branch trace buffer count bit 6
	7	Branch trace buffer count bit 7
1	0	Branch trace buffer count bit 8
	1	Branch trace buffer count bit 9
	2	Branch trace buffer count bit 10
	3	Branch trace buffer count bit 11
	4	Branch trace buffer count bit 12
	5	(not used)
	6	(not used)
	7	(not used)

Note: Byte 1, bits 5 through 7 are ignored as the actual length of the buffer is always a multiple of 8 bytes.

Output X'7C' (Set PCI Level 3)

This instruction sets a program controlled interrupt (PCI) at level 3. This allows a program level to transfer a processing requirement to a program level of different priority. A program controlled interrupt request is immediately effective. As this instruction performs a function, the bit settings of the register are not used.

Input X'7D' (CCU Hardware Check Register)

This instruction causes the register specified by R to be loaded with the contents of the CCU hardware check register (two bytes). For the meaning of the bits of this register, refer to the maintenance documentation. (This instruction returns X'00' for 3720).

Output X'7D' (Set PCI Level 4)

This instruction sets a program controlled interrupt (PCI) at level 4. This allows a program level to transfer a processing requirement to a program level of different priority. A program controlled interrupt request is immediately effective. As this instruction performs a function, the bit settings of the register are not used.

Input X'7E' (CCU Level 1 Interrupt Requests)

This instruction sets the bits in the register specified by R to indicate which type of interrupt request level 1 is set. The register also includes the CCU hardware error and adapter error summary bits. The bits of this register have the following meaning:

Byte	Bit	Meaning
0	0	MOSS inoperative
	1	CCU hardware error summary
	2	(not used)
	3	Level 5 I/O error
	4	Invalid operation
	5	Adapter level 1 interrupt request
	6	(not used)
	7	CCU level 1 interrupts during I/O summary
1	0	Address compare level 1 interrupt
	1	Addressing exception on instruction fetch
	2	Storage protect exception on instruction fetch
	3	Addressing exception on program execution
	4	Storage protect exception on program execution
	5	(not used)
	6	IPL level 1 interrupt
	7	(not used)

Byte 0, Bit 0 - MOSS Inoperative: this bit, when on, indicates that the MOSS is inoperative.

Byte 0, Bit 1 - CCU Hardware Error Summary: this bit, when on, indicates that a bit has been set in register X'7D' (CCU hardware check register). The bit by itself does not cause a level 1 interrupt. Register X'7D' may be examined via the Input X'7D' instruction.

Note: This bit may also be set if a program error (such as an Invalid Operation) occurs whilst in level 1.

Byte 0, Bit 3 - Level 5 I/O Error: this bit, when on, indicates that the program attempted to execute an I/O instruction while running in level 5.

Byte 0, Bit 4 - Invalid Operation: this bit, when on, indicates that the program has attempted to execute an invalid operation code, or an I/O instruction to an invalid external register address.

Byte 0, Bit 5 - Adapter Level 1 Interrupt Request: this bit, when on, indicates that one of the channel adapters or communication scanners has raised a level 1 interrupt.

Byte 0, Bit 7 - CCU Level 1 Interrupts during I/O Summary: this bit, when on, indicates that one or more bits have been set in register X'76' (CCU level 1 interrupt on I/O operations) to indicate the cause of the level 1 interrupt. Register X'76' may be examined via the Input X'76' instruction.

Byte 1, Bit 0 - Address Compare Level 1 Interrupt: this bit, when on, indicates that an address compare has occurred.

Byte 1, Bit 1 - Addressing Exception on Instruction Fetch: this bit, when on, indicates that an addressing exception occurred during instruction fetch.

Byte 1, Bit 2 - Storage Protect Exception on Instruction Fetch: this bit, when on, indicates that a storage protection violation occurred during instruction fetch.

Byte 1, Bit 3 - Addressing Exception on Program Execution: this bit, when on, indicates that an addressing exception occurred during instruction execution.

Byte 1, Bit 4 - Storage Protect Exception on Program Execution: this bit, when on, indicates that a storage protection violation during instruction execution.

Byte 1, Bit 6 - IPL Level 1 Interrupt: this bit, when on, indicates that the MOSS has raised a level 1 interrupt request to force an IPL from the MOSS.

Output X'7E' (Set Program Interrupt Mask Bits)

This instruction is used to set the program level interrupt mask. When a mask bit is on, interrupt requests for the corresponding program level are ignored. When the mask bit for program level 5 is on, program execution at that level is suspended. The mask bits have the following meaning:

Byte	Bit	Meaning
0	0-7	(not used)
1	0	(not used)
	1	Mask adapter program level 1 requests
	2	Mask program level 2 requests
	3	Mask program level 3 requests
	4	Mask program level 4 requests
	5	Mask program level 5 execution
	6	(not used)
	7	(not used)

Input X'7F' (CCU L2, 3, or 4 Interrupt Requests)

This instruction sets the bits of the register specified by R to indicate which level (2, 3, or 4) and type of interrupt is set. The bits of this register have the following meaning:

Byte	Bit	Meaning
0	0	Program controlled interrupt (PCI) level 2
	1	MOSS diagnostic interrupt request level 2
	2	MOSS diagnostic interrupt request level 3
	3	MOSS service interrupt request level 4
	4	MOSS service interrupt response level 4
	5	(not used)
	6	CE/operator interrupt request level 3
	7	Program controlled interrupt (PCI) level 4
1	0	(not used)
	1	(not used)
	2	(not used)
	3	(not used)
	4	(not used)
	5	Interval timer interrupt request level 3
	6	Program controlled interrupt (PCI) level 3
	7	Supervisor Call level 4

Byte 0, Bit 0 - Program Controlled Interrupt (PCI) Level 2: this bit, when on, indicates that a program controlled interrupt has occurred at level 2.

Byte 0, Bit 1 - MOSS Diagnostic Interrupt Request Level 2: this bit, when on, indicates that the MOSS has set a level 2 interrupt to the CCU for diagnostic purposes. The interrupt request may be reset by executing an Output X'77' instruction with the 'Reset MOSS Diagnostic Interrupt Request Level 2' (byte 1, bit 3).

Byte 0, Bit 2 - MOSS Diagnostic Interrupt Request Level 3: this bit, when on, indicates that the MOSS has set a level 3 interrupt to the CCU for diagnostic purposes. The interrupt request may be reset by executing an Output X'77' instruction with the 'Reset MOSS Diagnostic Interrupt Request Level 3' (byte 0, bit 3).

Byte 0, Bit 3 - MOSS Service Interrupt Request Level 4: this bit, when on, indicates that the MOSS has requested a level 4 interrupt to request the control program to execute a function defined in the CCU Request Control Block (Mailbox) in CCU storage. The interrupt request may be reset by executing an Output X'77' instruction with the 'Reset MOSS Service Interrupt Request Level 4' (byte 0, bit 4).

Byte 0, Bit 4 - MOSS Service Interrupt Response Level 4: this bit, when on, indicates that the MOSS has requested a level 4 interrupt to inform the control program that a MOSS function requested by the CCU has been executed, and that the response is available in the CCU Response Control Block (Mailbox) in CCU storage. The interrupt request may be reset by executing on Output X'77' instruction with the 'Reset MOSS Service Interrupt Response Level 4' (byte 0, bit 5).

Byte 0, Bit 6 - CE/Operator Interrupt Request Level 3: this bit, when on, indicates that the CE or operator has requested a CCU level 3 interrupt via the MOSS. It replaces the panel interrupt button, and is used to inform the program that it must read the data entry and function select switches and/or

set the display indicators. The interrupt request may be reset by executing on Output X'77' instruction with byte 0 bit 2 (Reset MOSS Panel Interrupt Request Level 3) set to 1.

Byte 0, Bit 7 - Program Controlled Interrupt (PCI) Level 4: this bit, when on, indicates that a program controlled interrupt has occurred at level 4.

Byte 1, Bit 5 - Interval Timer Interrupt Request Level 3: this bit, when on, indicates that an interval timer interrupt has occurred at level 3.

Byte 1, Bit 6 - Program Controlled Interrupt (PCI) Level 3: this bit, when on, indicates that a program controlled interrupt has occurred at level 3.

Byte 1, Bit 7 - Supervisor Call Level 4: this bit, when on, indicates that a supervisor call request at level 5 has occurred at level 4.

Output X'7F' (Reset Program Interrupt Mask Bits)

This instruction is used to reset the program level interrupt mask. The mask bits have the following meaning:

Byte	Bit	Meaning
0	0-7	(not used)
1	0	(not used)
	1	Unmask adapter program level 1 requests
	2	Unmask program level 2 requests
	3	Unmask program level 3 requests
	4	Unmask program level 4 requests
	5	Unmask program level 5 execution
	6	(not used)
	7	(not used)

Programming Note:

If an interrupt for a particular level is pending when the mask bit is reset, the interrupt for that level takes place before the next instruction is executed.

CCU ERROR HANDLING

CCU Hardware Errors

All these errors cause a CCU hardstop and send a high priority interrupt request to the MOSS (the MOSS keeps running, unless it is stopped by the same failure). At the same time, a level 1 interrupt occurs in the CCU. Unless the CCU is running in the bypass check stop mode, this interrupt is queued but not executed.

The MOSS then executes a series of tests to collect error information, which is available to the MOSS operator.

Note: This information is not available to the control program, unless the error was forced by diagnostic routines running in the CCU.

The MOSS may now cause a re-IPL to take place.

CCU Program Errors

All program errors cause a level 1 interrupt to the CCU, unless the CCU is already running in level 1, in which case a hardstop occurs. Error information can be obtained by the program by executing an Input X'7E' (CCU Level 1 Interrupt Requests) instruction.

Programming Note:

Storage protection/addressing exception errors may be unrecoverable if they occur during program execution (load/store instruction) as in some cases, the instruction immediately following will have been executed before the interrupt to level 1.

Storage Protection

Setting Up the User Protect Key

er protect key, setting up

To set the user protect key, an Output X'73' instruction must be executed with byte 1, bits 2 and 3 set to 00. Byte 0, bits 3 through 7 contain the 5-bit user key address, and byte 1, bits 5 through 7 contain the key to be set. Byte 1, bit 4 must be on to set the key. This is shown in the figure below:

Byte	Bit	Meaning	
Ext		(not used)	
0	0	(not used)	
	1	(not used)	
	2	(not used)	
	3	User key addr	))
	4	User key addr	)
	5	User key addr	)-User protect key address
	6	User key addr	)
	7	User key addr	)
1	0	(not used)	
	1	Enable	Bit must be on to enable SP
	2	Key Type Bit 0	0)(These two bits indicate that
	3	Key Type Bit 1	0)(the user key is to be set
	4	Set Bit	1 Bit must be on to set the key
	5	Key Bit 0	)
	6	Key Bit 1	)-Key value
	7	Key Bit 2	)

Twenty-two different user key addresses are possible. They are assigned as follows:

User Key Address	User
X'00'	Channel Adapter 1
X'01'	Channel Adapter 2
X'02'	Channel Adapter 3
X'03'	Channel Adapter 4
X'04'	Channel Adapter 5
X'05'	Channel Adapter 6
X'06'	(reserved)
X'07'	(reserved)
X'08'	(reserved)
X'09'	(reserved)
X'0A'	(reserved)
X'0B'	(reserved)
X'0C'	(reserved)
X'0D'	(reserved)
X'0E'	(reserved)
X'0F'	All Communication Scanners
X'10'	(reserved)
X'11'	Program Level 1
X'12'	Program Level 2
X'13'	Program Level 3
X'14'	Program Level 4
X'15'	Program Level 5

Setting Up the Storage Key

To set the storage key, an Output X'73' instruction must be executed with byte 1, bits 2 and 3 set to 01. Byte Ext, bits 2 through 7 and byte 0, bits 0 through 4 contain the 11-bit address of the 2048-byte block, and byte 1 bits 5 through 7 contain the key to be set. Byte 1, bit 4 must be on to set the key. This is shown in the following figure:

Byte	Bit	Meaning	
Ext	2	SKA Bit 0	)))-Key address 0-5)))
	3	SKA Bit 1	
	4	SKA Bit 2	
	5	SKA Bit 3	
	6	SKA Bit 4	
	7	SKA Bit 5	
0	0	SKA bit 6	)))-Key address 6-10))
	1	SKA Bit 7	
	2	SKA Bit 8	
	3	SKA Bit 9	
	4	SKA Bit 10	
	5	0	
	6	0	
	7	0	
1	0	(not used)	
	1	Enable	Bit must be on to enable SP
	2	Key Type Bit 0	0)(These two bits indicate that
	3	Key Type Bit 1	1)(the storage key is to be set
	4	Set Bit	1 Bit must be on to set the key
	5	Key Bit 0	)
	6	Key Bit 1	)-Key value
	7	Key Bit 2	)

Setting Up the Address Exception Key

To set the address exception key, an Output X'73' instruction must be executed with byte 1, bits 2 and 3 set to 10. Byte Ext, bits 2 through 7 and byte 0, bits 0 through 4 contain the 11-bit address, and byte 1 bit 7 must be set to 0 for each block of storage that is installed, and to 1 for each block that is not installed. For the 3725 this block size is 4096 byte, for the 3720 it is 256K bytes. If the storage modularity is not respected in the Output X'73' instruction for the relevant controller, then it will return 'no operation'. Byte 1, bit 4 must be on to set the key. This is shown in the following figure:

Byte	Bit	Meaning	
Ext	2	SKA Bit 0	)))-Key address 0-5)))
	3	SKA Bit 1	
	4	SKA Bit 2	
	5	SKA Bit 3	
	6	SKA Bit 4	
	7	SKA Bit 5	
0	0	SKA bit 6	)))-Key address 6-10))
	1	SKA Bit 7	
	2	SKA Bit 8	
	3	SKA Bit 9	
	4	SKA Bit 10	
	5	0	
	6	0	
	7	0	
1	0	(not used)	
	1	Enable	Bit must be on to enable SP
	2	Key Type Bit 0	1)(These two bits indicate that
	3	Key Type Bit 1	0)(the exception key is to be set
	4	Set Bit	1 Bit must be on to set the key
	5	0	
	6	0	
	7	Not installed	Bit set for non-installed blocks

Setting Up the Read-Only Key

To set the read-only key, an Output X'73' instruction must be executed with byte 1, bits 2 and 3 set to 11. Byte Ext, bits 2 through 7 and byte 0, bits 0 through 4 contain the 11-bit address of the 2048-byte block, and byte 1 bit 7 must be set to 1 for each block of storage that is designated as read-only. Byte 1, bit 4 must be on to set the key. This is shown in the following figure:

Byte	Bit	Meaning	
Ext	2	SKA Bit 0	)))-Key address 0-5)))
	3	SKA Bit 1	
	4	SKA Bit 2	
	5	SKA Bit 3	
	6	SKA Bit 4	
	7	SKA Bit 5	
0	0	SKA bit 6	)))-Key address 6-10))
	1	SKA Bit 7	
	2	SKA Bit 8	
	3	SKA Bit 9	
	4	SKA Bit 10	
	5	0	
	6	0	
	7	0	
1	0	(not used)	
	1	Enable	Bit must be on to enable SP
	2	Key Type Bit 0	1)(These two bits indicate that
	3	Key Type Bit 1	1)(the read-only key is to be set
	4	Set Bit	1 Bit must be on to set the key
	5	0	
	6	0	
	7	Read-only bit	Bit set for read-only blocks

Time Measurement

For timing there is a general purpose timer and a utilization counter. As they use the same hardware mechanism, these two features are mutually exclusive and cannot be used simultaneously. Neither feature causes an interrupt.

The controller also includes an interrupting timer, causing an interrupt every 100 milliseconds.

High/Low Resolution Timer

The high/low resolution timer is a 22-bit counter that is incremented by the CCU clock. It may be programmed to operate in two different modes:

- High resolution: the timer provides intervals from 0 to 0.838 seconds by increments of 200 nanoseconds.
- Low resolution: the timer provides intervals from 0 to 57.2 minutes by increments of 819 microseconds.

The timer is controlled by byte 0, bits 0 through 2 of the Output X'7A' as follows:

Bit	Meaning
0	Enable/disable timer (1 = enable)
1	High/low resolution (1 = low resolution)
2	Must be 0 to select the timer

The timer may be read via the Input X'7A' instruction.

Programming Notes

1. The timer increments continuously. To make a measurement, the program must issue an Output X'7A' instruction to reset the timer and to select the correct mode. When the event being measured occurs, the program must issue an Input X'7A' instruction to obtain the value of the counter, and multiply it by the correct factor to obtain the time in seconds.

2. The timer cannot be used in single step mode.

3. Timer overflow is never signalled. It is therefore up to the user to select the high/low resolution mode and to ensure that the measurement interval fits into the maximum count of the counting mechanism:

   a. If the interval is less than 0.838 seconds, either the high or the low resolution mode may be used.

   b. If the interval is between 0.838 secords and 57.2 minutes, the low resolution mode must be used.

   c. If the interval is greater than 57.2 minutes, the high/low resolution counter should not be used. The 100-millisecond interrupting timer should be used instead.

Utilization Counter

The utilization counter is a 22-bit counter that is incremented by the system clock; it counts CCU busy time (including cycle steal). It operates in either high- or low-resolution mode:

- High resolution: the counter provides intervals from 0 to 0.838 seconds by increments of 200 nanoseconds.

- Low resolution: the counter provides intervals from 0 to 57.2 minutes by increments of 819 microseconds. It may be selected by executing the Output X'7A' instruction with byte 0, bit 2 set to 1; the counter may then be read via the Input X'7A' instruction.

Programming Notes

1. The counter increments continuously. To measure CCU busy time, the program must issue an Output X'7A' instruction to reset the counter and to select the correct mode. At the end of the measuring period, the program must issue an Input X'7A' instruction to obtain the value of the counter, and, if necessary, multiply it by the correct factor to obtain the time in seconds.

2. The counter cannot be used in single step mode.

3. Counter overflow is never signalled. It is therefore up to the user to select the high/low resolution mode and to ensure that the measurement interval fits into the maximum count of the counting mechanism:

   a. If the interval is less than 0.838 seconds, either the high or the low resolution mode may be used.

   b. If the interval is between 0.838 seconds and 57.2 minutes, the low resolution mode must be used.

   c. If the interval is greater than 57.2 minutes, the high/low resolution counter should not be used. The 100-millisecond interrupting timer should be used instead.

100-Millisecond Interval Timer (Interrupting)

The interval timer provides an interrupt request at program level 3 every 100 milliseconds. It may be used to maintain a real-time clock in storage, perform long and short I/O timeouts, and perform supervisory functions on a periodic basis. The interrupt may be reset by executing an Output X'77' instruction with the 'Reset Interval Timer Level 3 Interrupt' bit (byte 1, bit 1) set to 1.

The 100-millisecond timer interrupt is disabled for program stop or single instruction step mode.

CCU Diagnostic Facilities

The CCU has a certain number of test facilities to allow the control program to test the controller hardware.

Bypass CCU Check Stop/MOSS Interrupt

The CCU check stop/MOSS interrupt caused by a CCU hardware check may be masked by executing an Output X'79' (Utility) instruction with byte 1, bit 4 (Set Bypass CCU Check Stop Mode) set to 1. It may be un-masked by executing an Output X'79' instruction with byte 1, bit 5 (Reset Bypass CCU Check Stop Mode) set to 1.

Note: Enabling/disabling the CCU check stop/MOSS interrupt may also be controlled from the MOSS.

Inhibit Channel Adapter/Communication Scanner Level 1 Interrupt

Level 1 interrupts caused by a channel adapter or communication scanner can be masked by executing an Output X'7E' (Set Program Interrupt Mask Bits) instruction with byte 1, bit 1 (Mask Adapter Program Level 1 Requests) set to 1.

The interrupts may be unmasked by executing an Output X'7F' (Reset Program Interrupt Mask Bits) instruction with byte 1, bit 1 set to 1.

Force CCU Checks

The control program can force wrong parity on the ALU output by executing an Output X'78' instruction. This data with bad parity may be used to perform further checking by moving it about the CCU. This causes a 'no operation' on 3720.

CHAPTER 4. CHANNEL ADAPTER

The CCU is an interrupt-driven processor, and almost all processing for the channel adapter is done in response to channel adapter interrupts. This chapter is therefore organized in the following way:

- Section 1 contains basic information concerning the channel adapter.
- Section 2 describes the channel adapter interrupt system. This provides the means by which the channel adapter indicates to the CCU that it requires service.
- Section 3 describes the channel adapter I/O system. This provides the means by which the CCU responds to the channel adapter interrupts.
- Section 4 describes programming considerations for the handling of interrupts, commands, and initial status responses.
- Section 5 describes the two-processor switch.
- Section 6 describes a number of special topics.

SECTION 1. CHANNEL ADAPTER BASIC INFORMATION

The channel adapter allows the controller to be attached to the selector, block multiplexer, or byte multiplexer channels of a S/370 or similar host. Up to six channel adapters may be attached (up to 4 in the 3725 Model 2 and up to 2 in the 3720).

Modes of Operation

The channel adapter may run in either Native Subchannel mode (NSC), or in Emulation Subchannel mode (ESC). With proper programming support, the channel adapter allows the controller to operate in either NSC mode, or in ESC mode, or in both modes simultaneously.

The NSC mode is supported for all types of host channel (byte multiplex, selector, block multiplex), and allows the servicing of any number of lines up to 256 using only one unique host subchannel address. Line address decoding is handled entirely by the control program.

The ESC mode is supported for byte multiplex channels only, and allows the controller to emulate the 2701, 2702, and 2703 control units using existing host programs and subchannel addresses. A separate subchannel address is required for each line.

Notes:

1. Initial program load (IPL) must always be done in NSC mode.

2. Many of the channel adapter operations are identical in both NSC and ESC modes. Throughout this chapter, the exceptions and/or differences in operation due to NSC or ESC mode are noted by "NSC" or "ESC" at the start of the paragraph that describes the particular operation. All text that is not specially marked "NSC" or "ESC" may be assumed to apply equally to both modes of operation.

Basic Operation and Data Flow

The channel adapter receives an address and a command from the host processor and determines whether the host wants to communicate in NSC or ESC mode; the correct mode of operation is then set. The channel adapter then requests a level 3 interrupt to make this information available to the control program via Input instructions.

In PIO operation, the data coming from the host channel interface is placed in the data buffers, from where the control program must retrieve it by executing Input instructions. In AIO mode, the data from the host is placed directly into main storage. Channel End and Device End status are generated by the control program when the complete message or block of data has been received.

When the data is going **to** the host channel, the control program sends an Attention status to the channel. The host processor then initiates an initial selection sequence with a read command. The control program in PIO or AIO mode must then load the buffers with the data to be transferred. The data from the buffers can now be transferred across the channel interface. Channel End and Device End status are generated by the control program when the complete message or block of data has been sent.

Data Transfer Methods

Two methods may be used to transfer data between the channel adapter and the host channel:

- Program-initiated operation (PIO)
- Adapter-initiated operation (AIO)

Program-Initiated Operation (PIO)

With PIO, data buffering at the channel adapter interface is provided for up to four bytes of data, the buffers being serviced by programming. Program intervention is required for every four bytes. PIO is relatively slow, and should not be used on channel adapters connected to selector or block multiplexer channels. It may be used for channels attached to byte multiplexer channels in cases where performance is not a critical concern.

Adapter-Initiated Operation (AIO)

With AIO, data buffering at the channel adapter interface is provided for 16 bytes of data, the buffers being serviced by cycle stealing. Up to 255 bytes of data may be transferred by this method before program intervention becomes necessary. AIO should always be used on selector or block multiplex channels.

Controlling the Channel Adapter

The channel adapter is controlled by instructions issued by the control program. These instructions are of two types only: 'Adapter Input/Output' (IOH) and 'Adapter Input/Output Immediate' (IOHI). Using these instructions, the channel adapter registers may be examined or set, buffers may be loaded or read, and cycle stealing may be initiated.

Note: Throughout this chapter, the channel IOH and IOHI instructions are referred to as Input X'n' or Output X'n' for simplicity.

Access to a channel adapter may be obtained at program levels 3 and 1. Program level 3 is used for all routine servicing of the channel adapter. This level is entered via a level 3 interrupt, initiated by either an event occurring on the channel interface, or by a program controlled interrupt (PCI) from another program level. Program level 1 is used for servicing channel adapter error conditions. See under the heading 'Input X'D' (Channel Adapter Interrupt Check Register) for more details.

Programming Note

All IOH/IOHI instructions are **privileged**, that is, any attempt to execute them in background program level 5 causes an input/output check, and a level 1 interrupt.

Channel Adapter States

The channel adapter may be in one of five states:

- Ready state
- Initial selection state

- Data transfer state
- Status transfer state
- Disabled state

Ready State

In the ready state, the channel adapter may accept instructions, but is not in one of the three active states (initial selection, data transfer, status transfer).

Initial Selection State

The channel adapter enters the initial selection state when an initial selection is started by the host processor. The channel adapter continually monitors its channel interface for its assigned addresses. When an address is detected, the channel adapter enters the initial selection state, and proceeds with the initial selection. If a standard command is sent on initial selection and is received without error (correct parity), an initial status of all zeros is returned to the channel, unless the command is I/O No-Op or Test I/O.

Note: Non-standard commands sent to a block multiplex or selector channel receive an initial status of channel end.

During initial selection, the I/O device address and the channel command are stored in the initial selection address and command register. The initial selection hardware then causes a level 3 interrupt, and control is passed to the level 3 interrupt program.

Data Transfer State

The channel adapter enters the data transfer state when the control program initiates a data transfer sequence. Data is transferred across the interface from the host channel to the channel adapter, or from the channel adapter to the host, by hardware. When the data transfer is ended, the channel adapter hardware calls the control program with a level 3 interrupt request.

Status Transfer State

The channel adapter enters the status transfer state when the control program initiates a status transfer sequence. During this sequence, the status byte is transferred to the host. When the status transfer is ended, the channel adapter hardware calls the control program with a level 3 interrupt request.

Disabled State

The channel adapter is in the disabled state when it is not enabled by the control program or by the panel switches. In the disable state, the selection signals are propagated to the next adapter on the channel.

Channel Adapter Device Addresses

Channel adapter device addresses are required on two separate occasions:

- At initial selection, the channel adapter must be able to recognize the device address presented to it
- On a byte multiplex channel, the channel adapter must present a valid device address to the channel before it can transfer data or status information.

Channel Adapter Device Addresses for Initial Selection

The address byte presented by the channel during initial selection must have correct parity, or the channel adapter will not decode the device address. If the parity is correct, the channel adapter will recognize a device address or addresses determined by plug options wired by the customer engineer from information supplied by the user.

NSC: the NSC device address can be assigned any value in the range 0 through 255. If the two-processor switch is installed on a channel, the two NSC interfaces (A and B) are assigned separately, and may be either the same or different. As the NSC uses only one subchannel address, the **line** address must be transferred from the host in the form of data. The location and the format of the terminal addresses must be coordinated between the host access method and the control program.

ESC: the ESC device addresses must form a group of contiguous addresses. The lowest address in the group may be set to 0, or to any multiple of 16 from 16 through 240. The highest address in the group may be set to one of the values 4n-1, where n = 1 through 64, that is, from 3 through 255 by steps of 4. If the two-processor switch is installed, the addresses are the same for interfaces A and B.

Programming Notes:

1. If emulation is not required, the machine can be wired at installation time so that the lowest ESC address is higher than the highest address. In this way, no ESC addresses are recognized.

2. The address assigned for the NSC may be one of the addresses in the range assigned to the ESC. The NSC address has priority, and the address is lost to the ESC.

3. After power on, the channel adapter does not immediately recognize the ESC addresses, even if the interface is enabled. To make the ESC addresses operational, the program must issue an Output X'7' instruction to set the 'Set ESC Operational' bit (byte 1, bit 5) to 1. If the host sends Start I/O to an ESC address before this bit is set, the resulting condition code for the host instruction is set to 3 (not operational).

4. An initial selection causes a channel adapter initial selection level 3 interrupt. The program may determine the I/O device address by issuing an Input X'1' instruction. Once the program has set the 'Set ESC Operational' bit as described above, all the assigned ESC addresses become operational, and the interrupt request may be caused by an initial selection sequence for any of these addresses, or for the NSC address. The program must therefore be prepared to handle initial selection sequences for all assigned operational addresses, both NSC and ESC.

Channel Adapter Device Addresses for Data/Status Transfer

When the control unit initiates a data/status transfer, it must provide the correct device address associated with the transfer.

NSC: for control unit initiated data/status transfers on the NSC, the plugged hardware address is used.

ESC: as the device address for control unit initiated data/status transfers on the ESC is variable, it must be provided by the control program. This is done by executing an Output X'3' instruction, with the data/status transfer address in byte 0. The channel adapter hardware checks only that the issued address is in the range of plugged addresses; if the address is outside this range, the address compare error bit (byte 0, bit 5) is set in register X'D', and a level 1 interrupt occurs.

Programming Notes:

1. If the address presented is incorrect, but within the plugged range, the channel adapter has no means of detecting the error; no error is signaled, but improper channel operation will occur.

2. The I/O address that was presented to the channel by the last Output X'3' instruction may be determined by executing an Input X'3' instruction. The Input X'3' instruction should only be issued in interrupt level 3.

SECTION 2. CHANNEL ADAPTER INTERRUPT REQUESTS

The channel adapter can raise interrupt requests at level 1 and at level 3.

- Level 1 interrupt requests are caused by check or error conditions.
- Level 3 interrupt requests are caused by two different conditions:
  - Initial selection interrupt requests are raised when the channel adapter receives an address and a command across the channel interface.
  - Data/status interrupt requests are raised when the channel adapter requires data or status service.

Level 1 Interrupt Requests

When an error condition is detected in the channel adapter, a level 1 interrupt occurs, and a bit is set in the Channel Adapter Level 1 Interrupt Check Bit register to indicate the type of error.

Level 3 Interrupt Requests

There are two types of interrupt request at level 3:

- Channel Adapter Initial Selection Level 3 interrupt request
- Data/Status Transfer Level 3 interrupt request

Channel Adapter Initial Selection Level 3 Interrupt Request

This type of interrupt request may be due to:

- An initial selection sequence
- A system reset sequence
- An NSC status cleared indication
- An ESC TIO status cleared indication

When an Initial Selection interrupt request occurs, the condition causing the interrupt may be determined by executing an Input X'0' (Initial Selection Control register) instruction.

Channel Adapter Data/Status Level 3 Interrupt Request

This type of interrupt request may be set by:

- The end of an inbound data transfer sequence
- The end of an outbound data transfer sequence
- The end of a status transfer sequence
- Any level 1 interrupt occurring during any one of the above three data/status transfers.

- A Suppress Out Monitor condition
- A program requested interrupt

When a Data/Status interrupt request occurs, the condition causing the interrupt may be determined by executing an Input X'2' (Data/Status Control register) instruction.

SECTION 3. CHANNEL ADAPTER INPUT/OUTPUT

The channel adapter contains a number of registers, most of which are accessible to the program via the IOH/IOHI instructions. These registers are described in detail below under the heading 'Channel Adapter IOH/IOHI Instructions - Detailed Bit Structure'. One very important group of registers used for channel adapter operations in AIO mode is physically located in the CCU, and is accessed by CCU Input and Output instructions (not IOH/IOHI):

Input/Output X'30' through X'35' - Fixed Pointer Registers

The Output instruction loads the CCU pointer address with the cycle steal data address for the channel adapter; the Input instruction may be used to read it back. The correspondence between register and channel adapter is as follows:

Register	Channel adapter
X'30'	1
X'31'	2
X'32'	3
X'33'	4
X'34'	5
X'35'	6

CHANNEL ADAPTER IOH/IOHI INSTRUCTIONS

The channel adapter IOH/IOHI instructions are used to transfer the contents of one of the general registers to one of the channel adapter registers (register X'n') or vice versa.

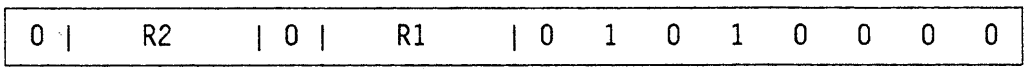
There are two types of channel adapter input/output instruction:

- Adapter Input/Output (IOH)
- Adapter Input/Output Immediate (IOHI)

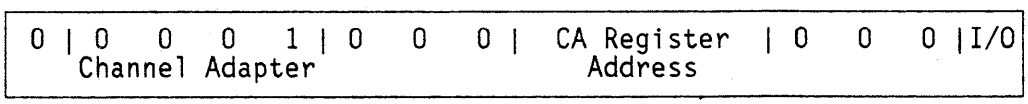
They are used by the channel adapter as follows:

Adapter Input/Output (IOH)

This instruction transfers the contents of the register specified by R1 to the channel adapter, or places information coming from the channel adapter into the register specified by R1. The adapter, the adapter command or register, and the direction of data movement are all specified by the contents of R2.



R2 must be loaded as follows:



Bits 1 through 4 (= 0001) indicate the channel adapters

Bits 5 through 7 **must** be zero.

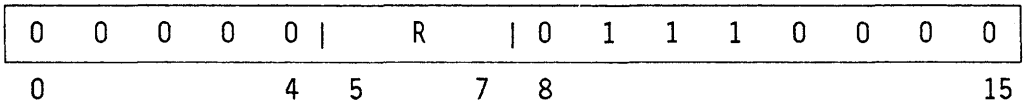
Bits 8 through 11 indicate the CA register address (X'x') to be used.

I/O = input/output bit: 0 = output, 1 = input

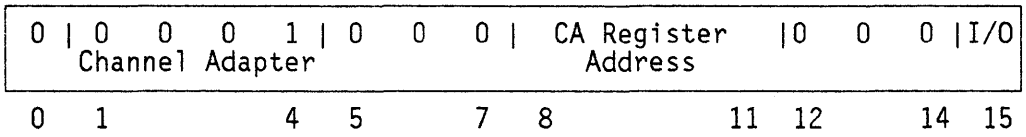
Adapter Input/Output Immediate (IOHI)

This instruction transfers the contents of the register specified by R to the channel adapter, or places information coming from the channel adapter into the register specified by R. The adapter, the adapter register, and the direction of data movement are all specified by the contents of the second halfword.

First halfword



Second halfword



Bits 1 through 4 (= 0001) indicate the channel adapters

Bits 5 through 7 **must** be zero.

Bits 8 through 11 indicate the CA register address (X'x') to be used.

I/O = input/output bit: 0 = output, 1 = input

CHANNEL ADAPTER ADDRESSING

As may be seen in the previous section, the IOH and IOHI instructions do not contain an explicitly defined address. The channel adapter is addressed indirectly via the contents of a special 3-bit register, controlled by byte 0 of the Output X'7' instruction. The table below shows the bits of the instruction used to control this 3-bit register:

Byte	Bit	Meaning
0	0	Enable auto-selection (all channel adapters)
	1	Disable auto-selection (all channel adapters)
	2	Select CA addressed by bits 4-6
	3	Execute output on CA addressed by bits 4-6
	4	Channel address bit 0)
	5	Channel address bit 1) CA address 1-6
	6	Channel address bit 2)
	7	Channel adapter reset

Bits 4 through 6 define the channel adapter as follows:

Bit 4 5 6	Channel Adapter
0 0 0	1
0 0 1	2
0 1 0	3
0 1 1	4
1 0 0	5
1 0 1	6

There are two modes of operation:

- Explicit selection by the control program.
- Selection by the auto-selection mechanism.

Channel Adapter Selection by the Control Program

This mode of channel adapter selection is controlled by byte 0, bits 2 through 6 of the Output X'7' instruction. Bits 4 through 6 contain the address of the channel adapter to be selected, and bits 2 and 3 control the operation:

- If bit 2 (Select Channel Adapter Addressed by Bits 4 through 6) is set to 1, the addressed channel adapter is selected for all subsequent channel adapter instructions until it is turned off.

  Note: The channel adapter (if any) that was selected by the auto-selection mechanism is changed.

- If bit 3 (Execute Output on Channel Adapter Addressed by Bits 4 through 6) is set to 1, the addressed channel adapter is temporarily selected for this instruction **only.** This mode is used to set up a program requested interrupt on the addressed channel via byte 1, bit 1 (Set Program Requested Interrupt) of this same Output X'2' instruction.

Note: The channel adapter (if any) that was selected by the auto-selection mechanism, or by byte 0, bit 3 is simply overridden, not changed.

Channel Adapter Selection by the Auto-Selection Mechanism

This mode of channel adapter selection is controlled by byte 0, bits 0 and 1 of the Output X'7' instruction. Bit 0 enables auto-selection and bit 1 disables it; both bits must not be on at the same. Auto-selection works as follows:

1. The control program enables auto-selection on all channel adapters by performing an Output X'7' instruction with byte 0, bit 0 set to 1.

2. When a level 3 interrupt is pending, the control program executes an Input X'F' (Channel Adapter Level 3 Interrupt Requests) instruction. If the 'Auto-Selection Complete' latch is not already set, the channel adapter with the highest priority interrupt request pending is selected; its address is available in byte 0, bits 4 through 6 of register X'F'. At the same time, the 'Auto-Selection Complete' latch is set to prevent further auto-selection until the current interrupt has been serviced and reset.

3. The control program executes the interrupt handling routine. All instructions to the channel adapter are now routed automatically to the selected channel adapter.

4. At the end of the interrupt handling routine, the control program must reset the 'Auto-Selection Complete' latch by executing one of the following instructions:

   - Output X'0' (bit configuration is ignored).
   - Output X'2' with either byte 0, bit 5 (Reset Initial Selection Interrupt), or byte 0, bit 6 (Reset Data/Status Interrupt) set to 1 as appropriate.
   - Output X'7' with byte 1, bit 3 (Reset System Reset/NSC Address Active), if the interrupt was due to a System Reset.
   - Output X'B' (bit configuration is ignored).

The order of priorities in the auto-selection mechanism is as follows:

1. Priority outbound level 3 interrupt.
2. Outbound level 3 interrupt.
3. Initial selection level 3 interrupt.
4. Inbound level 3 interrupt.
5. All other level 3 interrupts.

CHANNEL ADAPTER IOH/IOHI INSTRUCTIONS - DETAILED BIT STRUCTURE

Input X'0' (Initial Selection Control Register)

The register addressed by this instruction is set by the channel adapter hardware and contains information that identifies the event that set the channel adapter Initial Selection Level 3 interrupt. The instruction should be issued only when servicing a channel adapter initial selection level 3 interrupt request. This type of interrupt request may be set by:

1. the completion of an initial selection sequence.

2. the detection of various reset sequences.

During a normal initial selection sequence, the initial selection interrupt bit (byte 0, bit 0) is set. The remaining bits give supplementary information, or indicate certain reset and error conditions. The bits of this register have the following meaning:

Byte	Bit	Meaning
0	0	Initial selection interrupt
	1	Interface disconnect
	2	Selective reset
	3	Channel bus out check
	4	Emulation subchannel operation (ESC = 1)
	5	Initial status byte stacked
	6	Status byte cleared
	7	System reset
1	0-7	(not used)

Byte 0, Bit 0 - Initial Selection Interrupt: this bit, when on, indicates that a normal initial selection sequence has occurred. The initial status byte sent to the host processor is X'00', unless it is a command for the NSC and the channel command is non-standard; the initial status byte is then X'08', with the Channel End bit on. The I/O device address and the I/O command byte may be determined by executing an input X'1' instruction. If this bit is on, indicating a normal initial selection sequence, all other bits of this register should be off, with the possible exception of the ESC selection bit (byte 0, bit 4).

If byte 0, bit 0 is off, it indicates that the interrupt request was due to the detection of another condition as defined by the remaining bits of the register.

Programming Note:

Certain normal initial selection sequences do not cause the channel adapter initial selection level 3 interrupt to be set, and furthermore, do not set byte 0, bit 0. These sequences are as follows:

1. An initial selection sequence in which the channel adapter responds automatically with a 'Control Unit Busy' status indication (X'70' = 'Status Modifier', 'Control Unit End', and 'Busy') due to the channel adapter initial selection level 3 interrupt request having been already previously set.

2. An initial selection sequence for a channel I/O command byte X'03' ('I/O No-Op'). The initial status byte of 'Channel End' and 'Device End' is generated automatically by the channel adapter hardware.

3. A 'Test I/O' command has been sent to the NSC address when it was free of commands.

Byte 0, Bit 1 - Interface Disconnect: this bit, if on, indicates that the channel adapter level 3 interrupt request was caused by the detection of an interface disconnect sequence (Halt I/O) during initial selection. The addressed subchannel can be determined via an Input X'1' instruction.

Byte 0, Bit 2 - Selective Reset: this bit, if on, indicates that the channel adapter level 3 interrupt request was caused by the detection of a selective reset sequence during initial selection. The addressed subchannel can be determined via an Input X'1' instruction.

Note: The selective reset does not cause a reset of the channel adapter. If the unique NSC address (byte 0, bit 4 off), has received the selective reset, the program should execute an Output X'7' instruction with the 'reset system reset/NSC address active' bit (byte 1, bit 3) equal to 1. This resets the channel adapter hardware associated with the unique NSC address.

Byte 0, Bit 3 - Channel Bus Out Check: this bit, if on, indicates that the channel adapter level 3 interrupt request was caused by the detection of bad (even) parity on the I/O channel interface bus out when the channel I/O command byte was presented during initial selection. The channel adapter responds automatically with 'Unit Check' status. The addressed subchannel can be determined via an Input X'1' instruction.

Byte 0, Bit 4 - Emulation Subchannel Operation: this bit, if on, indicates that the channel adapter initial selection address and command register (X'1) contains an ESC address.

If the bit is off, the channel adapter initial selection address and command register (X'1') contains the unique NSC address.

Note: This bit is set to zero if the initial selection level 3 interrupt was due to a system reset.

Byte 0, Bit 5 - Initial Status Byte Stacked: this bit, if on, indicates that the channel adapter level 3 interrupt request was caused by the completion of an initial selection sequence in which the initial status byte presented to the channel has been stacked. The addressed subchannel and I/O command can be determined via an Input X'1' instruction.

The initial status byte stacked indication can occur in the following situations:

1. **NSC** (Byte 0, bit 4 = 0). The NSC status byte, prepared by the program, was presented to the host during an initial selection sequence, but was stacked (not accepted) by the host. As a result, a channel adapter initial selection level 3 interrupt request is set.

2. **ESC** (Byte 0, bit 4 = 1). The ESC status byte, prepared by the control program for a given subchannel, was presented to the host in response to an ESC 'Test I/O' command issued to the same subchannel. However, the status was stacked (not accepted) by the host.

Byte 0, Bit 6 - Status Byte Cleared: the interpretation of this bit depends on the state of Byte 0, bit 4:

1. **NSC** (Byte 0, bit 4 = 0). The NSC status byte, either prepared by the program in the X'6' register, or an early Channel End status that has been previously stacked, has been transferred to the host during an initial selection sequence. Thus, the NSC status byte has been cleared, and this has resulted in the setting of the channel adapter initial selection level 3 interrupt request. If the command during this initial selection sequence was 'Test I/O' (X'00'), the status is presented normally. If, however, the command was anything other than 'Test I/O', the 'Busy' bit is presented in addition to the other status bits. The control program must perform the channel adapter state and the status sent by performing the appropriate IOH. If the 'NSC address active' bit (Input X'7', byte 0, bit 5) is found to be on it means that the Device End status must be sent; if not, the program only resets the level 3 interrupt request.

   Note: Byte 0, bit 0 is **not** set.

2. **ESC** (Byte 0, bit 4 = 1). The channel adapter initial selection level 3 interrupt request results from the completion of an initial selection sequence in which a 'Test I/O' command to an ESC subchannel was serviced, and TIO status had previously been set up for that subchannel. The subchannel that was serviced, and the status that was presented, can be determined by executing an Input X'B' instruction.

   Note: Byte 0, bit 0 is **not** set.

Byte 0, Bit 7 - System Reset: This bit, when on, indicates that a system reset sequence has occurred on the channel, causing the channel adapter to reset also. This means that if this bit is on, all other bits obtained by the execution of the Input X'0' instruction are automatically 0.

Note: Since a system reset may occur at any time, all indications of previous channel sequences that have not yet been serviced are lost to the CCU program.

Output X'0' (Reset Initial Selection)

This instruction resets all the initial selection hardware latches and also the channel adapter level 3 interrupt request resulting from an initial selection sequence. As this instruction performs a function, the bit settings of the register are not used.

Note: This instruction does not reset a 'system reset' condition, nor the resulting channel adapter level 3 interrupt request.

Input X'1' (Initial Selection Address and Command Register)

The register addressed by this instruction is set by the channel adapter hardware with the address and the command received from the channel. The instruction should be issued only if a channel adapter initial selection interrupt has been set at level 3, and an Input X'0' has shown that the interrupt is due to an initial selection sequence. The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0-7	Address byte bits 0-7 (initial selection addr)
1	0-7	I/O cmd byte bits 0-7 (initial selection cmd)

Output X'1' (Initial Selection Address and Command Register)

This instruction allows the program to set register X'1' with an initial selection address and command **for diagnostic purposes only.**

Input X'2' (Data/Status Control Register)

The register addressed by this instruction is used to identify the event(s) which caused a channel adapter Data/Status level 3 interrupt. The instruction must normally be issued only for servicing a channel adapter level 3 data/status interrupt.

Note: If a system reset sequence occurs before the Input X'2' is executed, the bits which define the cause of the interrupt request, and the interrupt request itself are reset.

The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	Outbound data transfer sequence
	1	Inbound data transfer sequence
	2	Status transfer sequence
	3	Emulation subchannel operation (ESC = 1)
	4	Channel end presented
	5	Channel stop/interface disconnect
	6	Suppress out monitor interrupt
	7	Program requested interrupt
1	0	Channel bus out check
	1	Selective reset
	2	Suppress out
	3	Ending status stacked
	4	Priority outbound service
	5	Residual byte count bit 5
	6	Residual byte count bit 6
	7	Residual byte count bit 7

Byte 0, Bit 0 - Outbound Data Transfer Sequence: this bit, when on, indicates that the channel adapter data/status level 3 interrupt was caused by the ending of an outbound data transfer sequence (controller to host). If this bit is on, byte 0, bit 1 should be off. On a block multiplex channel, byte 0, bit 2 may also be on if the block is the last data block.

In PIO mode, byte 1, bits 5 through 7 indicate the residual byte count.

ESC: if the transfer was over an emulator subchannel (byte 0, bit 3 = 1), an Input X'3' instruction should be executed to determine the address of the subchannel.

Byte 0, Bit 1 - Inbound Data Transfer Sequence: this bit, when on, indicates that the channel adapter data/status level 3 interrupt was caused by the ending of an inbound data transfer sequence (host to controller). If this bit is on, byte 0, bits 0 and 2 should both be off.

In PIO mode, byte 1, bits 5 through 7 indicate the residual byte count; the data bytes transferred from the host processor may be obtained by executing the Input X'4' and X'5' instructions.

ESC: if the transfer was over an emulator subchannel (byte 0, bit 3 = 1), an Input X'3' instruction should be executed to determine the address of the subchannel.

Byte 0, Bit 2 - Status Transfer Sequence: this bit, when on, indicates that the channel adapter data/status level 3 interrupt was caused by the ending of a status transfer sequence. If this bit is on, byte 0, bit 1 should be off. Byte 0, bit 0 may be on or off.

ESC: if the transfer was over an emulator subchannel (byte 0, bit 3 = 1), an Input X'3' instruction should be executed to determine the address of the subchannel over which the transfer occurred and to obtain the status that was presented.

Byte 0, Bit 3 - Emulation Subchannel Operation:

NSC (Byte 0, Bit 3 Off)

The transfer sequence defined by byte 0, bits 0 through 2, was performed on the native subchannel using the assigned hardware address. In the case of a status transfer, the status was taken from the NSC status register (X'6').

ESC (Byte 0, Bit 3 On)

The transfer sequence defined by byte 0, bits 0 through 2, was performed on the ESC subchannel using the ESC address (and the status, in the case of a status transfer) taken from the ESC address and status register (X'3').

Byte 0, Bit 4 - Channel End Presented: this bit only applies to a native subchannel working with a block multiplex or selector channel; for all other combinations, it is not used.

Note: When this bit is on, byte 0, bit 1, or byte 0, bits 0 and 2 are also on.

When the bit is on, it indicates that the hardware has presented, or has tried to present, a 'Channel End' status to the host during a data transfer; a level 3 interrupt request is set up at the same time.

The hardware 'Channel End' is presented when the CCU program sets up an outbound (controller to host) data transfer sequence and a status transfer sequence at the same time (Output X'2', byte 0, bits 0 and 2 both on), and the data transfer to the host occurs normally.

The hardware 'Channel End' is also presented during any data transfer sequence if the host issues a 'Channel Stop' command.

Byte 0, Bit 5 - Channel Stop/Interface Disconnect: this bit, when on, indicates that during an inbound or outbound data transfer sequence, a channel stop or interface disconnect (Halt I/O) sequence has occurred (the CCU program cannot distinguish between the two). The transfer sequence is ended; the residual byte count may be greater than zero, and indicates the number of bytes that were not transferred (the residual byte count is contained in register X'2', byte 1, bits 5 through 7 for PIO operations, and in register X'C', byte 1, bits 0 through 7 for AIO operations).

The bit, when on, may also indicate that an interface disconnect (Halt I/O) occurred during a status transfer sequence.

Byte 0, Bit 6 - Suppress Out Monitor Interrupt: this bit, when on, indicates that the channel adapter data/status level 3 interrupt was set because the 'suppress out' tag line was found to be inactive after the control program had told the channel adapter to monitor for this condition. If one of the transfer bits (byte 0, bits 0 through 2) is on, this bit should be 0 for correct operation. The program signals to the channel adapter to monitor for the inactive condition of 'suppress out' by executing an Output X'7'

instruction with the 'Set Suppress Out Monitor Interrupt' bit (Byte 1, bit 0) set to 1.

Byte 0, Bit 7 - Program Requested Interrupt: this bit, when on, indicates that the channel adapter data/status level 3 interrupt was set because the program requested an interrupt at level 3 by executing an Output X'7' instruction with byte 1, bit 1 set 1. An Output X'2' instruction should be executed to reset this bit and the resulting channel adapter data/status level 3 interrupt request. **Byte 1, Bit 0 - Channel Bus Out Check:** this bit, when on, indicates that during an inbound (from host) data transfer sequence, a bad (even) parity condition was detected on bus out during the transfer of a data byte and that the transfer sequence was terminated. The number of bytes transferred prior to the bus out check may be found by examining the residual byte count (the residual byte count is contained in register X'2', byte 1, bits 5 through 7 for PIO operations, and in register X'C', byte 1, bits 0 through 7 for AIO operations).

Byte 1, Bit 1 - Selective Reset: this bit, when on, indicates that a selective reset sequence occurred during the transfer.

Byte 1, Bit 2 - Suppress Out: this bit, when on, indicates that the 'Suppress Out' tag line on the channel was active at the time that the Input X'2' instruction was executed.

Byte 1, Bit 3 - Ending Status Stacked: this bit, when on, indicates that during a final status transfer sequence, the status byte was not accepted by the channel (stacked).

ESC

The status byte that was presented may be examined by executing an Input X'3' instruction.

Byte 1, Bit 4 - Priority Outbound Service: this bit, when on, indicates to the auto-selection logic that an ESC address that has completed an outbound (to host) data transfer has the highest priority.

Byte 1, Bits 5 through 7 - Residual Byte Count: these bits only apply to PIO operations. They contain the residual byte count (number of bytes that were **not** transferred) for inbound or outbound data transfer sequences.

Note: For AIO operations, the residual byte count is obtained by executing an Input X'C' instruction.

Output X'2' (Data/Status Control Register)

The register addressed by this instruction is used to control the operation of the channel adapter. This instruction also resets the Program Requested Interrupt and Suppress Out Monitor bits. It should be issued only if a level 3 channel adapter initial selection interrupt has been set. The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	Set/reset outbound data transfer sequence (Note)
	1	Set/reset inbound data transfer sequence (Note)
	2	Set/reset status transfer sequence (Note)
	3	Set/reset ESC operation (Note)
	4	Set/reset PIO mode (Note)
	5	Reset initial selection interrupt
	6	Reset data/status interrupt
	7	(not used)
1	0	Set monitor for circle B
	1	(not used)
	2	Set monitor for 2848 ETX
	3	Set suppressible status
	4	Priority outbound service
	5	Request byte count bit 5
	6	Request byte count bit 6
	7	Request byte count bit 7

Note: Set = 1; reset = 0. **Byte 0, Bit 0 - Set/Reset Outbound Data Transfer Sequence:** this bit, when on, sets the Outbound Data Transfer bit and causes the channel adapter to initiate an outbound (controller to host) data transfer sequence.

If the bit is off, it causes the Channel Adapter Outbound Data Transfer Sequence bit to be reset to zero (if it is on).

NSC: if the transfer is over the native subchannel (byte 0, bit 3 = 0), the transfer is initiated using the assigned NSC address.

ESC: if the transfer is over an emulator subchannel (byte 0, bit 3 = 1), the transfer is initiated using the address contained in register X'3'.

Byte 0, Bit 1 - Set/Reset Inbound Data Transfer Sequence: this bit, when on, sets the Inbound Data Transfer bit and causes the channel adapter to initiate an inbound (host to controller) data transfer sequence.

If the bit is off, it causes the Channel Adapter Inbound Data Transfer Sequence bit to be reset to zero (if it is on).

NSC: if the transfer is over the native subchannel (byte 0, bit 3 = 0), the transfer is initiated using the assigned NSC address.

ESC: if the transfer is over an emulator subchannel (byte 0, bit 3 = 1), the transfer is initiated using the address contained in register X'3'.

Byte 0, Bit 2 - Set/Reset Status Transfer Sequence: this bit, when on, sets the Status Transfer bit and causes the channel adapter to initiate a status transfer sequence.

If the bit is off, it causes the Channel Adapter Status Transfer Sequence bit to be reset to zero (if it is on).

NSC: if the transfer is over the native subchannel (byte 0, bit 3 = 0), the transfer is initiated using the assigned NSC address and the NSC status byte (from register X'6').

The NSC status remains available until it is accepted by the host channel.

Note (NSC only): Byte 0, bits 0 and 2 may be on together; the channel adapter hardware then presents a Channel End status to the host.

ESC: if the transfer is over an emulator subchannel (byte 0, bit 3 = 1), the transfer is initiated using the ESC address and status contained in register X'3'.

When the status is presented to the channel, the 'ESC TIO Status Available' latch is reset. If the status is stacked, the ESC address and the status that was stacked are moved by hardware to the ESC TIO Address and Status byte register (X'B'); at the same time, the ESC TIO Status Available latch is set.

Byte 0, Bit 3 - Set/Reset ESC Operation:

NSC (Byte 0, Bit 3 Off)

The transfer sequence defined by byte 0, bits 0 through 2, is initiated on the native subchannel using the assigned hardware address. In the case of a status transfer, the status is taken from the NSC status register (X'6').

ESC (Byte 0, Bit 3 On)

The transfer sequence defined by byte 0, bits 0 through 2, is initiated on the emulation subchannel using the ESC address (and the status, in the case of a status transfer) taken from the ESC address and status register (X'3'). **Byte 0, Bit 4 - Set/Reset PIO Mode:**

AIO Mode (Byte 0, Bit 4 Off)

The data transfer sequence defined by byte 0, bit 0 or 1, is executed in AIO mode using the cycle steal mechanism. Byte 1, bits 5 through 7 are not used. The request byte count and the special control functions are taken from register X'C'.

PIO Mode (Byte 0, Bit 4 On)

The data transfer sequence defined by byte 0, bit 0 or 1, is executed in PIO mode with program intervention required every 4 bytes. The request byte count is taken from byte 1, bits 5 through 7, while byte 1, bits 0 and 2 enable certain special control functions. The transfer byte count and special control bits contained in register X'C' are not used for PIO operations.

Byte 0, Bit 5 - Reset Initial Selection Interrupt: This bit, when on, causes the channel adapter to reset all the initial selection hardware latches and also the channel adapter level 3 interrupt request resulting from an initial selection sequence. This bit does not reset a 'system reset' condition, nor the resulting channel adapter level 3 interrupt request.

Byte 0, Bit 6 - Reset Data/Status Interrupt: This bit, when on, causes the channel adapter to reset the following bits in the Data/Status Control register (Input X'2'):

- Channel stop/interface disconnect (byte 0, bit 5)

- Channel bus out check (byte 1, bit 0)
- Selective reset (byte 1, bit 1)
- Stacked ending status (byte 1, bit 3)

The channel adapter data/status level 3 interrupt is also reset. In addition, if one of the transfer bits (byte 0, bits 0 through 2) is on, the channel adapter hardware also raises the 'Request In' channel interface tag line in order to initiate the transfer sequence.

Byte 1, Bit 0 - Set Monitor for Circle B: this bit turns on the monitoring for the 'Circle B' character on inbound data transfer only. If the bit is off, the monitoring is reset.

Byte 1, Bit 2 - Set Monitor for 2848 ETX: this bit turns on the monitoring for the 2848 'ETX' character on inbound data transfer only. If the bit is off, the monitoring is reset.

Byte 1, Bit 3 - Set Suppressible Status: this bit is set by the control program after a status has been stacked on the channel to inhibit a status transfer to the host as long as 'Suppress Out' is active. It is also set:

- When the control program presents an ending status to a selective reset.
- When an asynchronous 'Attention' status occurs.

Byte 1, Bit 4 - Priority Outbound Service: this bit applies to outbound (controller to host) operations on the ESC only. When on, the bit indicates an ESC priority outbound data transfer sequence to the auto-selection logic. Priority outbound service and NSC interrupts have the highest auto-select priority.

Byte 1, Bits 5 through 7 - Request Byte Count: These bits apply to PIO operations only; for AIO operations, they are ignored. They are used to indicate the number of bytes to be transferred to or from the host. A maximum of 4 bytes can be transferred at one time.

Input/Output X'3' (ESC Address and Status Byte Register)

The register addressed by this instruction contains the following information:

- When transferring data to or from the channel, byte 0 contains the address to be used by the ESC. Byte 1 is not used.
- When transferring status information to the channel, byte 0 contains the address to be used by the ESC; byte 1 contains the status.

These two instructions should be issued only if a level 3 channel adapter data/status interrupt has been set, prior to informing the channel adapter that an ESC data/status transfer is required.

Notes:

1. The ESC I/O device address provided by the last Output X'3' instruction executed is presented to the channel in all subsequent ESC transfer sequences. This instruction must therefore be executed each time a transfer sequence is required for a **different** I/O address.
2. For a status transfer sequence, the ESC final status must be provided in byte 1.
3. For reasons of compatibility, the program should ensure that the ESC status bits that are provided are consistent with those that are set under similar circumstances by the 2701, 2702, or 2703.
4. The Input X'3' instruction may be used to determine the ESC address and status that were provided to the channel when Output X'3' was last executed. It may therefore be used for checking purposes, or to obtain the information if it is not kept elsewhere.

The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0-7	ESC address byte bits 0-7 (data/status transfer)
1	0	ESC status byte bit 0 (attention)
	1	ESC status byte bit 1 (status modifier)
	2	ESC status byte bit 2 (control unit end)
	3	ESC status byte bit 3 (Busy)
	4	ESC status byte bit 4 (channel end)
	5	ESC status byte bit 5 (device end)
	6	ESC status byte bit 6 (unit check)
	7	ESC status byte bit 7 (unit exception)

Input/Output X'4' and X'5' (Data Buffer Registers)

The registers addressed by these instructions are used to hold data during data transfers in either direction between the channel adapter and the host channel. The way in which these registers are used depends on whether the buffers are being loaded in program initiated operation mode, or in adapter initiated operation mode. The bits of the registers have the following meaning:

Register X'4' (Data Buffer Bytes 1 and 2 or 5 and 6)

Byte	Bit	Meaning
0	0-7	Data buffer byte 1 or 5 bits 0-7
1	0-7	Data buffer byte 2 or 6 bits 0-7

Register X'5' (Data Buffer Bytes 3 and 4 or 7 and 8)

Byte	Bit	Meaning
0	0-7	Data buffer byte 3 or 7 bits 0-7
1	0-7	Data buffer byte 4 or 8 bits 0-7

Note: Bytes 5 through 8 are used for diagnostic purposes only.

Program-Initiated Operation (PIO)

To transfer data in PIO mode, byte 0, bit 4 must be set to 1. Four one-byte buffers are used for data transfer, two bytes (1 and 2) being contained in register X'4', and two bytes (3 and 4) in register X'5'.

During an inbound operation (from host): byte 0, bit 1 of register X'2' is on. The four one-byte buffers are loaded by the channel adapter hardware. When the four bytes have been loaded, a data/status level 3 interrupt occurs. The level 3 control program must then empty the buffers by software using the Input X'4' and X'5' instructions. Program intervention is thus required, in general, for every 4 bytes.

Programming Notes:

1. After power on, the buffers should be loaded via the Output X'4' and X'5' commands to prevent parity errors when reading.

2. Before executing the Input X'4' and X'5' instructions, the program should first examine the residual byte count in register X'2':

   - Buffer byte 1 contains valid data if the residual count is less than the requested transfer count (issue Input X'4' instruction).

   - Buffer byte 2 contains valid data if the residual count is at least 2 less than the requested transfer count (issue Input X'4' instruction).

   - Buffer byte 3 contains valid data if the residual count is at least 3 less than the requested transfer count (issue Input X'4' and X'5' instructions).

- Buffer byte 4 contains valid data only if the residual count is 0 and a 4-byte transfer was requested (issue Input X'4' and X'5' instructions).

During an outbound (to host) operation, the four one-byte buffers are loaded by the program using the Output X'4' and X'5' instructions. The program must then start the hardware data transfer by setting on byte 0, bit 0 in the Data/Status Control Register (X'2'). When the four data bytes have been transferred across the channel, the hardware causes a data/status level 3 interrupt to ask for four more bytes to be loaded into the buffers.

Programming Note:

To ensure data integrity, the request byte count contained in byte 1, bits 5 through 7 of register X'2' must be consistent with the number of data bytes loaded into the data buffers.

Adapter-Initiated Operation (AIO)

In AIO mode, two 8-byte buffers are used alternately for data transfer. They are called the A and B Data Buffers. The operation of loading and unloading the buffers is done entirely by hardware; program intervention is never required during normal operation. The two buffers are switched between the CCU and the channel adapter:

- During a host write operation, one Data Buffer is loaded by the channel adapter hardware while the other is emptied by the CCU cycle steal mechanism.
- During a host read operation, one Data Buffer is loaded by the CCU cycle steal mechanism while the other is emptied by the channel adapter hardware.

There are, however, occasions when it is necessary for the **program** to load or read the contents of the A/B Data Buffers, for example, during diagnostic routines, or for recovery purposes. The program can access them via the Input/Output X'4' and X'5' instructions. The A or B Data Buffer must first be selected via the A or B Data Buffer Diagnostic bit in the NSC Status/Control register (byte 0, bit 6). If the bit is off, the A Data Buffer is selected; if it is on, the B Data Buffer is selected. The Input/Output X'4/5' instructions can then access bytes 1 through 4 directly, and bytes 5 through 8 by first setting on the Diagnostic Storage Mode bit in the NSC Status/Control register (Output X'6', byte 0, bit 3). The following diagram should make this clear:

	Diagnostic Storage Mode Bit	
Instruction	Off	On
IOH/IOHI X'4'	Bytes 1 and 2	Bytes 5 and 6
IOH/IOHI X'5'	Bytes 3 and 4	Bytes 7 and 8

Programming Note

After power on, the buffers should be loaded via the Output X'4' and X'5' commands to prevent parity errors when reading.

Input X'6' (NSC Status/Control Register)

The register addressed by this instruction contains the current NSC status byte. The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	Channel adapter switched to interface B
	1	Channel adapter switched to interface A
	2	(not used)
	3	(not used)
	4	(not used)
	5	(not used)
	6	(not used)
	7	(not used)
1	0	NSC status byte bit 0 (attention)
	1	NSC status byte bit 1 (status modifier)
	2	NSC status byte bit 2 (control unit end)
	3	NSC status byte bit 3 (Busy)
	4	NSC status byte bit 4 (channel end)
	5	NSC status byte bit 5 (device end)
	6	NSC status byte bit 6 (unit check)
	7	NSC status byte bit 7 (unit exception)

Byte 0, Bit 0 - Channel Adapter Switched to Interface B: this bit, when on, indicates that the channel adapter is switched to interface B; byte 0, bit 1 cannot be on at the same time.

Note: The execution of an Output X'6' instruction with byte 0, bit 1 set to 1 will also make this bit active.

Byte 0, Bit 1 - Channel Adapter Switched to Interface A: this bit, when on, indicates that the channel adapter is switched to interface A; byte 0, bit 0 cannot be on at the same time.

Note: The execution of an Output X'6' instruction with byte 0, bit 0 set to 1 will also make this bit active.

Byte 1, Bits 0 through 7 - NSC Status Byte: these are the bits that were set into the NSC status register when an Output X'6' instruction was executed. The bits have the usual meanings of the device status byte.

Output X'6' (NSC Status/Control Register)

The register addressed by this instruction is used to set the current status of the NSC. This status is sent to the channel interface during NSC status transfer sequences. It is also used to set certain conditions in the adapter.

The instruction should be executed before signalling to the channel adapter that an NSC final status transfer sequence is required, if the status byte has not been previously given to the channel adapter. If the status byte has been given previously to the channel adapter, but has been stacked by the channel, it need not be given again.

The instruction should only be executed when an initial selection, data/status, or program controlled interrupt is set. When the NSC final status transfer sequence occurs, the status byte provided by this output is presented to the channel.

This instruction should also be used when presenting an asynchronous status, when presenting the final status byte ending a channel I/O command on the NSC, or if an early channel end is stacked.

The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	Set force A busy
	1	Set force B busy
	2	Force error
	3	Diagnostic storage mode (set = 1; reset = 0)
	4	(not used)
	5	Check the checkers
	6	A/B data buffer diagnostic mode
	7	Reset to neutral state
1	0	Set NSC status byte bit 0 (attention)
	1	Set NSC status byte bit 1 (status modifier)
	2	Set NSC status byte bit 2 (control unit end)
	3	Set NSC status byte bit 3 (busy)
	4	Set NSC status byte bit 4 (channel end)
	5	Set NSC status byte bit 5 (device end)
	6	Set NSC status byte bit 6 (unit check)
	7	Set NSC status byte bit 7 (unit exception)

Byte 0, Bit 0 - Set Force A Busy: this bit, when on, forces interface A of a two-processor switch (TPS) into the busy state, and sets interface B into the long term allegiance state (see under the heading 'Two-Processor Switch Feature' below for a definition of these terms).

Byte 0, Bit 1 - Set Force B Busy: this bit, when on, forces interface B of a two-processor switch (TPS) into the busy state, and sets interface A into the long term allegiance state.

Programming Note:

If for any reason both bits 0 and 1 are active at the same time, interface B is forced to the busy state.

Byte 0, Bit 2 - Force Error: this bit, when on, is used to force errors on the channel interface and driver receiver cards for checking purposes. It should only be set when the channel adapter is in the disabled state.

Byte 0, Bit 3 - Diagnostic Storage Mode: this bit, when on, sets the diagnostic storage mode latch. This causes the Input/Output X'4' and X'5' instructions to access bytes 5 through 8 of one of the two 8-byte data buffers instead of bytes 1 through 4. When the bit is off, it resets the diagnostic storage mode latch, and the instructions mentioned above go back to accessing bytes 1 through 4.

Byte 0, Bit 5 - Check the Checkers: this bit, when on, is used in conjunction with byte 0, bit 2 to allow the checkout of various checking circuits in the channel adapter.

Note: Bits 2 and 5 need not necessarily be on at the same time.

Byte 0, Bit 6 - A/B Data Buffer Diagnostic Mode: this bit is used to select one of the two 8-byte data buffers, A and B. When the bit is off, the Input/Output X'4' and X'5' instructions access data buffer A . When the bit is on, the Input/Output X'4' and X'5' instructions access data buffer B .

Byte 0, Bit 7 - Reset to Neutral State: this bit, if on, returns the channel adapter to the neutral state from the long-term allegiance state (see under the heading 'Two-Processor Switch Feature' below for a definition of these terms).

Byte 1, Bits 0 through 7 - Set NSC Status Byte: when the Output X'6' instruction is executed, these bits are transferred to the NSC status byte. They have the usual meanings of the device status byte.

Input X'7' (Channel Adapter Condition Register)

The register addressed by this instruction contains information mainly concerning the enabled/disabled status of the channel adapter interfaces.

Note: The channel adapter condition register contains information concerning **all** the channel adapters.

The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	Channel adapter 5 enabled
	1	(not used)
	2	Channel adapter 6 enabled
	3	(not used)
	4	(not used)
	5	NSC address active
	6	PIO mode
	7	(not used)
1	0	Channel adapter 1 interface A enabled
	1	Channel adapter 1 interface B enabled
	2	Channel adapter 2 interface A enabled
	3	Channel adapter 2 interface B enabled
	4	Channel adapter 3 interface A enabled
	5	Channel adapter 3 interface B enabled
	6	Channel adapter 4 interface A enabled
	7	Channel adapter 4 interface B enabled

Byte 0, Bit 0 - Channel Adapter 5 Enabled: this bit is set to 1 by the channel hardware when channel adapter 5 is enabled. It is reset when channel adapter 5 is disabled.

Byte 0, Bit 2 - Channel Adapter 6 Enabled: this bit is set to 1 by the channel hardware when channel adapter 6 is enabled. It is reset when channel adapter 6 is disabled.

Byte 0, Bit 5 - NSC Address Active: this bit is set by hardware under the following conditions:

- When the NSC is initially selected by accepting a command.
- When the program executes an Output X'2' instruction with byte 0, bit 2 ('Status Transfer Sequence') on.

The bit remains active until the 'Device End' ending status is accepted for this command.

The bit is reset when an NSC 'Device End' status is accepted by the host channel on an NSC final status transfer.

Byte 0, Bit 6 - PIO Mode: this bit, when on, indicates that the last Output X'2' instruction set the PIO data transfer mode; all subsequent data transfer sequences use the PIO mode until it is reset. When the bit is off, it indicates that the last Output X'2' instruction set the AIO data transfer mode.

Programming Note:

After a reset, the bit is set to 0, forcing AIO mode.

Byte 1, Bit 0 - Channel Adapter 1, Interface A Enabled: this bit is set to 1 by the channel hardware when interface A of channel adapter 1 is enabled and reset when it is disabled. If a TPS is not installed on this channel, it is set when channel adapter 1 is enabled and reset when it is disabled.

Byte 1, Bit 1 - Channel Adapter 1, Interface B Enabled: this bit is set to 1 by the channel hardware when interface B of channel adapter 1 (equipped with a TPS) is enabled.

Byte 1, Bit 2 - Channel Adapter 2, Interface A Enabled: this bit is set to 1 by the channel hardware when interface A of channel adapter 2 is enabled and reset when it is disabled. If a TPS is not installed on this channel, it is set when channel adapter 2 is enabled and reset when it is disabled.

Byte 1, Bit 3 - Channel Adapter 2, Interface B Enabled: this bit is set to 1 by the channel hardware when interface B of channel adapter 2 (equipped with a TPS) is enabled.

Byte 1, Bit 4 - Channel Adapter 3, Interface A Enabled: this bit is set to 1 by the channel hardware when interface A of channel adapter 3 is enabled and reset when it is disabled. If a TPS is not installed on this channel, it is set when channel adapter 3 is enabled and reset when it is disabled.

Byte 1, Bit 5 - Channel Adapter 3, Interface B Enabled: this bit is set to 1 by the channel hardware when interface B of channel adapter 3 (equipped with a TPS) is enabled.

Byte 1, Bit 6 - Channel Adapter 4, Interface A Enabled: this bit is set to 1 by the channel hardware when interface A of channel adapter 4 is enabled and reset when it is disabled. If a TPS is not installed on this channel, it is set when channel adapter 4 is enabled and reset when it is disabled.

Byte 1, Bit 7 - Channel Adapter 4, Interface B Enabled: this bit is set to 1 by the channel hardware when interface B of channel adapter 4 (equipped with a TPS) is enabled.

Output X'7' (Channel Adapter Control Register)

This instruction is recognized by all channel adapters. The main purpose of this register is to select one of the six channel adapters:

- For the duration of the instruction (temporary selection).
- Until the channel adapter selection is changed, either by the auto-selection mechanism, or by another Output X'7' instruction.

This instruction is also used to control channel adapter operations by setting and resetting control latches. The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	Enable auto-selection) cannot be on at
	1	Disable auto-selection) the same time.
	2	Select channel adapter addressed by bits 4-6
	3	Execute output on CA addressed by bits 4-6
	4	Channel adapter selection bit 0)
	5	Channel adapter selection bit 1) CA address 1-6
	6	Channel adapter selection bit 2)
	7	Channel adapter reset
1	0	Set suppress out monitor
	1	Set program requested interrupt
	2	Reset channel adapter interrupt level 1 checks
	3	Reset system reset/NSC address active
	4	Set allow channel interface enable (A and B)
	5	Set ESC operational
	6	Set ESC command free
	7	Set allow channel interface disable (A and B)

Byte 0, Bit 0 - Enable Auto-Selection: this bit, when on, enables the auto-selection mechanism for all channel adapters.

Byte 0, Bit 1 - Disable Auto-Selection: this bit, when on, disables the auto-selection mechanism for all channel adapters.

Programming Note to Bits 0 and 1: Byte 0, bits 0 and 1 must not both be on together.

Byte 0, Bit 2 - Select Channel Adapter Addressed by Bits 4 through 6: this bit, when on, causes the channel adapter whose address is contained in byte 0, bits 4 through 6 to be selected for all subsequent channel operations.

Byte 0, Bit 3 - Execute Output on Channel Adapter Addressed by Bits 4 through 6: this bit, when on, allows a **particular** channel adapter to be **temporarily** selected in order to set or reset one or more of the conditions specified by byte 0, bit 7, and byte 1, bits 0 through 7. Byte 0, bits 4 through 6 indicate in which channel adapter the condition is to be set or reset.

Note: The channel adapter which was selected by the auto-selection mechanism, or by the execution of this instruction with byte 0, bit 2 on, is not changed.

Programming Notes to Bits 2 and 3:

1. If one of the installed channel adapters has already been selected, either by the auto-selection mechanism or by a previous Output X'7' instruction, bits 2 and 3 may both be off.

2. If none of the installed channel adapters has been previously selected, either bit 2 or bit 3 must be on, and a valid channel adapter address must be contained in bits 4 through 6.

Byte 0, Bits 4 through 6 - Channel Adapter Selection Bits: these bits form the address of the channel adapter selected by byte 0, bits 2 or 3, either temporarily, or for subsequent instructions. The three bits are decoded as follows:

Bit 4 5 6	Channel Adapter
0 0 0	1
0 0 1	2
0 1 0	3
0 1 1	4
1 0 0	5
1 0 1	6

Note: If the Output X'7' instruction addresses a channel adapter that is not installed, the CCU hardware times out and sets the 'PIO Halt Remember' bit in register X'D'. The lagging address register (LAR) points to the failing instruction. All channel adapters raise a level 1 interrupt.

Byte 0, Bit 7 - Channel Adapter Reset: this bit simulates a 'power-on reset' in the channel adapters. It should be executed only when the channel interface is not enabled, or as a last resort when the channel adapter is hung up on the interface.

Programming Note:

The program should first execute an Output X'7' instruction with byte 1, bit 7 'Set Allow Interface Disable' = 1 in an attempt to disable both interfaces A and B (if installed) before executing this instruction.

Byte 1, Bit 0 - Set Suppress Out Monitor: this bit, when on, causes the channel adapter to monitor for the inactive state of the 'Suppress Out' tag line. If this inactive state is detected, the channel adapter sets a data/status level 3 interrupt request, and also byte 0, bit 6 of register X'2' ('Suppress Out Monitor Interrupt'), which may be read using an Input X'2' instruction.

Programming Note:

This bit may be used by the program after a stacked status condition to cause the channel adapter to signal when the suppress status indication has been removed.

Byte 1, Bit 1 - Set Program Requested Interrupt: this bit, when on, causes a channel adapter data/status interrupt request and byte 0, bit 7 of register X'2' ('Program Requested Interrupt') to be set immediately, unless one of the following conditions occurs:

1. A data/status transfer sequence has been initiated.

2. The host is initiating an initial selection sequence on the channel.

3. Chaining is indicated.

4. In two-processor switch operation, if a tagged 'Device End' status is being presented to a previous 'Busy' status.

In this case, the level 3 data/status interrupt is not set until these sequences are complete.

Byte 1, Bit 2 - Reset Channel Adapter Interrupt Level 1 Checks: this bit, when on, causes the channel adapter to reset the channel adapter level 1 check latches, which in turn resets the channel adapter level 1 interrupt request.

Byte 1, Bit 3 - Reset System Reset/NSC Address Active: this bit, when on, causes the channel adapter to reset 'System Reset' (Input X'0', byte 0, bit 7) and 'NSC Address Active' (Input X'7', byte 0, bit 5). If the channel adapter initial level 3 interrupt request is found to be due to the detection of a system reset sequence, this bit must be used to reset 'system reset' and the associated level 3 interrupt.

Byte 1, Bit 4 - Set Allow Channel Interface Enable (A and B): this bit, when on, causes the channel adapter to set the 'Allow Channel Interface Enable' latch for both interfaces A and B. When the 'Enable Interface A' and/or 'Enable Interface B' signal(s) is sent to the channel adapter from the MOSS, the appropriate interfaces are enabled if 'Select Out' from the channel is not active. This bit must not be on simultaneously with byte 1, bit 7.

Programming Note:

After a power on reset, the channel interface cannot be enabled until the Output X'7' instruction is executed with this bit on.

Byte 1, Bit 5 - Set ESC Operational: this bit, when on, causes the channel adapter to make the emulator subchannel (ESC) addresses operational.

Byte 1, Bit 6 - Set ESC Command Free: this bit, when on, causes the channel adapter to reset the 'ESC Command Active' latch (this latch is set when the channel adapter hardware detects an initial selection sequence to an ESC address). When the latch is off, it indicates that the ESC part of the channel adapter is free of commands. Since the channel adapter cannot disable the interface until it is free of commands, the program must set this bit whenever it detects that it is free of ESC commands; if the hardware has previously set this latch due to an ESC initial selection, the channel adapter will not disable the interface.

Byte 1, Bit 7 - Set Allow Channel Interface Disable (A and B): this bit, when on, causes the channel adapter to set the 'Allow Channel Interface Disable' latch for both interfaces A and B. This latch overrides channel adapter enable/disable when the interfaces are free of commands, no chaining is specified, not in an initial selection sequence, there is no pending device end status, the 'Operational In' driver is not active, and 'Select Out' is not active. This bit must not be on simultaneously with byte 1, bit 4.

Programming Note:

If a status with at least the Unit Check bit set (Output X'6', byte 1, bit 6) is exchanged **before** the enable/disable latch has been overriden, a channel adapter reset (Output X'7', byte 0, bit 7) must be sent to the channel adapter. This reset must be sent within 500 miliseconds after the channel adapter has been found to be in the disabled state (as indicated by the corresponding bit(s) in the Input X'7' instruction).

Input/Output X'B' (ESC Test I/O Address and Status Register)

The register addressed by these instructions contains the Test I/O address in byte 0 and the Test I/O status in byte 1. The register is loaded either by the control program via the Output X'B' instruction, or by the stacking of a final status on the ESC.

The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0-7	ESC TIO address byte bits 0-7
1	0	ESC TIO status byte bit 0 (attention)
	1	ESC TIO status byte bit 1 (status modifier)
	2	ESC TIO status byte bit 2 (control unit end)
	3	ESC TIO status byte bit 3 (busy)
	4	ESC TIO status byte bit 4 (channel end)
	5	ESC TIO status byte bit 5 (device end)
	6	ESC TIO status byte bit 6 (unit check)
	7	ESC TIO status byte bit 7 (unit exception)

Input X'C' (Cycle Steal Mode Control Register)

The register addressed by these instructions is used in AIO mode; it contains various cycle steal controls in byte 0, and a residual byte count in byte 1. The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	SYN monitor latch
	1	DLE remember control latch
	2	USASCII monitor control latch
	3	EBCDIC monitor control latch
	4	(not used)
	5	(not used)
	6	(not used)
	7	(not used)
1	0-7	Residual byte count bits 0-7

Byte 0, Bit 0 - SYN Monitor Latch: this bit is used only on BSC inbound (from host) operations. It indicates that a stream of four SYN characters was detected in the incoming data stream during the current operation. See under the heading 'BSC Control Character Recognition' for the use of this bit.

Byte 0, Bit 1 - DLE Remember Control Latch: this bit is used only on BSC inbound (from host) operations and indicates the state of the 'DLE Remember Control' latch. This latch is set by the channel adapter hardware each time that a DLE character is detected in the incoming data stream; it is reset by the following character (if it is not another DLE). If the DLE is the **last** character in the transfer sequence, the latch will stay on, and the input instruction will find byte 0, bit 1 on. See under the heading 'BSC Control Character Recognition' for the use of this bit.

Programming Note:

If the Input X'C' instruction finds this bit on, it must restore the bit to activity when a new inbound transfer sequence is initiated for the same subchannel address (this is to continue the test for the start of transparent mode).

Byte 0, Bit 2 - USASCII Monitor Control Latch: this bit is used only on BSC inbound (from host) operations and indicates that monitoring was carried out by the channel adapter hardware on the last transfer sequence for certain USASCII control characters. See under the heading 'BSC Control Character Recognition' for the use of this bit.

Byte 0, Bit 3 - EBCDIC Monitor Control Latch: this bit is used only on BSC inbound (from host) operations and indicates that monitoring was carried out by the channel adapter hardware on the last transfer sequence for certain EBCDIC control characters. See under the heading 'BSC Control Character Recognition' for the use of this bit.

Byte 1, Bits 0 through 7 - Residual Byte Count: this byte contains the residual byte count for the transfer sequence that has just ended. On outbound (to host) operations, the residual count indicates the number of bytes that were **not** transferred to the host. On inbound (from host) operations, the residual count indicates the **difference** between the number of bytes requested, and the number of bytes actually transferred from the host.

Output X'C' (Cycle Steal Mode Control Register)

The register addressed by these instructions is used in AIO mode; it contains various cycle steal controls in byte 0, and a byte count in byte 1. The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	SYN monitor control latch (Note)
	1	DLE remember control latch (Note)
	2	USASCII monitor control latch (Note)
	3	EBCDIC monitor control latch (Note)
	4	(not used)
	5	(not used)
	6	(not used)
	7	(not used)
1	0-7	Request byte count bits 0-7

Note: 1 = set; 0 = reset

Byte 0, Bit 0 - SYN Monitor Control Latch: this bit is used only on BSC inbound (from host) operations. When on, it sets the 'SYN Monitor Control' latch. It causes the hardware to monitor the data coming from the host for SYN characters. If four consecutive SYN characters are detected in the incoming data stream, the data transfer is terminated, and a channel adapter data/status level 3 interrupt is requested. The 'SYN Monitor Control Latch' is reset when any non-SYN character is detected. When the bit is off, the 'SYN Monitor Control' latch is reset, and SYN monitoring is stopped.

Byte 0, Bit 1 - DLE Remember Control Latch: this bit is used only on BSC inbound (from host) operations and is used to restore the state of the 'DLE Remember Control' latch at the start of a new transfer sequence (this is to continue the test for the start of transparent mode). See under the heading 'BSC Control Character Recognition' for the use of this bit.

Byte 0, Bit 2 - USASCII Monitor Control Latch: this bit is used only on BSC inbound (from host) operations. When on, it indicates that monitoring for certain USASCII control characters is to be carried out by the channel adapter hardware. See under the heading 'BSC Control Character Recognition' for the use of this bit.

Byte 0, Bit 3 - EBCDIC Monitor Control Latch: this bit is used only on BSC inbound (from host) operations. When on, it indicates that monitoring for certain EBCDIC control characters is to be carried out by the channel adapter hardware. See under the heading 'BSC Control Character Recognition' for the use of this bit.

Byte 1, Bits 0 through 7 - Request Byte Count: this byte contains the count of the number of bytes that are to be transferred to/from the host.

Programming Note:

The Request Byte Count loaded into byte 1 of register X'C' must never be greater than the storage size of the buffer reserved for the AIO data transfer. The total number of bytes transferred by the channel adapter will never be greater than the initial value loaded by the Output X'C' instruction.

The contents of the cycle steal address pointer register are unpredictable at the end of any data transfer, and should not be used to determine the exact number of bytes that were transferred. The exact number of bytes transferred may be determined via an Input X'C' instruction. The cycle steal pointer must always be set via an Output X'30' through X'35' instruction before starting an AIO transfer using an Output X'2' instruction.

Input X'D' (Channel Adapter Level 1 Interrupt Check Register)

The register addressed by this instruction is set by hardware with the various checks that can cause a level 1 interrupt. Other bits do not themselves cause a level 1 interrupt, but may help to localize the cause of the interrupt. The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	PIO bus parity error
	1	Internal bus parity error
	2	CCU interconnection card check
	3	(not used)
	4	Channel interface card check
	5	Address compare error
	6	Initiate service latch ungated
	7	(not used)
1	0	Output exception check
	1	PIO halt remember
	2	Cycle steal halt remember
	3	Bus in check interface A
	4	Ground Fault Error
	5	Bus in check interface B
	6	Driver/receiver card check interface A
	7	Driver/receiver card check interface B

Byte 0, Bit 0 - PIO Bus Parity Error: this bit, when on, indicates that a bad parity has been detected on the PIO bus between the CCU and the channel adapter. If the error was detected on data transferred from the CCU to the channel adapter, this bit is set. This bit does not cause a level 1 interrupt request.

Byte 0, Bit 1 - Internal Bus Parity Error: this bit, when on, indicates that a bad parity has been detected between the channel interface card and the CCU interconnection card.

Byte 0, Bit 2 - CCU Interconnection Card Check: this bit, when on, indicates that a hardware failure has been detected on the CCU interconnection card. Four different hardware failures may cause this check:

1. PIO bus parity error on inbound (from host) operations.
2. Internal bus parity error on outbound (to host) operations.
3. I/O command decoder failure.

   Note: If an invalid instruction (X'8', X'9', or X'A') is executed, the I/O command decoder will fail and cause this check; in this case, it is a program check.

4. Byte counter failure.

Byte 0, Bit 4 - Channel Interface Card Check: this bit, when on, indicates that a hardware failure has been detected on the channel interface card.

Byte 0, Bit 5 - Address Compare Error: this bit, when on, indicates that the program has addressed an emulator subchannel that is outside of the plugged address range.

Byte 0, Bit 6 - Initiate Service Latch Ungated: this bit, when on, is set on at the start of a data or status transfer initiated by the control program. It indicates that the channel adapter is starting a control unit initiated sequence by raising 'Request In' to the host, or that the channel adapter is actually transferring data or a status to the host. This bit does not cause a level 1 interrupt request.

Byte 1, Bit 0 - Output Exception Check: this bit, when on, indicates that the channel adapter hardware has detected an invalid Output instruction. Output instructions, with the single exception of Output X'7', are not allowed during data/status transfer. The Output X'B' instruction is only allowed during an initial select level 3 interrupt. This bit does not cause a level 1 interrupt request.

Byte 1, Bit 1 - PIO Halt Remember: this bit, when on, indicates that the CCU has detected an error during an Input/Output operation, and has activated the 'Halt' signal. This bit does not cause a level 1 interrupt request.

Byte 1, Bit 2 - Cycle Steal Halt Remember: this bit, when on, indicates that the CCU has detected an error during cycle stealing, and has activated the 'Halt' signal. This bit does not cause a level 1 interrupt request.

Byte 1, Bit 3 - Bus In Check Interface A: this bit, when on, indicates that a hardware failure has occurred on the channel adapter internal bus path during a data or address transfer to the host. This condition was detected on interface A.

Byte 1, Bit 4 - Ground Fault Error: this bit, when on, indicates that a channel bus in or tag in (except Select In) signal is shorted to ground.

Byte 1, Bit 5 - Bus In Check Interface B: this bit, when on, indicates that a hardware failure has occurred on the channel adapter internal bus path during a data or address transfer to the host. This condition was detected on interface B.

Byte 1, Bit 6 - Driver/Receiver Card Check Interface A: this bit, when on, indicates:

1. A hardware failure has been detected on the interface A driver/receiver card caused by a parity error on the channel interface card during an inbound (from host) data transfer. This parity error is detected when the bus out check is inactive, but the channel interface card detects bad parity on data transferred from the bus out register.

2. A hardware failure was detected in the Bus In or Tag In (except Select In) interface drivers.

Byte 1, Bit 7 - Driver/Receiver Card Check Interface B: this bit, when on, indicates:

1. A hardware failure has been detected on the interface B driver/receiver card caused by a parity error on the channel interface card during an inbound (from host) data transfer. This parity error is detected when the bus out check is inactive, but the channel interface card detects bad parity on data transferred from the bus out register.

2. A hardware failure was detected in the Bus In or Tag In (except Select In) interface drivers.

Programming Note to Byte 0, Bit 0 and Byte 1, Bits 1 and 2: These bits do **not** cause a level 1 interrupt request to the CCU. The channel adapter

hardware reports the error, and the control program takes appropriate action, thus avoiding double reporting of errors.

Input X'E' (Channel Adapter Level 1 Interrupt Requests)

The register addressed by this instruction indicates which channel adapter(s) has a level 1 interrupt pending. It may be read via the Input X'E' instruction when servicing a level 1 interrupt.

Note: The channel adapter level 1 interrupt requests register contains information concerning **all** the channel adapters.

The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	Channel adapter 5 level 1 interrupt request
	1	(not used)
	2	Channel adapter 6 level 1 interrupt request
	3	Channel adapter (any) level 1 interrupt request
	4	Channel adapter address bit 0)
	5	Channel adapter address bit 1) CA address 1-6
	6	Channel adapter address bit 2)
	7	(not used)
1	0	Channel adapter 1 level 1 interrupt request
	1	(not used)
	2	Channel adapter 2 level 1 interrupt request
	3	(not used)
	4	Channel adapter 3 level 1 interrupt request
	5	(not used)
	6	Channel adapter 4 level 1 interrupt request
	7	(not used)

Byte 0, Bit 0 - Channel Adapter 5 Level 1 Interrupt Request: this bit, when on, indicates that channel adapter 5 has a level 1 interrupt request.

Byte 0, Bit 2 - Channel Adapter 6 Level 1 Interrupt Request: this bit, when on, indicates that channel adapter 6 has a level 1 interrupt request.

Byte 0, Bit 3 - Channel Adapter (Any) Level 1 Interrupt Request: this bit, when on, indicates that one or more of the channel adapters has a level 1 interrupt request.

Byte 0, Bits 4 through 6 - Channel Adapter Address: these bits identify the currently selected channel adapter, as follows:

Bit 4 5 6	Channel Adapter
0 0 0	1
0 0 1	2
0 1 0	3
0 1 1	4
1 0 0	5
1 0 1	6

Byte 1, Bit 0 - Channel Adapter 1 Level 1 Interrupt Request: this bit, when on, indicates that channel adapter 1 has a level 1 interrupt request.

Byte 1, Bit 2 - Channel Adapter 2 Level 1 Interrupt Request: this bit, when on, indicates that channel adapter 2 has a level 1 interrupt request.

Byte 1, Bit 4 - Channel Adapter 3 Level 1 Interrupt Request: this bit, when on, indicates that channel adapter 3 has a level 1 interrupt request.

Byte 1, Bit 6 - Channel Adapter 4 Level 1 Interrupt Request: this bit, when on, indicates that channel adapter 4 has a level 1 interrupt request.

Input X'F' (Channel Adapter Level 3 Interrupt Requests)

The register addressed by this instruction indicates which channel adapter is currently selected, and the status of its level 3 interrupts. It may be read via the instruction when servicing a level 3 interrupt.

If the 'Auto-Selection Complete' latch is not set, execution of this instruction initiates the auto-select mechanism (the 'Auto-Selection Complete' latch is set when the interrupt request information is presented in response to the Input X'F' instruction). The latch is reset by:

- An Output X'0' instruction.
- An Output X'2' instruction with either byte 0, bit 5 or 6 set to 1.
- An output X'7' instruction with byte 1, bit 3 set to 1 (for a system reset only).
- An Output X'B' instruction.

For a full description of the auto-selection mechanism, refer to the section 'Channel Adapter Selection by the Auto-Selection Mechanism' at the beginning of this chapter.

The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	(not used)
	1	Two-processor switch installed
	2	Selected CA initial selection L3 interrupt request
	3	Selected CA data/status L3 interrupt request
	4	Channel adapter address bit 0)
	5	Channel adapter address bit 1) CA address 1-6
	6	Channel adapter address bit 2)
	7	(not used)
1	0-7	(not used)

Byte 0, Bit 1 - Two-Processor Switch Installed: this bit, when on, indicates that the currently selected channel adapter is equipped for two-processor switch operation.

Byte 0, Bit 2 - Selected CA Initial Selection L3 Interrupt Request: this bit, when on, indicates that the currently selected channel adapter has an initial selection level 3 interrupt request pending.

Byte 0, Bit 3 - Selected CA Data/Status L3 Interrupt Request: this bit, when on, indicates that the currently selected channel adapter has a data/status level 3 interrupt request pending.

Byte 0, Bits 4 through 6 - Channel Adapter Address: these bits identify the currently selected channel adapter, as follows:

Bit 4 5 6	Channel Adapter
0 0 0	1
0 0 1	2
0 1 0	3
0 1 1	4
1 0 0	5
1 0 1	6

SECTION 4. CHANNEL ADAPTER PROGRAMMING CONSIDERATIONS

CHANNEL ADAPTER INTERRUPT REQUEST HANDLING

Level 1 Interrupt Requests

When an error condition is detected in the channel adapter, a level 1 interrupt occurs, and a bit is set in the Channel Adapter Level 1 Interrupt Check Bit register (X'D') to indicate the type of error, as follows:

Byte	Bit	Meaning
0	0	PIO Bus Parity Error
	1	Internal Bus Parity Error
	2	CCU Interconnection Card Check
	3	(not used)
	4	Channel Interface Card Check
	5	Address Compare Error
	6	(not used)
	7	(not used)
1	0	Output Exception Check
	1	PIO Halt Memory Latch
	2	AIO Halt Memory Latch
	3	Bus In Check Interface A
	4	Bus In Check Interface B
	5	(not used)
	6	Driver/Receiver Card Check I/F A
	7	Driver/Receiver Card Check I/F B

These bits are available to the program via the Input X'D'instruction.

Level 3 Interrupt Requests

There are two types of interrupt request at level 3:

- Channel Adapter Initial Selection Level 3 interrupt request
- Data/Status Transfer Level 3 interrupt request

The type of interrupt request, and the channel number, may be identified by issuing an Input X'F' (Channel Adapter Level 3 Interrupt Requests) instruction:

- Byte 0, bit 2 indicates a Channel Adapter Initial Selection Interrupt Request.
- Byte 0, bit 3 indicates a Channel Adapter Data/Status Interrupt Request.

Both types of interrupt requests may be active at the same time; this condition is indicated by both bits being on in the register.

Channel Adapter Initial Selection Level 3 Interrupt Request

When an Initial Selection interrupt request occurs, the cause of the interrupt may be determined by executing an Input X'0' (Initial Selection Control register) instruction:

- Byte 0, bit 0 indicates that the interrupt was caused by a normal initial selection interrupt request.
- Byte 0, bit 1 indicates that the interrupt request was caused by an interface disconnect sequence (Halt I/O).
- Byte 0, bit 2 indicates that the interrupt request was caused by a selective reset.
- Byte 0, bit 3 indicates that the interrupt request was caused by the detection of bad (even) parity on the I/O channel interface bus out when the channel I/O command byte was presented during initial selection.
- Byte 0, bit 7 indicates that the interrupt request was caused by a system reset.

Note: If the interrupt was caused by a system reset, all the bits of register X'0' will be off, except the System Reset bit (byte 0, bit 7). This is because the system reset sequence resets all the latches in the channel, with the exception of the system reset bit. This means that if an initial selection sequence occurs just before a system reset sequence, all the indications will be lost to the program, unless the initial selection interrupt was completely serviced before the system reset occurred.

Once the Channel Adapter Initial Selection Level 3 Interrupt Request is set, the channel adapter hardware replies with a short control unit busy (CE, End, Status Modifier, and Busy) to all attempts at initial selection until the control program tells the channel adapter to reset the condition that caused the initial selection interrupt. During this period, no channel commands can be accepted. The control program should therefore tell the channel adapter to reset the interrupting condition as soon as possible.

Note: During this period, subsequent data/status transfer sequences are also inhibited. This means that it is possible to have both an initial selection request and a data/status request simultaneously only if the data/status request occurs first, and if the subsequent initial selection request is not due to a System Reset condition.

A Channel Adapter Initial Selection Level 3 Interrupt Request may be reset either by executing an Output X'0' instruction (the bit configuration is ignored), or by executing an Output X'2' instruction with byte 0, bit 5 (reset initial selection) set to 1.

Notes:

1. If the interrupt was due to a System Reset, it must be reset by executing an Output X'7' instruction with byte 1, bit 3 (reset system reset/NSC address active) set to 1.
2. When executing Output X'2' or Output X'7' instructions to reset interrupt requests, the control program must be careful to set/reset all bits correctly to achieve the desired result.

Channel Adapter Data/Status Level 3 Interrupt Request

When a Data/Status interrupt request occurs, the cause of the interrupt may be determined by executing an Input X'2' (Data/Status Control register) instruction:

- Byte 0, bit 0 indicates that the interrupt was caused by the ending of an outbound (to host) data transfer sequence.
- Byte 0, bit 1 indicates that the interrupt was caused by the ending of an inbound (from host) data transfer sequence.
- Byte 0, bit 2 indicates that the interrupt was caused by the ending of a status transfer sequence.
- Byte 0, bit 7 indicates that the interrupt was a program requested interrupt.

Note: If a system reset occurs immediately afterwards, all the bits of register X'2' will be off, but the System Reset bit (byte 0, bit 7 of register X'0' will be on. This is because the system reset sequence resets all the latches in the channel, with the exception of the system reset bit. This means that if a data/status sequence occurs just before a system reset sequence, all the indications will be lost to the program, unless an Input X'2' instruction was executed before the system reset occurred. A Channel Adapter Data/Status Level 3 Interrupt Request may be reset by executing an Output X'2' instruction with byte 0, bit 6 (reset data/status interrupt) set to 1. Unless the control program wants to immediately initiate another transfer sequence, byte 0, bit 0 (reset outbound data transfer), bit 1 (reset inbound data transfer) and bit 2 (reset status transfer) should be set to 0. The execution of an Output X'2' instruction also resets the 'program requested interrupt' (byte 0, bit 7 of Input X'2'), and the 'suppress out monitor interrupt' (byte 0, bit 6 of Input X'2').

Note: When executing Output X'2' or Output X'7' instructions to reset interrupt requests, the control program must be careful to set/reset all bits correctly to achieve the desired result.

Simultaneous Initial Selection and Data/Status Interrupts

When servicing channel adapter level 3 interrupts, the control program should first check whether both Initial Selection and Data/Status interrupts are set, or only one of them.

If only a Channel Adapter Initial Selection Level 3 interrupt request is set, this may be serviced without worrying about the possibility of a Channel Adapter Data/Status Level 3 request occurring afterwards, since any pending data/status transfer sequences are inhibited until the initial selection request is reset. However, if Output X'2' is used to reset the interrupt, it must be remembered that this instruction also controls data/status transfers, and the control program must be prepared to reinitiate a pending transfer sequence if it still requires that transfer to be executed.

If both interrupt requests are set (initial selection request received after data/status request), the program must service both requests before executing the Output X'2' instruction. Remember too that a system reset sequence can occur before the **Input** X'2' is executed.

INITIAL SELECTION SEQUENCES

Channel Commands

Channel commands are commands issued by the channel to the controller. All I/O command byte combinations are valid to the channel adapter hardware provided that good parity is found on the channel interface bus out.

The following channel commands are standard:

		Applicable to	
Hex	Meaning	NSC	ESC
00	Test I/O (TIO)	*	*
01	Write	*	*
02	Read	*	*
03	I/O No-op	*	*
04	Sense	*	*
05	Write IPL	*	
09	Write break	*	
E4	Sense ID	*	

Test I/O (TIO) - (X'00')

NSC

When this command is issued to the NSC address, the channel adapter replies with the current status of the NSC:

- If the NSC is free of commands, the channel adapter replies with an X'00' with a hardware generated status byte during the initial status presentation to the Test I/O command.
- If the NSC is active, and the status is not available in the NSC status register, the channel adapter replies with a hardware generated busy (X'10') status during the initial status presentation to the test I/O command.
- If the NSC has a pending status available in the NSC (software) status register, this status is sent to the channel in response to the Test I/O command. An initial selection level 3 interrupt is raised, with byte 0, bit 6 (Status Byte Cleared) set in register X'0'. There is no busy bit in this status.

  Note: The TIO command must be recognized by the control program if there is a status stacked.

ESC

- When this command is issued to an ESC address, the line address presented to the channel adapter on the channel bus out is compared with the line address contained in register X'B'. If the two addresses compare, the status contained in Byte 1 of register X'B' is presented to the channel.
- If the two addresses do not compare (this will usually be the case), the Test I/O status is not immediately available for that address; the channel adapter hardware presents a short control unit busy status X'70' (Status

Modifier, Control Unit End, and Busy), and raises a channel adapter initial selection level 3 interrupt request.

The control program must obtain the ESC address and command via an Input X'1' instruction, and then execute an Output X'B' instruction to load register X'B' with the ESC address of the line to be serviced in Byte 0, and the status of the line in Byte 1.

When the next initial selection sequence occurs, the channel adapter hardware compares the address on bus out with the contents of Byte 0 of register X'B'. If it is for the same address, an address compare occurs, and the status, contained in Byte 1 of register X'B', is sent to the channel. At the same time, register X'0' Byte 0, Bit 4 (ESC Operation), and Byte 0, Bit 6 (Status Byte Cleared) are set on by hardware, and a channel adapter initial selection level 3 interrupt occurs. The control program may obtain this information by executing an Input X'0' instruction.

Notes:

1. If the addresses do not compare during the initial selection sequence, or if the command is not Test I/O, the hardware handles the selection sequence as a standard initial selection.

2. From the time that the Test I/O command is first issued until the control program reacts by executing the Output X'B' instruction, the channel adapter hardware responds to all initial selection sequences from the host with the short control unit busy status X'70' (Status Modifier, Control Unit End, and Busy).

Write - (X'01')

This command is used to transfer data or control information from the host to the controller. When the command is issued, the channel adapter hardware accepts the command and returns an initial selection status of X'00'. The control program must decode the command and initiate the appropriate action.

Read - (X'02')

This command is used to transfer data from the controller to the host. When the command is issued, the channel adapter hardware accepts the command and returns an initial selection status of X'00'. The control program must decode the command and initiate the appropriate action.

I/O No-Op - (X'03')

NSC

This command is a pseudo-command. When the command is issued, the channel adapter hardware returns an immediate initial selection status of channel end and device end (X'0C') if the channel adapter is free.

If the channel adapter has an Initial Selection or a Program Requested Interrupt pending, a Control Unit Busy status (X'70') with bits 1 (Status Modifier), 2 (Control Unit End), and 3 (Busy) is returned instead.

If the original channel end/device end status is not stacked, no initial selection level 3 interrupt occurs.

If a pending status is available (previous NSC status byte stacked), the channel adapter presents this stacked status to the No-Op command, along with the busy bit. An initial selection level 3 interrupt occurs with byte 0, bit 6 (Status Byte Cleared) in register X'0'.

Note: The No-Op command must be recognized by the control program if there is a status stacked.

ESC

This command is a pseudo-command. When the command is issued to a valid ESC address, the channel adapter hardware returns an immediate initial selection status of channel end and device end (X'0C') if the channel adapter is free.

If the channel adapter has an Initial Selection or a Program Requested Interrupt pending, a Control Unit Busy status (X'70') with bits 1 (Status Modifier), 2 (Control Unit End), and 3 (Busy) is returned instead.

Sense - (X'04')

This command is used to transfer a single byte of sense information from the controller to the host. When the command is issued, the channel adapter hardware accepts the command and returns an initial selection status of X'00'. The normal ending status is channel end and device end (X'0C'), unless a Halt I/O command is detected when the channel adapter is not initialized. In this case, the response is channel end, device end, and unit check (X'0E'). The control program must decode the command create the correct sense byte, and send it to the host. The transfer takes place in the same way as a data transfer with a single byte of data.

Write IPL - (X'05')

This command prepares the controller to receive the control program from the host. No data is actually transferred by the command.

1. The initial status is always channel end, thus causing the channel to disconnect immediately. This is necessary because the controller may or may not be ready; if it is not ready, it may take several minutes before the loader program is loaded and ready to receive the control program from the host.

2. A level 3 interrupt to the CCU is raised with channel end; at the same time, the MOSS is informed of the 'Write IPL' command by hardware, and initializes the controller with the loader program, if necessary.

3. When the loader program in the controller is ready, it sends device end to the channel. If the controller is ready immediately, the device end follows the channel end almost at once; however, they never occur together in the same status presentation.

Write Break - (X'09')

This command is exactly the same as the normal write command, with one exception: the command code to be found in byte 1 of register X'1' is X'09' instead of X'01'. This allows the host to inform the control program of the point it has reached in the host CCW chain. The control program should react in the same way as to a normal write command.

Sense ID - (X'E4')

This command is used to determine the unit type. The control program must set up an outbound transfer sequence (to host) to transfer the Unit Identification and Level.

Non-Standard Commands

As previously stated, the channel adapter recognizes all I/O command byte combinations as valid, provided that correct parity is detected on the channel interface bus out. It is the responsibility of the control program to test for validity at the control program level. If an invalid command is received at this level, the control program must end the command by setting up a final status transfer with at least Channel End, Device End, and Unit Check (X'0E').

NSC

When a non-standard command is received by the channel adapter, it replies with an initial status of channel end (X'08') and raises an initial selection level 3 interrupt to the CCU.

If the CE status is stacked, a level 3 interrupt is presented to the CCU to indicate the stacked status. The channel end status is available until an Output X'6' is performed, or until a Halt I/O or a Selective Reset is sent from the host.

ESC

When a non-standard command is received by the channel adapter, it replies with an all-zero initial status and raises an initial selection level 3 interrupt to the CCU. If the ESC control program determines that a particular command byte is invalid, it must terminate the command with an ending status of at least X'0E' (Channel End, Device End, and Unit Check) to that ESC address.

Channel Initial Status

NSC Initial Status

At initial selection, the status returned to the channel may be one of the following:

1. X'00' - All Zero Status

   This status is returned to the channel when:

   a. the hardware has accepted a standard command.

   b. the channel command is Test I/O and the channel adapter is free of commands.

2. X'02' - Unit Check

   This status is returned to the channel when the NSC hardware detects an even parity on the Channel Bus Out for the command byte.

3. X'08' - Channel End

   This status is returned to the channel by the NSC hardware as an immediate Initial Status when the command is a control type command.

4. X'0C' - Channel End and Device End

   This status is returned to the channel by the NSC hardware as an immediate Initial Status to a No-Op command.

5. X'10' - Busy

   This status is returned to the channel by the NSC hardware when the NSC is already active with another command and has not yet presented a final status for that command.

6. X'70' - Status Modifier, Control Unit End, and Busy

   This status is returned to the channel when:

   a. the channel adapter has an Initial Selection Level 3 interrupt request pending. This is because the channel adapter has accepted a previous command and has not yet reset the interrupt request.

   b. the channel adapter has detected a System Reset and caused an Initial Selection Level 3 interrupt, but the control program has not yet reset the interrupt.

   c. the channel adapter has detected a Selective Reset during a service transfer sequence and caused a Data/Status Transfer Level 3 interrupt, but the control program has not yet reset the interrupt.

   d. a program requested interrupt is pending, but the control program has not yet reset the interrupt.

7. Any Pending Status without the Busy Bit

   This initial status is returned by the NSC hardware when a Test I/O command is issued to the NSC and a hardware generated status is pending.

8. Any Pending Status with the Busy Bit

   This initial status is returned by the NSC hardware when any command other than Test I/O is issued to the NSC and a hardware generated status is pending.

ESC Initial Status

At initial selection, the status returned to the channel may be one of the following:

1. X'00' - All Zero Status

   This status is returned to the channel when the channel adapter accepts an initial selection command byte other than I/O No-Op (X'03') or Test I/O (X'00'), and none of the Control Unit Busy (X'70') conditions (see below) are active.

2. X'02' - Unit Check

   This status is returned to the channel when the ESC hardware detects an even parity on the Channel Bus In for the command byte.

3. X'0C' - Channel End and Device End

   This status is returned to the channel by the ESC hardware as an immediate Initial Status to a No-Op command.

4. X'70' - Status Modifier, Control Unit End, and Busy

   This status is returned to the channel for a Test I/O command when:

   a. an Initial Selection level 3 interrupt request is pending.

   b. a Test I/O command has been issued to an ESC address, but ESC TIO status is not set for that address.

   c. a command is still in progress on another ESC address (ending status not yet accepted).

5. ESC Test I/O Pending Status

   This initial status is returned by the ESC hardware when a TIO command is issued to an ESC address and the ESC TIO Address/Status register (X'B') has been loaded with that subchannels address and status.

Stacked Initial Status

Some initial status responses to channel commands may be stacked by the host. When this happens, the channel adapter hardware causes a channel adapter Initial Selection Level 3 interrupt request.

The initial statuses listed below may be presented by the channel adapter hardware and could be stacked by the channel:

All Zero Initial Status (X'00')

This status is never stacked by the channel unless the command is Test I/O to the NSC. See below under the heading 'Any Initial Status on Test I/O (NSC)'.

Channel End/Device End Initial Status to I/O No-Op (X'0C')

The device address and command may be obtained by executing an Input X'1' instruction. When Input X'0' is executed, byte 0, bit 5 (Stacked Initial Status) will be active.

Unit Check Initial Status (X'02')

This status is caused by a bad parity on Bus Out during command byte transfer. The device address may be obtained by executing an Input X'1' instruction. When Input X'0' is executed, byte 0, bit 3 (Channel Bus Out Check), and bit 5 (Stacked Initial Status) will be active.

Any Initial Status on Test I/O (NSC)

This is the only case in which an all zero status may have been stacked. The NSC address and command may be obtained by executing an Input X'1' instruction. When Input X'0' is executed, byte 0, bit 5 (Stacked Initial Status) will be active. When a Test I/O initial status is stacked for the NSC address, the control program should **not** execute an Output X'6' (NSC Status/Control Register) to put the stacked status in the NSC Status Register. This is because the NSC hardware saves the stacked (pending) status from a Test I/O command into the NSC Status register. The NSC status register contains X'00' if the channel adapter is free of commands or contains the pending or stacked status otherwise. The channel adapter hardware does not reset this register until the host channel has accepted it. The control program can present this status by executing an Output X'2' instruction, unless of course a TIO command retrieves it first.

Any Initial Status on Test I/O (ESC)

The ESC address and command may be obtained by executing an Input X'1' instruction. When Input X'0' is executed, byte 0, bit 5 (Stacked Initial Status) will be active. From this point onwards, the control program should treat this stacked Test I/O status as if it were a status that was stacked during an ESC final status transfer.

SECTION 5. TWO-PROCESSOR SWITCH FEATURE

The two-processor switch feature (TPS) is an optional feature that allows the controller to be attached to a pair of symmetrical tightly-coupled multiprocessors. Additionally, this feature allows the controller to be attached to two different channel interfaces on a single host as an I/O device with alternate path capability. Each channel interface has its own NSC address. ESC operation is only allowed when the channel adapter is operating in a partitioned mode; i.e. only one of the two channel interfaces is enabled at any one time.

In addition to the usual features of a normal channel adapter, a channel adapter fitted with the two-processor switch feature adds the following capabilities:

- Each channel interface may be enabled independently.
- Both channel interfaces (A and B) may be **enabled** (on line) simultaneously. However, simultaneous **operation** over the two interfaces is not permitted.
- When both interfaces are enabled at the same time, contention problems are resolved automatically by the hardware.
- The interfaces may be enabled or disabled via the control program. If an interface is disabled, it bypasses 'Select Out'.
- The channel adapter hardware automatically provides 'allegiance' to a single channel interface for the duration of a channel I/O operation, that is, from initial selection and reception of the first command right up to the reception of a 'Device End' ending status for the last command that does not indicate command chaining.
- During the period when the allegiance of the channel adapter hardware is directed to one channel interface, any initial selection attempt by the other channel results in an 'Abbreviated Device Busy' (X'10') status to that channel.

  When the other channel has ended its I/O operation, the channel adapter hardware automatically presents a 'Device End' status to the other channel (the one that received the 'Abbreviated Device Busy' status).
- Either interface may be enabled or disabled from the control panel. If only one interface is enabled in this way at any one time, it is called the **manual partitioning mode.** Operation with both NSC and ESC addresses is permitted in this mode.

Note for 3725 Model 2 only

The 3725 Model 2 cannot be equipped with a two-processor switch.

States of a Channel Adapter plus TPS

When both channel interfaces are enabled, the channel adapter is in one of two states:

Neutral State

In this state, the channel adapter is not switched to either interface, but is available to both.

Switched State

In this state, the channel adapter has allegiance to one of the interfaces; that is, an initial selection sequence from the host has switched the channel adapter to that interface. Channel commands can now only be accepted on this interface from the channel that sent the command.

Note, however, that the channel adapter continues to monitor the activity on the other interface, and responds to initial selection requests on this interface either by temporarily suspending the completion of the initial selection sequence, or by responding with the short 'Device Busy' status (X'10'). During this status sequence, the channel adapter remains connected to the other interface.

Types of Allegiance

Allegiance to an interface may be either short- or long-term, and may change from short- to long-term during the operation. Short- and long-term allegiance is discussed in detail below under the heading 'Duration of Channel Interface Allegiance'.

Instantaneous Allegiance

Instantaneous allegiance is a short-term allegiance. The channel adapter enters this state when it traps 'Select Out' because of a poll to 'Request In' (from the neutral state), or because of an initial selection from the host. Instantaneous allegiance causes the channel adapter hardware to switch temporarily to that interface during the initial selection sequence (until Status In is raised). Instantaneous allegiance ends with the presentation of initial status by the channel adapter.

Implicit Allegiance

Implicit allegiance is a long-term allegiance, and covers the entire execution of an I/O operation. It starts when the channel adapter replies with an 'all zeros' status to an initial selection sequence with a channel command that requires information transfer to complete it:

- Data transfer command
- Ending status to complete a command
- A No-Op command with the command chaining bit on.

It ends when the channel accepts a 'Device End' status for the last command without command chaining indicated.

Any attempt by the channel attached to the opposite interface to select the channel adapter during this time is rejected with a 'Device Busy' status (X'10') during a short busy sequence on the channel interface.

Note: If the I/O operation was ended by an 'interface disconnect', the implicit allegiance lasts until either:

- The control program returns the channel adapter to the neutral state by executing an Output X'6' (NSC Status/Control Register) with byte 0, bit 7 (Reset to Neutral State) set to 1.
- The control program presents 'Device End' status to the channel.

Contingent Allegiance

Contingent allegiance is a type of long-term allegiance. It occurs only for a channel adapter that has ended a command sequence with a 'Unit Check' indication in the status byte. The intention is to ensure that exactly the same path may be used by the host to recover sense data from the controller after the host has received a 'Unit Check' status. While in the contingent allegiance state, any attempt at initial selection by the other interface receives a device busy status (X'10') in reply. Contingent allegiance ends when the control program decodes a channel command (other than Test I/O or No-Op).

Duration of Channel Interface Allegiance

When the channel adapter owes allegiance to one of the interfaces, this allegiance may be either short- or long-term. Both implicit and contingent allegiance are long-term allegiances. A short-term (or instantaneous) allegiance exists only for the duration of initial selection.

If, during the period of short-term allegiance on one interface, an initial selection attempt occurs on the other, the channel adapter logic detects the rise of the 'Select Out' tag, but then temporarily blocks the completion of the selection sequence. There are now two possibilities:

1. The first interface passes to the neutral state (this is the case, for instance, when an asynchronous status has been presented, and either accepted or stacked, or if a No-Op without chaining has been issued by the host. The second interface itself now enters the short-term allegiance state and continues its initial selection routine. The 'Device Busy' status is **not** issued.
2. The first interface passes from short- to long-term allegiance. This occurs if a command was accepted by the channel during the initial selection. The opposite interface now replies to its initial selection with the short 'Device Busy' (X'10') status.

Status Presentation

In the discussion which follows, an **untagged** status is a status that is offered to both channel interfaces at the same time. A **tagged** status is a status that must be offered to a particular interface only.

Untagged Asynchronous Status Presentation

In a Tightly-Coupled Symmetric environment, if the control program wants to present an asynchronous status, it sets this status in the channel adapter logic by means of an Output X'6' (NSC Status/Control Register) instruction, and then starts the transfer by means of an Output X'2' (Data/Status Control Register) instruction, in the usual way.

If both interfaces are enabled, the hardware treats this status as an untagged status, and offers it to both channels by raising the 'Request In' tag to both interfaces. The first channel to poll (by raising the 'Select Out' tag in response to 'Select In') is connected to the channel adapter and receives the status. For the duration of this sequence (considered by the adapter as a short-term allegiance), any channel polls on the opposite interface are ignored by bypassing 'Select Out'. A channel initiated initial selection attempt on the other interface is temporarily suspended by trapping 'Select Out' as described above.

If the status just presented is accepted or stacked by the channel, the channel adapter sets a Data/Status Level 3 Interrupt and returns to the neutral state; any Control Unit initiated polls while the channel adapter is in this Data/Status Level 3 Interrupt state are bypassed on both interfaces. For a stacked status, when the control program re-initiates the status transfer via another Output X'2' instruction, the status is offered to both interfaces.

A Channel Initiated initial selection sequence on either interface causes the channel adapter to switch to the short-term allegiance state for that interface, present the status (accompanied by the 'Busy' bit if the command is anything other than Test I/O), cause an Initial Selection Level 3 Interrupt request, and return to the neutral state.

During this interrupt, any other Channel Initiated initial selection sequences to either interface cause the channel adapter to present Control Unit Busy (X'70') as initial status and return to the neutral state.

Tagged Status Presentation

When the channel adapter has presented a 'Busy' status (X'10') for a Channel Initiated initial selection routine on one interface while the other is in the long-term allegiance state, the channel adapter presents a tagged (over the same interface) asynchronous Device End status when the channel adapter returns to the neutral state (see however the note below) and none of the following conditions is present:

- An initial selection level 3 interrupt request is pending due to a normal initial selection, or to a system reset or selective reset during initial selection.
- A data/status level 3 interrupt request is pending due to a selective reset during data/status transfer.
- A program requested interrupt is pending.

The Device End status can only be presented when these pending interrupts interrupts have been reset.

Note: If a disable has been requested (an Output X'7', with byte 1, bit 7 = Set Allow Channel Interface Disable was executed, or the channel switch on the control panel was switched from the ON to the OFF position while the interface

was in the long term allegiance state) **before** the channel adapter returns to the neutral state, then the tagged device end status is **not** sent.

During this time, if the opposite channel polls the adapter in response to a 'Request In', the resulting 'Select Out' tag is bypassed.

Similarly, if a Channel Initiated initial selection sequence occurs on the opposite channel and the other interface is in instantaneous allegiance, the sequence is temporarily suspended until the Device End status has been presented or stacked. The sequence is then completed at the end of the Device End presentation.

Note: If an asynchronous status is set up by the control program while a tagged Device End status is pending for either interface, this asynchronous status is presented along with the Device End for that interface.

Effect of System Reset

System Reset over Interface with Allegiance

When the channel adapter recognizes the system reset, it completely resets the adapter, ends the channel adapter allegiance, and sets an Initial Selection Level 3 interrupt request. However, if a Device End status caused by a previous Device Busy status over the opposite interface is pending, it is not reset.

During this time, if the opposite channel polls the adapter in response to a 'Request In' from some other control unit, the resulting 'Select Out' tag is bypassed.

Similarly, any Channel Initiated initial selection sequences to either interface cause the channel adapter to switch to that interface, present Control Unit Busy (X'70') as initial status, and return to the neutral state.

System Reset over Interface without Allegiance

When a system reset occurs for the interface that does not have allegiance, it resets the pending tagged Device End status (if any) for that interface. The remaining adapter hardware is not reset, and there is no Initial Selection Level 3 Interrupt request.

System Reset when Adapter is in Neutral State

When a system reset occurs for a channel adapter in the neutral state, it resets only a pending tagged Device End status (if any) for the interface over which the system reset was received. The remaining adapter hardware is not reset, and there is no Initial Selection Level 3 Interrupt request.

Note: If a system reset is presented to both interfaces simultaneously, it completely resets the adapter and sets an Initial Selection Level 3 interrupt request.

Effect of Selective Reset

Selective Reset over Interface with Allegiance

When the channel adapter recognizes the selective reset, it returns to the neutral state and sets an Initial Selection Level 3 interrupt request. If the selective reset is received on an interface having a tagged Device End pending due to a previous presentation of Busy, this tagged Device End is reset.

During the Initial Selection Interrupt, if the opposite channel polls the adapter in response to a 'Request In' from some other control unit, the resulting 'Select Out' tag is bypassed.

Similarly, if a Channel Initiated initial selection sequence occurs to either interface, the channel adapter switches to that interface, enters the short-term allegiance state, presents Control Unit Busy (X'70') as initial status, and returns to the neutral state.

Selective Reset over Interface without Allegiance

A selective reset cannot occur for the interface that does not have allegiance; therefore, the channel adapter hardware is not reset, and there is no Initial Selection Level 3 Interrupt request.

SECTION 6. CHANNEL ADAPTER - SPECIAL TOPICS

BSC Control Character Recognition

The controller contains hardware circuits which search for the following characters or sequence of characters in the incoming data stream from the host:

1. End of Transmission Block (ETB)
2. End of Text (ETX)
3. Data Link Escape (DLE) followed by Start of Text (STX).

These hardware circuits may be activated by setting bit 2 (ASCII) or bit 3 (EBCDIC) in byte 0 of the Cycle Steal Mode Control register (X'C').

A further hardware circuit searches for SYN characters. It is activated by setting byte 0, bit 0 of the Cycle Steal Mode Control register (X'C').

The use of these circuits depends on whether the transmission is taking place in normal or in transparent mode.

Normal Text Mode Operation

The control program must select the monitoring required by setting the appropriate bit in the Cycle Steal Mode Control register via an Output X'C' instruction:

- Set byte 0, bit 2 on if ASCII monitoring is required
- Set byte 0, bit 3 on if EBCDIC monitoring is required
- Set both bits off if monitoring is not required

When monitoring for ASCII or EBCDIC control characters, if an ETB or ETX character is detected in the stream coming from the host channel, the transfer sequence is terminated **after** the ETB/ETX character has been transferred to the channel adapter. Channel stop is set in register Input X'2' by hardware.

Transparent Text Mode Operation

Suppose that the channel adapter is working in normal text mode, and has been set to monitor for ETB/ETX in either ASCII or EBCDIC. If ETB/ETX is detected, the transfer sequence is ended as described above.

At the same time, however, the channel adapter searches for the two-character sequence DLE/STX, which indicates the start of transparent text. When the sequence DLE/STX is detected, the ASCII/EBCDIC monitor control latch is reset, monitoring is stopped, and the transparent text mode is entered. The control program is informed of this event by the absence of the monitor bits in the Cycle Steal Mode Control register.

Note: It may happen that DLE is the last character of one host write operation and STX is the first character of the next. To ensure correct operation when this occurs, a special bit, the DLE Remember Latch (bit 1), is set in the Cycle Steal Mode Control register. The latch is set when a DLE character is detected, and reset when a non-DLE character is transferred. If the DLE character is the last character in a host write operation, when the program issues an Input X'C' instruction in response to the level 3 interrupt request (Channel End), it will find that the DLE Remember latch is on. The next time that a host write operation occurs for the same address, the control program must restore the DLE Remember bit via an Output X'C' instruction. If the first character of the host write operation is STX, the transparent text mode is entered.

Monitoring for SYN Characters

The controller also contains hardware that searches the incoming data stream from the host for SYN characters. This hardware is activated via one of the ASCII/EBCDIC Control Latches as described above; in addition, a special bit (bit 0) in the Cycle Steal Mode Control register called the SYN Monitor Control Latch must be set. If register X'C', byte 0, bit 2 is on, the circuits search for ASCII SYN characters; if byte 0, bit 3 is on, the hardware searches for EBCDIC SYN.

Each time that the hardware detects a SYN character, it sets Byte 0, bit 0 (SYN Monitor Latch) of Input register X'C'. The next non-SYN character resets it.

If four consecutive SYN characters are detected, the channel adapter disconnects from the channel, and raises a Data/Status level 3 interrupt. Byte 0, bit 0 of register X'C' is left in the on state, and can be read via an Input X'C' instruction.

270X Emulation Considerations

When emulating a 270X, attention must be paid to the following points:

2702/2703 Two-Channel Switch Support

The 3725/3720 does not support the 2702/2703 automatic two-channel switch hardware feature. If the channel issues a 'Reserve' or a 'Release' command, the controller must reject it by returning an ending status of 'Channel End', 'Device End', and 'Unit Check', and by setting the 'Command Reject' bit in the sense byte.

Busy Response to Start I/O and Test I/O

Under normal operating conditions, both the 2702 and 2703, being multiple subchannel adapters, can present a control unit busy condition to a Start I/O or a Test I/O instruction. The reason for this busy condition is that the control unit must store a received command before another operation can be initiated. This busy condition lasts for a maximum of 1 millisecond for the 2702, and 90 microseconds for the 2703.

The 3725/3720 being also a multiple subchannel adapter (in 270X emulation mode) can also present a control unit busy condition since it cannot accept a new command until the previous command has been serviced by the control program. In general, if a Start I/O or a Test I/O terminates with a condition

code of 1 and a Control Unit Busy status, the instruction must be reissued until the busy condition ends.

CHAPTER 5. COMMUNICATIONS SCANNER

This chapter gives the reader a basic understanding of how the communications scanner operates, and the means to program it.

The communications scanner is a processor dedicated to controlling a small number of telecommunication lines. It is controlled by means of the IOH and IOHI instructions, which are used only to initiate operations. After the instruction has been issued to the scanner, cycle stealing is used to transfer control information and data between the CCU and the communications scanner at high speed. The operation continues without further intervention by the program until all the control information and data have been transferred. An interrupt then informs the CCU that the operation has completed. Communications scanner interrupts are at two different levels:

- At level 2 if the operation was completed.
- At level 1 if the communications scanner fails.

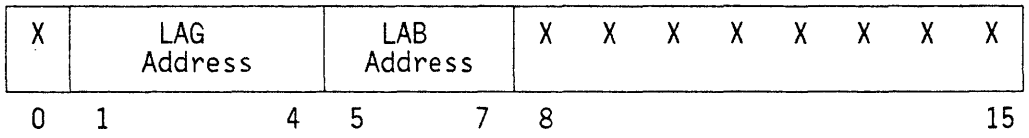
LINE ADDRESSING (3725)

The line addressing scheme used in the controller consists of three elements.

1. Line attachment base (LAB) address
2. Line attachment group (LAG) address
3. Line interface (LI) address

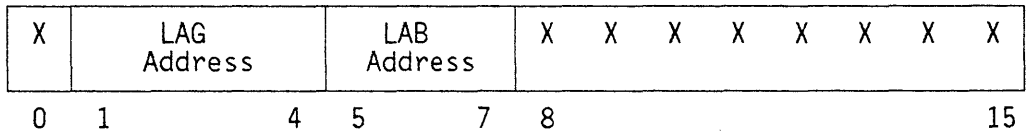
Line Attachment Base (LAB)

The LAB is a three-bit field which is decoded to address one of the LABs. It is contained in bits 5 through 7 of the second halfword of the instruction, as follows:



Line Attachment Group (LAG)

The LAG address is a four-bit field with two possible formats which are used to select one of two groups of 16 lines within the LAB. It is contained in bits 1 through 4 of the second halfword of the instruction, as follows:



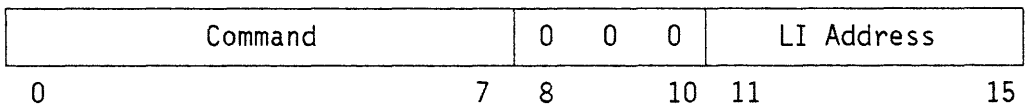
Only three different bit combinations are possible: 0010, 0100, and 0110. The combination 0010 selects the first group of 16 lines of a line attachment base, and 0100 selects the second group.

Note: In the case of the LAB 2, this division corresponds to the two communications scanners.

The third combination, 0110, is used by the Get Line Identification instruction. It is interpreted by all the scanners as a broadcast invitation to the scanner holding the highest priority interrupt, to send its line identification to the CCU (see the Get Line Identification instruction).

Line Interface (LI) Address

The line interface address is a five-bit field which is decoded to address one of the 16 lines of a line interface group (LAG). It is contained in bits 11 through 15 of the register addressed by the R1-field of an IOH instruction, or by the R-field of an IOHI instruction, as follows:



The four high-order bits, 11 through 14, select the line address, while bit 15 selects the transmit or the receive interface in the case of duplex lines. In the case of half duplex lines, bit 15 must be zero.

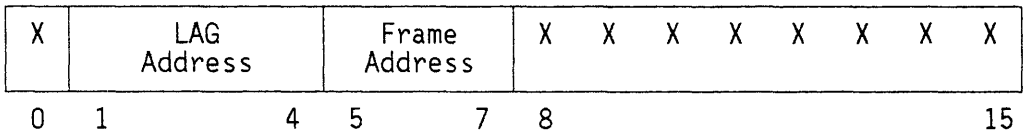
LINE ADDRESSING (3720/21)

The line addressing scheme used in the controller consists of three elements.

1. Frame
2. Line Attachment Group (LAG)
3. Line interface (LI) address

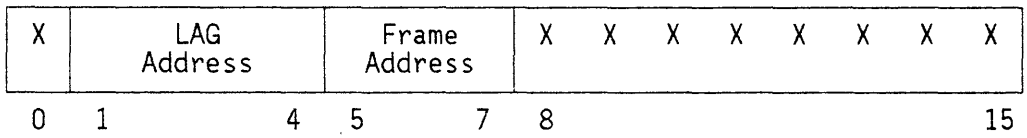
Frame Address

The frame address is a three-bit field which is decoded to address one of the frames. It is contained in bits 5 through 7 of the second halfword of the instruction, as follows:



Line Attachment Group (LAG)

The LAG address is a four-bit field with two possible formats which are used to select one of two groups of lines within the frame. It is contained in bits 1 through 4 of the second halfword of the instruction, as follows:

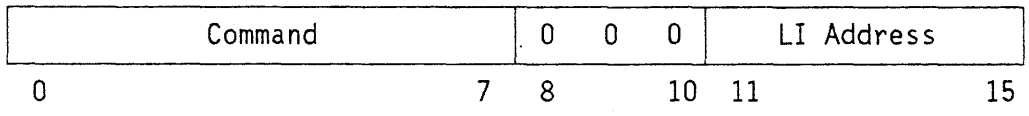


Only three different bit combinations are possible: 0010, 0100, and 0110. The combination 0010 selects the first group of lines in a 3720, and the first scanner in a 3721. 0100 selects the second group of lines in a 3720, and the second scanner in a 3721.

The third combination, 0110, is used by the Get Line Identification instruction. It is interpreted by all the scanners as a broadcast invitation to the scanner holding the highest priority interrupt, to send its line identification to the CCU (see the Get Line Identification instruction).

Line Interface (LI) Address

The line interface address is a five-bit field which is decoded to address one of the lines of a half-frame. It is contained in bits 11 through 15 of the register addressed by the R1-field of an IOH instruction, or by the R-field of an IOHI instruction, as follows:



The four high-order bits, 11 through 14, select the line address, while bit 15 selects the transmit or the receive interface in the case of duplex lines. In the case of half duplex lines, bit 15 must be zero.

RESERVED STORAGE AREAS

Parameter/Status Area

For each line interface, the control program must reserve a **parameter/status area** (PSA) in main storage. The scanner may access the PSA in cycle steal mode, in blocks of 16 bytes or less, to obtain control information from the CCU, or to pass status information to the CCU. The PSA is divided into two areas:

- A parameter area, four full words (16 bytes) long, used for transferring control information from the CCU to the scanner.
- A status area, three full words (12 bytes) long, used for transferring status information from the scanner to the CCU.

The seven full words making up the PSA must be contiguous, but the group of seven words may be located anywhere in storage. PSAs may be grouped together for convenience, or they may be situated in storage according to the requirements of the program. If required, more than one PSA may be prepared for each line. Access to the alternative PSA may then be obtained simply by changing the PSA address in the LVT (see below). This is particularly useful if it is required to change the line characteristics dynamically (from normal mode to character mode, for example). The contents of the PSA depend on the instruction, the mode (normal or character), and the line type. The PSA is discussed in more detail with each command later in this chapter.

Line Vector Table

The line vector table (LVT) consists of 512 fullword locations, two for each of the lines that may be attached. For the 3725 there can be 256 lines, for 3720 there can be 60 lines. There are therefore two entries in the LVT for each line. The LVT entries corresponding to unused line positions must be left empty. In the case of a duplex line, the first (even address) entry must point to the transmit PSA for the corresponding line; the second (odd address) must point to the receive PSA. In the case of a half duplex line, the first entry (even address) must point to the unique PSA that is used for both transmit and receive. The second (odd address) entry is not used.

Each communication scanner knows the starting address of the LVT, and since it also knows the absolute address of each of its telecommunication lines, it can calculate the corresponding LVT address, and therefore locate the address of the corresponding PSA. The relationship between LVT entry and PSA is shown below in Figure 5-1 on page 5-5.

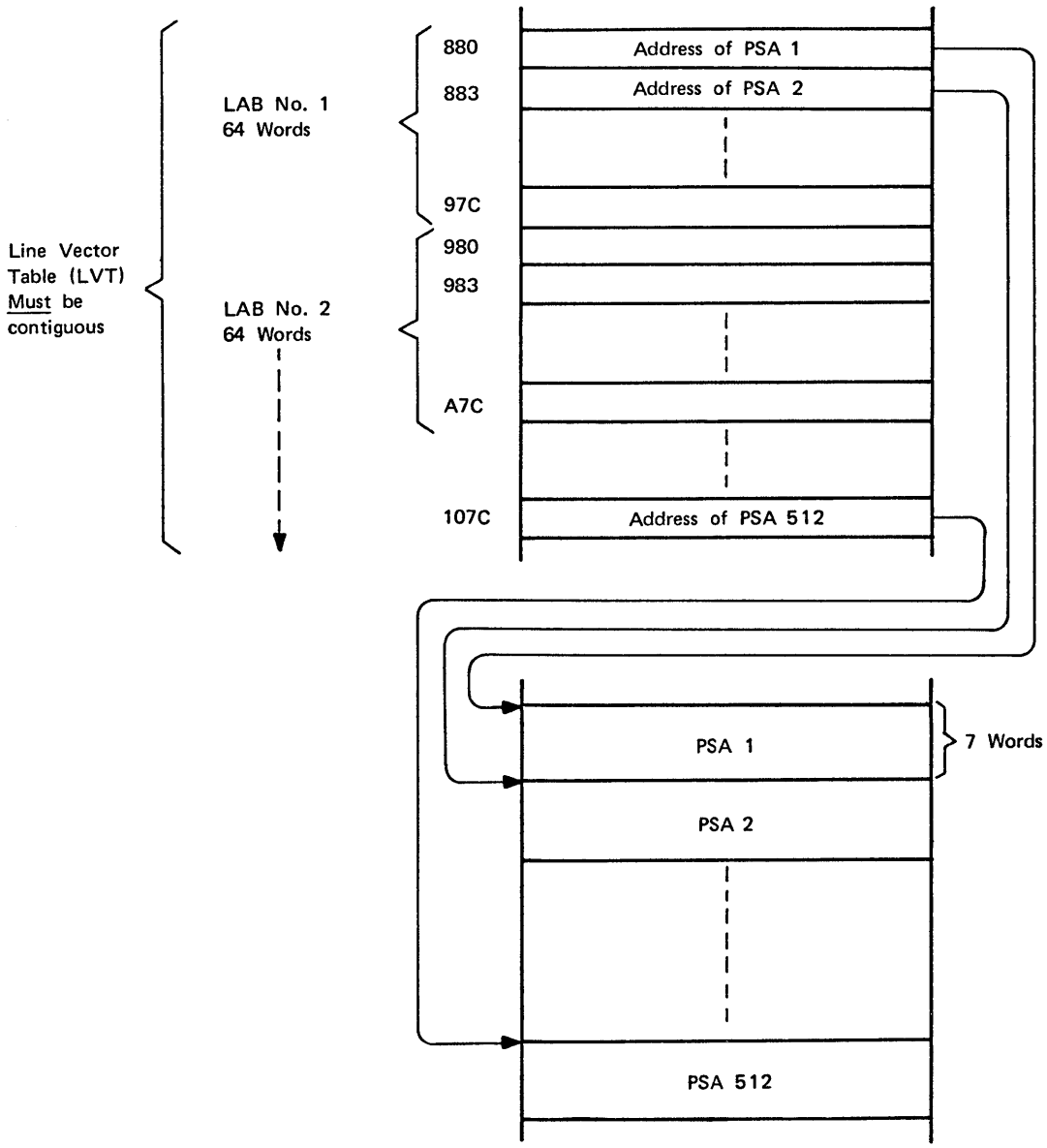


Figure 5-1. Relationship between PSA and LVT

Notes:

1. The LVT may be shorter than 512 words if not all lines are installed. For example, if the absolute address of the highest installed line number is X'15', only twice X'15' = X'2A' full words are required for the LVT.

2. The starting address of the LVT is set to the default value of X'880' each time the scanner is initialized. If necessary, this address may be changed by stopping all operations, moving the table to the required location, and informing all the scanners in turn of the new address by means of the Set Line Vector Table High/Low instructions. Knowing the new LVT start address, the scanner may calculate the new location of the corresponding PSA address.

Note: If the scanner is re-initialized, and an LVT address other than X'880' is used, the LVT address must be re-initialized also.

Buffers and Data Areas

These are areas reserved for the temporary storage of data (and other information) in transit through the controller. They are accessed by the scanner via the cycle steal mechanism. The address of the buffer (or data area) to be used by the scanner is part of the parameter area of the PSA. The format of the buffers depends on whether they are NCP type or 270X Emulation type.

NCP Type Buffer Format

NCP type buffers **must** have the following format:

Buffer Prefix	Offset	Data (up to 256 bytes)
0|1|2|3|4|5|6|7		

The eight bytes of the buffer prefix have the following meaning:

Bytes 0 through 3: These four bytes contain the 21-bit address of the next buffer in the buffer chain.

Bytes 4 and 5: (not used)

Byte 6: This byte contains the offset value. The offset is the number of bytes between the end of the buffer prefix and the first data byte.

Byte 7: This byte specifies the actual number of data bytes in the buffer.

Programming Notes:

1. This is the buffer format used by the IBM Network Control Programs.
2. For the first buffer in a chain, the count and offset are not used. The count and offset values provided in the parameter zone of the PSA (both transmit and receive operations) are used instead; any value placed in bytes 6 and 7 are ignored.
3. For the second and subsequent buffers in a chain, the link pointer, offset, and count, are all valid for a transmit operation.
4. For the second and subsequent buffers in a chain, the link pointer only is valid for a receive operation; the offset is always zero, and the count is taken from the 'set mode data' loaded into the scanner by the Set Mode command for that line.

270X Emulation Type Buffer Format

270X Emulation type buffers are simply data areas located in storage; they have no particular format.

INSTRUCTIONS

The IOH and IOHI instructions are used to move control information between the CCU and the communications scanner. Only four basic instructions are used to control the communications scanner:

- Start line
- Start line initial
- Get line identification
- Set line vector table high/low

One of these instructions, Set Line Vector Table High/Low, is used only exceptionally, to change the start address of the LVT. Another, Start Line Initial, is used only when a line is initialized, either after IPL, or after a dynamic change to a new PSA. Only the Start Line and Start Line Initial instructions include a command.

Note: In the instruction descriptions which follow, the instruction may be indifferently IOH or IOHI. The IOHI instruction is shown for convenience.

Start Line Instruction

First halfword

0	0	0	0	0	R	0	1	1	1	0	0	0	0
0				4	5	7	8						15

Second halfword

0	Line Group* (0010/0100)	LAB Address	0 0 0 0 Start Line	0/1 C/M	0	0/1 N/C	0 Out
0	1 4	5 7	8 11	12	13	14	15

* 0010 selects the first group of 16 lines on a LAB; 0100 selects the second group. In the case of LAB 2, this division corresponds to the two communications scanners.

C/M = CCU/MOSS bit: 0 = initiated by CCU, 1 = initiated by MOSS

N/C : 0 = normal interface, 1 = character mode interface

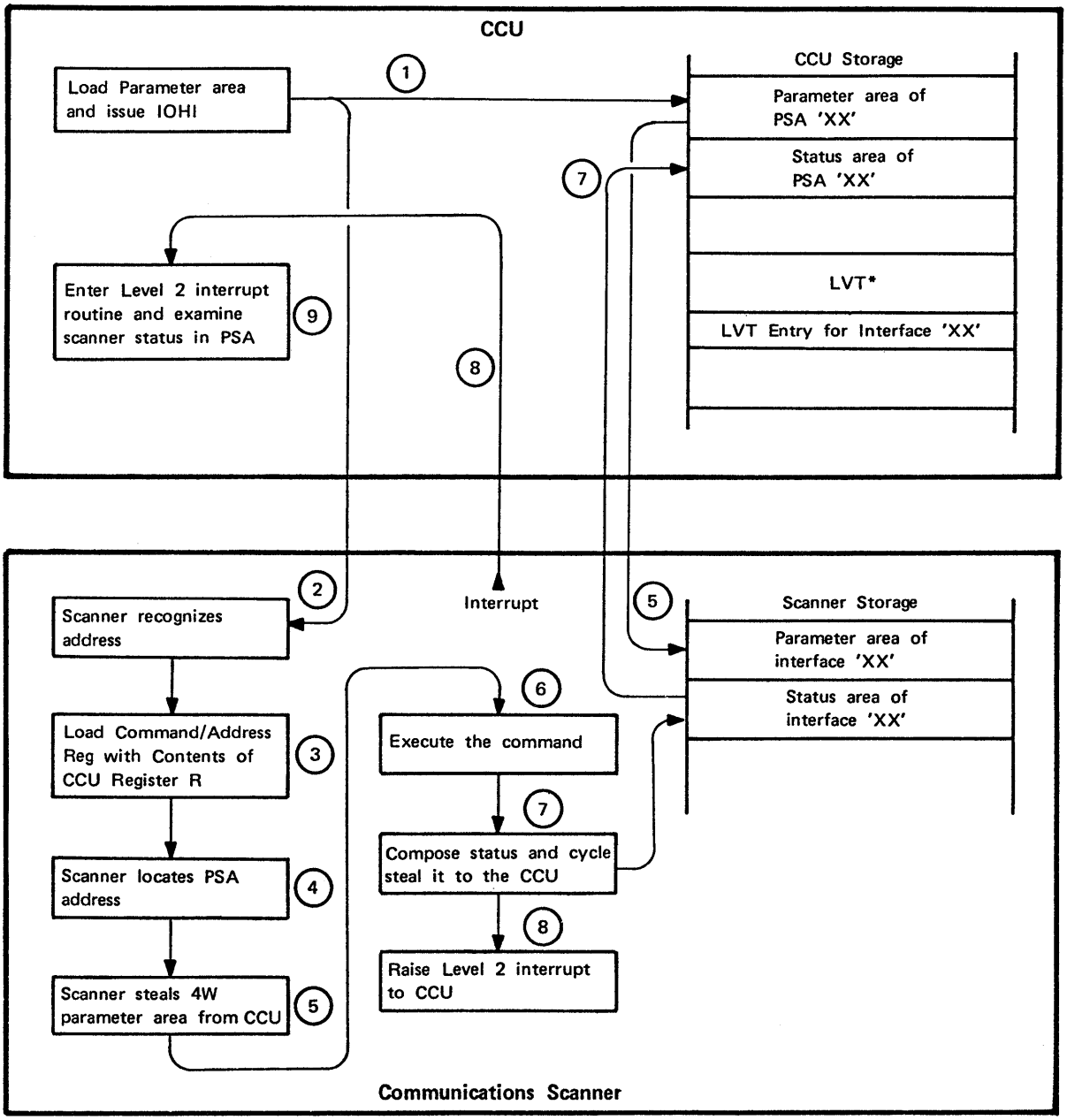
Out = 0: start line is an output operation

Contents of register R (field of first halfword)

Command	0 0 0	Line Address*
0 7	8 10	11 15

* These are the five low-order bits of the line address. Bit 15 selects the transmit/receive interface in the case of duplex lines. In the case of half duplex lines, bit 15 must be zero.

The instruction transfers the contents of register R into the command/address register of the scanner addressed by the second halfword of the instruction, and executes the command on the addressed line. The operation proceeds as shown below in Figure 5-2.



* LVT is not used by the Start Line Instruction

Figure 5-2. Start Line Instruction

1. The CCU loads the parameter area of the PSA and issues the IOHI instruction.

2. The instruction is executed. The scanner which recognizes the address contained in the second halfword interprets bits 8 through 11 as a start line operation. The halfword itself is stored by the scanner.

3. The contents of working register R are transferred to the command/address register of the addressed scanner. The scanner now knows the absolute address of the line and the command to be executed on it.

4. The scanner consults its internal tables and locates the address of the PSA corresponding to the line. The scanner cycle steals up to four fullwords from the parameter area of the PSA. The scanner now has the command and the necessary information to execute it.

5. The scanner executes the command. The commands are very varied, and are treated in detail later. If data transfer is involved, the scanner cycle steals the data to/from the appropriate CCU buffer areas (buffer address in PSA).

6. When the command has been executed, the scanner composes its status (up to three fullwords) and cycle steals it to the status area of the addressed PSA.

7. The scanner raises a level 2 interrupt to the CCU.

8. The CCU accepts the interrupt and examines the status.

   Note: The interrupt mechanism is treated in detail in the Get Line Identification Instruction.

Start Line Initial Instruction

The Start Line Initial instruction is functionally very similar to the Start Line instruction. The only difference in structure occurs in the second halfword of the instruction:

0	Line Group* (0010/0100)	LAB Address	0 0 0 1 Start Line Init	0/1 C/M	0	0/1 N/C	0 Out
0	1 4	5 7	8 11	12	13	14	15

The operation is executed exactly as for the Start Line instruction, except that at step 4 (see above), the scanner uses the absolute line address and the start address of the LVT to calculate the address of the PSA. The scanner then cycle steals the two full words from the LVT that point to the transmit and receive PSAs for a duplex line, or to the common PSA for a half duplex line (the second word is then not used). Operation then continues normally from step 5 (see above).

The Start Line Initial instruction is required on two occasions only:

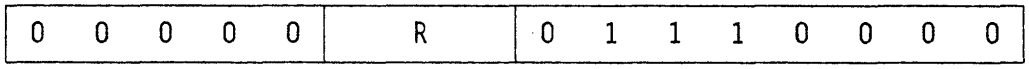
1. After IPL. At this time, the scanner has no means of knowing the address of a particular PSA. The Start Line Initial instruction must be used when a line is addressed for the first time, in order to provide the scanner with this information. Once the line has been initialized, the scanner keeps the PSA address(es) in its own storage.

Note: If a Start Line Initial instruction is used after this time, no program damage can result, but the operation is slowed down by the extra calculation and cycle steals required.

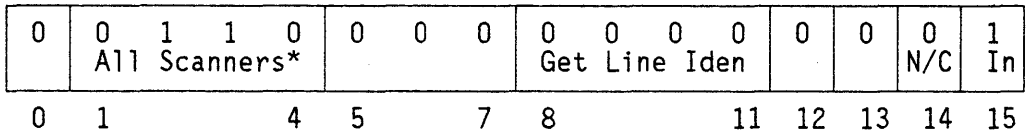
2. After a dynamic switchover to a new PSA. To do this, the CCU first stops all operations on the line in question, loads the corresponding LVT entry with the new PSA address(es), and issues a Start Line Initial instruction to the line.

Get Line Identification Instruction

First halfword



Second halfword

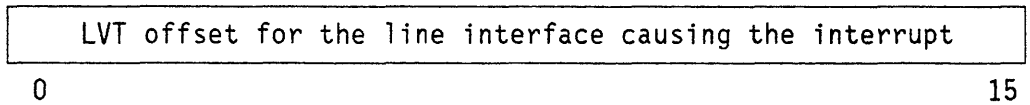


* 0110 indicates a broadcast operation to all scanners.

N/C : always 0 for this instruction

In = 1: Get line identification is an input operation

Contents of register R after execution of the instruction:



Note: This is the meaning for the IBM Network Control and Emulation Programs. It may be any combination of bits enabling the control program to locate the PSA. It is set into the scanner by means of a Set Mode command from the CCU (bytes 8 and 9 of the set mode parameter zone for a half duplex line or for the transmit interface of a duplex line; bytes 10 and 11 for the receive interface of a duplex line).

The effect of the instruction is to transfer an LVTs offset value into the register specified by R. This offset value, when added to the start address of the LVT, forms the PSA address of the line interface with the highest priority interrupt pending.

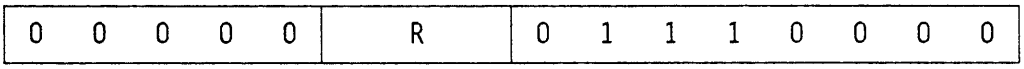
A hardware interrupt priority selection mechanism ensures that the scanner having the line with the highest priority of interrupt is the only one to present its interrupt to the CCU. This mechanism continuously searches for the highest priority interrupt. When the Get Line Identification instruction is issued, the mechanism stops. If two or more scanners have interrupts of the same priority, the mechanism stops on the first scanner with this priority. At the end of the instruction, the Level 2 interrupt is reset.

The operation proceeds as follows:

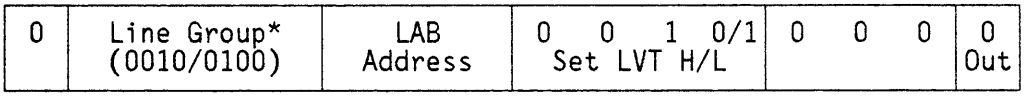
1. The scanner having an interrupt to present to the CCU loads the status area of the PSA with its ending status via the cycle steal mechanism. It then raises a Level 2 interrupt.
2. The CCU accepts the interrupt, but does not yet know which scanner presented it.
3. The CCU issues the Get Line Identification instruction to all scanners.
4. The scanner with the highest priority interrupt pending is selected by the priority selection mechanism.
5. The Line ID for the line interface that raised the interrupt is transferred from the scanner into CCU working register R. The Level 2 interrupt is reset, and the priority selection mechanism restarts.
6. The CCU uses this address to access the status area of the PSA.
7. The CCU examines the status area and takes the necessary action.

Set Line Vector Table High/Low Instruction

First halfword



Second halfword



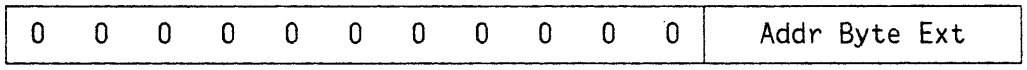
* 0010 selects the first group of 16 lines; 0100 selects the second group. In the case of LAB 2, this division corresponds to the two communications scanners.

Bits 24-27 = 2, Set LVT high; bits 24-27 = 3, Set LVT low

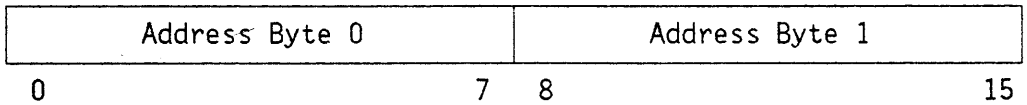
Out = 0: Set LVT high/low is an output operation

Contents of register R (field of first halfword)

1. Set Line Vector Table High



2. Set Line Vector Table Low

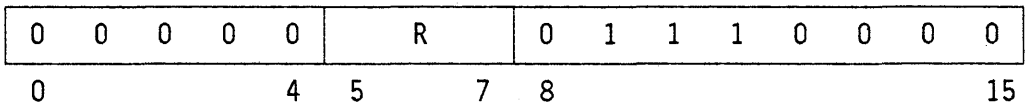


The effect of the instruction is to transfer the address of the Line Vector Table (contained in register R) to the communications scanner. The complete

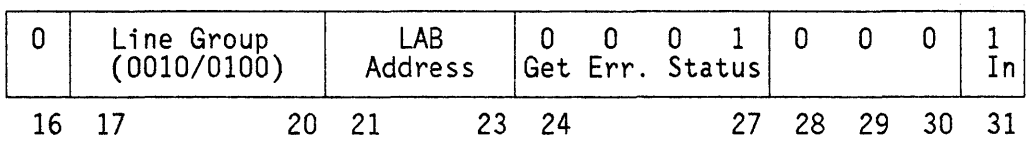
operation requires two instructions, the first to transfer the extension byte of the address, the second to transfer bytes 0 and 1 of the address. This instruction is required only if the address of the LVT is changed; in this case, the new address must be transmitted to **all** the communications scanners.

Get Error Status Instruction

First halfword

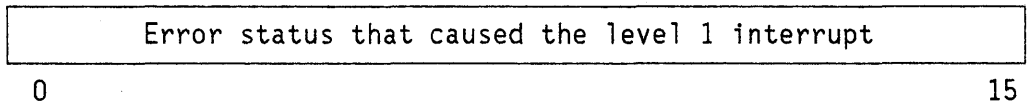


Second halfword



In = 1: Get error status is an input operation

Contents of register R after execution of the instruction:

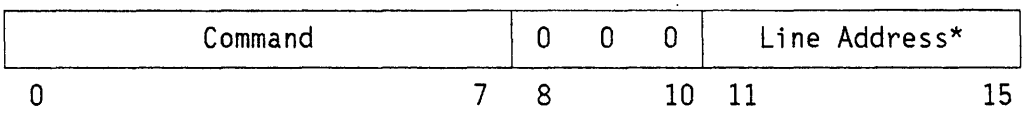


The effect of the instruction is to transfer an error status into the register specified by R.

This command must be issued to the scanner to determine the cause of a level 1 interrupt request coming from the scanner (caused by a hardware or program error). The two-byte status contains information on the source of the error.

COMMANDS

The command is used by the Start Line and Start Line Initial instructions only. It comprises bits 0 through 7 of the contents of register R:



All commands use the parameter/status area (PSA). In addition, some commands (those responsible for data transfer) use a buffer/data area.

Once a command is received by the scanner, it becomes 'outstanding', and remains in this state until the scanner raises a level 2 interrupt request. If a new command is received when a previous command is outstanding, the new command is rejected, and a level 1 interrupt request occurs.

Notes:

1. The 'Halt' and 'Halt Immediate' commands do **not** create an outstanding command condition.

2. The 'Halt' and 'Halt Immediate' commands may be issued at any time, even if another command is outstanding. They are not rejected and no level 1 interrupt occurs.

The commands are divided into five groups:

1. **Common Commands:** These are commands which can be issued in either normal or character mode.

   Note: The 'character mode' is a mode of operation that emulates a 370X Communications Scanner Type 2. Instructions issued in character mode have the character bit (bit 14 of the second halfword) set to 1 in the instruction.

   The common commands are:

Command	Hex
Set Mode	X'01'
Enable	X'02'
Disable	X'03'
Monitor Incoming Call	X'04'
Dial	X'05'
Change	X'06'
Raise DTR	X'08'
Flush Data	X'09'
Reset-D	X'0B'
Reset-N	X'0C'
Halt	X'F0'
Halt Immediate	X'F1'

2. **NCP Commands:** These are commands which can be issued by the NCP or similar programs on SDLC or X.21 lines, or on BSC lines working in normal mode. Channel operations must be in native mode.

   The NCP commands are:

Command	Hex
SDLC Transmit Control	X'10'
SDLC Transmit Data	X'11'
SDLC Transmit Continue	X'1D'
SDLC Receive Monitor	X'12'
SDLC Receive	X'13'
SDLC Receive Continue	X'14'
X.21 Call Request	X'15'
X.21 Monitor Incoming Call	X'16'
X.21 Clear Request	X'17'
NCP BSC Control	X'18'
NCP BSC Transmit	X'19'
NCP BSC Transmit Continue	X'1A'
NCP BSC Receive	X'1B'
NCP BSC Receive Continue	X'1C'

3. **EP Commands:** These are commands which can be issued by the EP and similar programs on BSC lines operating in normal mode. Channel operations must be in 270X emulation mode.

   The EP commands are:

Command	Hex
EP BSC Transmit Initial	X'20'
EP BSC Transmit SYN	X'21'
EP BSC Transmit Data	X'22'
EP BSC Poll	X'23'
EP BSC Receive	X'24'
EP BSC Receive Continue	X'25'
EP BSC Prepare	X'26'
EP BSC Monitor for Phase	X'27'
EP BSC Address Prepare	X'28'
EP BSC Search	X'29'

4. **Character Mode Commands:** These are commands which can be issued by the NCP, the EP, and similar programs on BSC or Start-Stop lines, using the character mode. Channel operations are in native mode or in emulation mode, depending on whether the program is of NCP or EP type.

   The character mode commands are:

Command	Hex
Write ICW (1-byte transfer)	X'40'
Start-Stop Transfer (4-byte burst)	X'41'
Read ICW	X'F2'

5. **Miscellaneous Commands:** This is a small group of commands used to start-stop line tracing for modem testing.

   The miscellaneous commands are:

Command	Hex
IBM 386X/58XX Test	X'2B'
Trace	X'2C'
Stop Trace	X'2D'
Wrap	X'2E'

Summary of Command Operation Modes

The channel adapter operates in either native mode (controller requires only one subchannel address) or in 270X emulation mode (controller requires one subchannel address per line). The communication scanner operates either in normal mode, in character mode (emulating a 370X Communications Scanner Type 2), or in 4-byte burst mode (included with the character mode commands). However, not all combinations are possible. The table below shows the different possibilities.

Program type	Channel mode	Protocol	Commands used	Scanner mode
NCP or similar	Native	SDLC	SDLC	Normal
		X.21	X.21	Normal
		BSC	NCP BSC	Normal
		BSC	Character Mode Write ICW	Character
		S/S	Character Mode Write ICW	Character
		S/S	Character Mode S/S Transfer	Burst
EP or similar	Emulation	BSC	EP BSC	Normal
		BSC	Character Mode Write ICW	Character
		S/S	Character Mode Write ICW	Character
		S/S	Character Mode S/S Transfer	Burst

Commands in Numerical Order

Hex	Command
X'01'	Set Mode
X'02'	Enable
X'03'	Disable
X'04'	Monitor Incoming Call
X'05'	Dial
X'06'	Change
X'08'	Raise DTR
X'09'	Flush Data
X'0B'	Reset-D
X'0C'	Reset-N
X'10'	SDLC Transmit Control
X'11'	SDLC Transmit Data
X'12'	SDLC Receive Monitor
X'13'	SDLC Receive
X'14'	SDLC Receive Continue
X'15'	X.21 Call Request
X'16'	X.21 Monitor Incoming Call
X'17'	X.21 Clear Request
X'18'	NCP BSC Control
X'19'	NCP BSC Transmit
X'1A'	NCP BSC Transmit Continue
X'1B'	NCP BSC Receive
X'1C'	NCP BSC Receive Continue
X'1D'	SDLC Transmit Continue
X'20'	EP BSC Transmit Initial
X'21'	EP BSC Transmit SYN
X'22'	EP BSC Transmit Data
X'23'	EP BSC Poll
X'24'	EP BSC Receive
X'25'	EP BSC Receive Continue
X'26'	EP BSC Prepare
X'27'	EP BSC Monitor for Phase
X'28'	EP BSC Address Prepare
X'29'	EP BSC Search
X'2B'	IBM 386X/58XX Test
X'2C'	Trace
X'2D'	Stop Trace
X'2E'	Wrap
X'40'	Write ICW
X'41'	Start-Stop Transfer
X'F0'	Halt
X'F1'	Halt Immediate
X'F2'	Read ICW

For consistency in the descriptions that follow, each command is broken down into:

1. A brief description of the purpose of the command.
2. The parameter/status area (PSA) in graphic form, followed by a detailed description of the individual bytes.
3. The data area (if any) in graphic form, followed by a detailed description of the individual bytes.
4. Any special notes, conditions, and limitations.

Note: In the tables that follow, byte 0 of the Parameter Zone is the Trace Correlation Counter (TCC). This byte is not used by the communications scanner, but only by the control program when tracing the interface.

COMMON COMMANDS

The common commands are used by the NCP/EP and similar programs on lines operating in both normal mode and character mode (C bit on in the instruction). These commands are used to initiate, enable, and disable lines.

Set Mode Command (X'01')

The Set Mode command is used to personalize the line interface(s). It must be the first command issued to each line after IPL. If any other command is issued first, it is rejected and a level 1 interrupt is raised. The Set Mode command uses a data area to transfer information supplementary to that contained in the PSA. The data includes:

- Data link control and transmission protocol information
- Buffer information
- Address checking information
- Timer information

The Set Mode command may be issued only to an even (transmit) interface. If it is issued to an odd (receive) interface, or to a line that has not been previously initialized by a Start Line Initial command, the command is rejected. It may be issued at any time as long as no other command is outstanding. Set Mode must be issued to all lines regardless of protocol and mode.

Parameter Zone

Word 1	TCC	-	-	-
Word 2	Byte Count	Set Mode Data Address (Bytes X, 0, 1)		
Word 3	Line ID/FDX Transmit ID		FDX Receive ID	
Word 4	EOR 5	EOR 6	EOR 7	EOR 8

Byte Count: This is the number of bytes of Set Mode data to be transferred (16 bytes).

Set Mode Data Address (Bytes X, 0, 1): These three bytes contain the starting address of the supplementary control information for the Set Mode command.

Line ID/FDX Transmit ID/FDX Receive ID: In the case of a half duplex line, only the first halfword is used. It is an identifier that is used by the control program to locate the PSA for that line. For the IBM Network Control Program and Emulation Program, the identifier takes the form of an LVT address containing the address of the corresponding PSA. In the case of a duplex line, two PSAs are used: one for the transmit interface, the other for receive; two identifiers are therefore required.

End of Record (EOR) Characters: These four bytes are used by the Start/Stop Transfer command only. They contain EOR characters 5 through 8. See under the Start/Stop Transfer command for a full discussion.

Status Zone (Normal Mode)

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	Modem-In	Modem-Out
Word 3	-	-	-	-

Status Zone (Character Mode)

Word 1	SCF	CCMD	-	LCS
Word 2	LCD/PCF	SDF	Modem-In	Modem-Out
Word 3	-	-	-	-

Status Control Field (SCF): Contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'01'.

Secondary Status (SES) Field: Contains the secondary status. This field is always X'00' for the Set Mode command in normal mode; in character mode, it is not used.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred. Refer to the end of this chapter for full details.

Modem-In and Modem-Out Fields: These fields are described in detail at the end of this chapter under the heading "Modem Control Fields".

Other Fields: The remaining fields of the status area are not used for the Set Mode command in normal mode. In character mode, two additional fields are used: LCD/PCF and SDF. The meaning of these fields is described under the heading "Character Mode Commands".

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See under the heading "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Set Mode Data Area

The Set Mode data area is a zone of 16 contiguous bytes containing supplementary information about a specific line. It has the following configuration:

Byte	Meaning			
0/3	Disable Timeout		Control 0	Control 1
4/7	Control 2	Control 3	Buffer size	-
8/11	Secondary station address		Response timeout	
12/15	Enable/dial timeout		Receive text timeout	

Bytes 8 through 15 are used only by the NCP or similar programs.

The applicability of the different fields depends on the command. In the tables that follow, BSCN = NCP BSC; BSCE = EP BSC mode; BSCC = BSC in character mode (NCP or EP); S/SC = start-stop in character mode (NCP or EP).
The meaning of the individual fields is:

Bytes 0 and 1 - Disable Timeout Value:
These two bytes are valid for SDLC, X.21, BSCN, BSCE, BSCC, and S/SC. They contain the timeout value of a timer which is started after having dropped the DTR signal in order to monitor the dropping of the DSR signal from the modem.
The timeout value is in increments of 0.1 second.

Byte 2: Control Byte 0

		Applicable to:					
Bit	Meaning	SDLC	X.21	BSCN	BSCE	BSCC	S/SC
0	Duplex	*	*				
1	230 kbps line/tributary support	*			*		
2	270X emulation mode				*	*	*
3	Transmit 2 flags/interrupt mode	*			*	*	
4	ITB is data				*		
5	EIB mode				*		
	58xx modem	*		*			
6	Transmit flags/option 1 modem	*			*	*	*
7	Primary station/2703 mode	*			*	*	*

The bits of control byte 0 have the following meanings:

Bit 0 - Duplex (NCP only):
This bit indicates that two line identifiers are present in the PSA, and that both transmit and receive interface addresses are to be used if the line is specified as SDLC in Control Byte 2, bits 0-3.

Bit 1 - 230 Kbps Line (SDLC only):
This bit indicates that the speed of this line is between 230 and 256 kilobits per second inclusive.

Bit 1 - Tributary Support (EP BSC only):
This bit indicates that the controller is defined as a master or tributary station in a non-centralized multipoint network.

Bit 2 - 270X Emulation Mode:
This bit, when on, indicates that the line is operating in 270X emulation (EP) mode.

Bit 3 - Transmit Two Flags (SDLC only):
This bit, when on, indicates that two flags must be transmitted before the A-field.

Bit 3 - Interrupt Mode (EP only):
This bit is used to indicate automatic answering on a switched line.

Bits 4 and 5 - ITB is Data and EIB mode (EP only):
These two bits work together as shown in the following table:

EIB Mode	ITB= Data	Receive	Transmit
0	0	Check BCC, but do not generate EIB (*)	Compute BCC, send it after ITB, ETB and ETX
0	1	Treat ITB as data	Treat ITB as data
1	-	Ignore ITB = Data bit, check BCC, generate EIB after ITB (*), ETB and ETX.	Ignore ITB = data bit, compute BCC, send it after ITB, ETB, and ETX. Do not skip the character after ITB.

* If a data check or overrun is detected, the error is reported in the SES or the SCF at the end of the command.

Bit 5 - 58xx modem (NCP only):
This bit, when on, indicates that NCP should report in SES bit 5 if the TI lead is on.

Bit 6 - Transmit Flags between Frames (SDLC only):
This bit, when on, indicates that if the 'Turn Line Around' modifier is off for an SDLC Transmit Control or SDLC Transmit Data command, continuous flags must be transmitted once the frame has been sent. If the bit is off, continuous idle characters (X'FF') must be transmitted.

Bit 6 - Option 1 Modem (EP only):
This bit indicates that the data set ready (DSR) line is permanently activated.

Bit 7 - Primary/Secondary Station (NCP only):
On SDLC lines this bit, when on, indicates that the controller is the primary station on the line.
If the bit is off, it indicates that the controller is the secondary station.

Bit 7 - 2703 Mode (EP only):
This bit indicates that, when enabling a leased line, DTR is turned on, but no timer is started to monitor for DSR.

Byte 3: Control Byte 1

		Applicable to:					
Bit	Meaning	SDLC	X.21	BSCN	BSCE	BSCC	S/SC
0	Generate answer tone/TWX	*		*		*	*
1	Switched line	*	*	*	*	*	*
2	Ring indicator mode	*		*	*	*	*
3	NRZI/secure line	*	*				*
4	Turn with RTS on (duplex)	*		*	*	*	*
5	Transmit with new sync	*		*	*	*	
6	Ignore bad pad				*		
7	Swift support/X.21 mode		*		*		

Note:
For EP BSC, the "Transmit with New Sync" function is performed using PCF state X'A'.

The bits of control byte 1 have the following meaning:

Bit 0 - Generate Answer Tone:
This bit indicates that on a switched line, an answer tone must be generated when completing a call-in connection.
The answer tone is generated by transmitting three seconds of continuous space signals.

Bit 0 - TWX Teletype:
This bit, when on, indicates that the line is connected to a leased Teletype terminal (start-stop only).

Bit 1 - Switched Line:
This bit indicates that the line is part of a switched facility.
If, in addition, the LIC is a LIC type 4 (X.21 interface), this bit indicates that the line is on a switched X.21 interface.

Bit 2 - Ring Indicator Mode:
This bit, when on, specifies that on a switched V24 interface, an incoming call is detected when the ring indicator line rises.
When the bit is off, incoming calls are detected via the data set ready (DSR) line.
This choice depends on the type of modem installed.

Bit 3 - Non-Return-to-Zero-Inverted (NRZI):
When on, this bit indicates that on an SDLC link, the data is to be transmitted in non-return-to-zero-inverted mode.

Bit 3 - Secure Line:
This bit, when on, specifies that on a start-stop switched line, the Data Carrier Detector must be continuously monitored when the line is receiving.

Bit 4 - Turn with RTS On:
This bit, when on, indicates a four-wire communication facility using half duplex protocol, so that the request to send (RTS) signal stays on permanently to avoid turnaround delays (synchronous equipment only).

Bit 5 - Transmit with New Sync:
This bit, when on, indicates that the "new sync" modem interface lead must be controlled.

The bit is valid only if the modem connected to the interface has the new sync feature, and if the communications scanner is the master (primary) station (as specified by byte 4, bit 5) of a 4-wire line (turn with RTS on, byte 3, bit 4 must be on) on which multipoint line control is used.

The scanner raises the "new sync" lead in the attached modem immediately before starting character transmission, and drops it when the last character has been transmitted.

Bit 6 - Ignore Bad Pad:
Indicates that on EP BSC lines, bad pad characters are to be ignored.

Bit 7 - Swift Support:
This bit, when on, indicates that the STX character is included in the BCC.

Bit 7 - X.21 Mode:
This bit, when on, indicates that the line is working in the X.21 mode.

Byte 4: Control Byte 2

The bits of control byte 2 have the following meaning:

Bits 0-3 - Line Control Definer (LCD) Field:
This field specifies the line protocol.
The four bits are decoded as one hexadecimal digit having the following meaning:

		Applicable to:					
Value	Meaning	SDLC	X.21	ESCN	BSCE	BSCC	S/SC
0	Start-stop 9/6						*
1	(not used)						
2	Start-stop 8/5						*
3	Autocall	*		*	*		
4	Start-stop 9/7						*
5	Start-stop 10/7						*
6	Start-stop 10/8						*
7	Start-stop 11/8						*
8	(not used)						
9	SDLC	*	*				
A	(not used)						
B	(not used)						
C	BSC (EBCDIC)			*	*	*	
D	BSC (ASCII)			*	*	*	
E	BSC (transparent ASCII)				*	*	
F	BSC (ASCII) without translation						

Bits 4-7 - Buffer Prefix Size Field:
the bits of this field specify the size of the prefix area of control program buffers.
The prefix may contain a link pointer to the next buffer in a chain, an offset, and a byte count.
The buffer prefix may be set to any value between 0 and 15 bytes, but is normally eight bytes long.

It is also used in 270X emulation mode for wrap testing of EP BSC and character mode BSC and S/S.

Byte 5: Control Byte 3

Bit	Meaning	Applicable to: SDLC	X.21	BSCN	BSCE	BSCC	S/SC
0	Synchronous line	1	1	1	1	1	0
1	) Line speed	*	*	*	*	*	*
2	) if business machine clock	*	*	*	*	*	*
3	) is used; see below.	*	*	*	*	*	*
4	)	*	*	*	*	*	*
5	External clock	*	*	*	*	*	*
6	Data rate select	*	*	*	*	*	*
7	Medium speed local attachment - 3725	*	*	*	*	*	*
	Take line speed from PLS file - 3720	*	*	*	*	*	*

The bits of byte 1 have the following meaning:

Bit 0 - Synchronous Line:
Must always be on for SDLC and BSC protocols and off for start-stop.
Bits 1 through 4 - Line Speed:
These bits set the speed of the internal clock for the line if business machine clocking is specified (external clock bit, bit 5, is off).
The four bits have the following meaning:

Synchronous lines

Bits 1	2	3	4	Speed (bps)
0	0	0	1	50
0	0	1	0	110
0	1	0	0	134.5
0	1	1	1	200
1	0	0	0	300
1	0	1	1	600
1	1	0	1	1200
0	0	1	1	special*
1	1	1	0	special*

Start/stop lines

Bits 1	2	3	4	Speed (bps)
0	0	0	0	50
1	1	1	0	75
1	1	0	1	100
0	0	1	1	110
0	1	0	1	134.5
0	1	1	0	200
1	0	0	1	300
1	0	1	0	600
1	1	0	0	1200
0	0	0	1	2400
0	0	1	0	4800
0	1	0	0	9600
0	1	1	1	19200
1	0	1	1	special*
1	1	1	1	special*
1	1	1	1	**

* non-standard line speed. ** asymmetrical (75 bps from terminal, 1200 bps to terminal).

Bit 5 - External Clock: This bit, if on, indicates that external (modem) clocking is used for this line. If off, it indicates that business machine clocking is used; the speed is then determined by bits 1-4 above.

Bit 6 - Data Rate Select: This bit, if on, selects a high data rate for the modem. If off, it selects a low data rate.

Bit 7 - Medium Speed Local Attachment (3725 only): This bit, when on, indicates that no DCE is required, and that the terminal is directly connected to the controller.

Bit 7 - Take Line Speed from PLS (3720 only): The line speed is taken from the Programmable Line Speed table loaded in the scanner.

Byte 6 - Buffer Size: This byte contains the maximum size of the data area available in a control program buffer for receive operations. It is equal to the buffer length minus the buffer prefix length. It is also used in 270X emulation mode for control lead wrap testing in EP BSC, and in character mode BSC and start-stop

Bytes 7 through 15 (SDLC Only):

Bytes 7: This byte is not used by SDLC.

Bytes 8 and 9 - SDLC Secondary Station Address: This two-byte field is valid for SDLC and X.21 only. The primary/secondary station bit (byte 2, bit 7) must be set to 0 to indicate a secondary station. It is used as the SDLC address compare field.

Bytes 10 and 11 - Reply Timeout Value: This two-byte field is valid for SDLC and X.21 only. The timeout is started when a transmission has ended, but a reply is awaited. The timeout value is in increments of 0.1 second.

Bytes 12 and 13 - Enable/Dial Timeout Value: This two-byte field is valid for SDLC and X.21 only. The timeout value is in increments of 0.1 second. It is used for two different timeouts:

1. Enable timeout. These two bytes contain the timeout value of a timer which is started after having raised the DTR signal, in order to monitor for the rise of the DSR signal from the modem.

2. Dial timeout. These two bytes contain the timeout value of a timer which is used to detect the failure of an 'Abandon Call and Retry' signal coming from an Automatic Calling Unit.

Bytes 14 and 15 - Receive Text Timeout Value: This two-byte field is valid for SDLC and X.21 only. It contains the timeout value of a timer which is started when reception has begun. It is refreshed every time a buffer is filled, and is reset to zero at the end of a frame. The timeout value is in increments of 0.1 second.

Bytes 7 through 9 (EP BSC Only):

Byte 7 - Tributary Support Selection Address: This byte contains the Tributary Address used by the EP BSC Address Prepare and EP BSC Search commands.

Byte 8 - Tributary Support Group Address; This byte contains the Group Address used by the EP BSC Address Prepare and EP BSC Search commands.

Byte 9 - Tributary Support Poll Address: This byte contains the Poll Address used by the EP BSC Address Prepare command.

Bytes 7 through 11 (Start-Stop Transfer Only):

Byte 7 - EOR Character Count: This byte indicates the number of End Of Reception (EOR) characters (up to eight) passed to the scanner by the Set Mode command. The EOR characters are contained in bytes 8 through 11 of the Set

Mode data (EOR characters 1 through 4), and in the last four bytes of the parameter zone (EOR characters 5 through 8).

Bytes 8 through 11 - EOR Characters: Up to four EOR characters (EOR bytes 1 through 4) contained in these fields may be transferred to the scanner. See the Start/Stop Transfer command for a full discussion.

Change Command (X'06')

The Change command may be used to update up to ten consecutive bytes of Set Mode data, for example:

- Change the internal (business machine) clock speed for a multiple terminal access (MTA) line. A multiple terminal access line is a line to which two or more terminals working at different speeds are connected.
- Change the timeout values.

If the Change command is issued to an interface that has not already received a Set Mode command, it is rejected. It may only be issued to an even (transmit) interface. If it is issued to an odd (receive) interface, the command is rejected. The Change command must be used with great care; the program should ensure that there is no traffic over the line when the change takes place. The result of the command is unpredictable if data is being exchanged over the line when the change takes place.

Programming Note:

If the change is issued on an EP line, the line is set to the No-Op state; the change is then executed, and the line is left in the No-Op state.

Parameter Zone (All commands except Start/Stop Transfer)

Word 1	TCC	-	-	-
Word 2	Change Count	Change Start	1st Byte	2nd Byte
Word 3	3rd Byte	4th Byte	5th Byte	6th Byte
Word 4	7th Byte	8th Byte	9th Byte	10th Byte

Parameter Zone (Start/Stop Transfer Command only)

Word 1	TCC	-	-	-
Word 2	Change Count	Change Start	1st Byte	2nd Byte
Word 3	3rd Byte	4th Byte	5th Byte	6th Byte
Word 4	EOR 5	EOR 6	EOR 7	EOR 8

Change Count: This byte contains the count of the number of bytes to be changed. It can take any value between one and 10. If greater than 10, no error is posted, but the change count is assumed to be 10; if it is zero, no change is made.

Note: For the Start/Stop Transfer command only, the change count must not exceed six, the last four bytes of the parameter zone being reserved for End Of Record characters 5 through 8. The last four bytes of the parameter zone are changed to these values.

Change Start: This byte contains the number of the first byte of Set Mode data to be changed. It can take any value from zero through 15.

Change Data Bytes 1 through 10: These bytes contain up to ten bytes of data to be changed.

Status Zone (Normal Mode)

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	Modem-In	Modem-Out
Word 3	-	-	-	-

Status Zone (Character Mode)

Word 1	SCF	CCMD	-	LCS
Word 2	LCD/PCF	SDF	Modem-In	Modem-Out
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information which describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, X'06'.

Secondary Status (SES) Field: Contains the secondary status. This field is always X'00' for the Change command in normal mode; in character mode, it is not used.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Modem-In and Modem-Out Fields: These fields are described in detail at the end of this chapter under the heading "Modem Control Fields".

Other Fields: The remaining fields of the status area are not used for the Change command in normal mode. In character mode, two additional fields are used: LCD/PCF and SDF. The meaning of these fields is described under the heading "Character Mode Commands".

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See under the heading "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Enable Command (X'02')

The Enable command is used to prepare the line for data transfer. It must be issued to a line before any data transfer commands can be executed on that line.

The effect of the Enable command depends on the nature of the interface:

1. **Leased EIA RS232/CCITT V24, CCITT V35, Bell 303 in NCP environment:**

   Data terminal ready (DTR) is turned on, and the enable timeout is started. The modem should respond with data set ready (DSR). For SDLC duplex and not secondary station, request to send (RTS) is turned on, and the enable timeout is started to monitor for clear to send (CTS).

   Note: In the case of a secure line, after DSR rises, the scanner monitors for RLSD.

2. **Leased EIA RS232/V24, CCITTV35, Bell 303 in an EP environment:**

   Processing is different, depending on the state of the "2703 Mode" and "Leased TWX" bits, and whether the line is BSC or Start-Stop:

   a. 2703 Mode and not TWX: DTR is turned on, no timer is started, DSR is not monitored. If in addition, the line is duplex, RTS is raised, and a 1-second timeout is started. When CTS rises, the line is set to the monitor for phase state. If a timeout occurs, the line is set to No-Op, and the LCS is set to F2.

   b. 2703 Mode and TWX: DTR is turned on, no timer is started, DSR is monitored. After DSR rises, RTS is turned on, and a one-second timeout is started; when CTS rises, the line is set to monitor for phase. If the timeout occurs, the line is set to No-Op, and the timeout condition is noted in the LCS. If in addition, the line is duplex, RTS is raised, and a one-second timeout is started. When CTS rises, the line is set to the monitor for phase state. If a timeout occurs, the line is set to No-Op, and the LCS is set to F2.

   c. Not 2703 Mode and TWX: DTR is turned on, no timer is started, DSR is monitored. After DSR rises, RTS is turned on, and a two-second timeout is started; when CTS rises, the line is set to monitor for phase. If the timeout occurs, the timeout condition is noted in the LCS. If in addition, the line is duplex, RTS is raised, and a two-second timeout is started. When CTS rises, the line is set to the monitor for phase state. If a timeout occurs, the line is set to No-Op, and the LCS is set to F2.

   d. Not 2703 Mode, not TWX, and BSC: DTR is turned on, no timer is started, DSR is monitored. If in addition, the line is duplex, RTS is raised, and a two-second timeout is started. When CTS rises, the line is set to the monitor for phase state. If a timeout occurs, the line is set to No-Op, and the LCS is set to F2.

   e. Not 2703 Mode, not TWX, and S/S: DTR is turned on, no timer is started, DSR is not monitored. If in addition, the line is duplex, RTS is raised, and a two-second timeout is started. When CTS rises, the line is set to the monitor for phase state. If a timeout occurs, the line is set to No-Op, and the LCS is set to F2.

3. **Switched V24 Interface with Autocall Unit (ACU):**

   a. Issue a "Raise DTR" command on the data line.

b. Issue a "Dial" command on the ACU interface.

c. Issue an "Enable" command on the data line to complete the connection. In NCP, the "Enable" command turns DTR on and monitors for DSR; no timer is started. In EP, the "Enable" command performs the same processing as for a leased line.

4. **Switched V24 Interface with Manual Call:** DTR is turned on, and the modem is monitored for DSR. No timeout is started.

5. **X.21 Interface:** On a **leased** X.21 interface, the C-lead is turned on (C = ON); the modem should reply by raising the I=lead (I = ON).

   Note: There is no Enable command for a **switched** X.21 interface.

Line Initialization

Once the Enable command has been successfully completed, the line may or may not be set to receive mode:

- **SDLC.** The receive interface is not started. It is started only when the first Receive command is received, or at turnaround time on a Transm command.
- **NCP BSC.** The receive interface is set to monitor for phase.
- **EP BSC Duplex.** RTS is raised on the receive interface, and a two-second timer is started to monitor the rise of CTS. When CTS rises, the receive interface is set to monitor for phase.
- **EP BSC Half Duplex.** The receive interface is set to monitor for phase.
- **NCP with Character Mode Start-Stop.** The receive interface is not started. The Write ICW command issued after Enable (to reset the service request) can start the receive interface, if specified in the PCF.
- **EP with Character Mode Start-Stop.** RTS is raised on the receive interface, and a 25.6-second timer is started to monitor the rise of CTS. When CTS rises, the receive interface is set to monitor for phase.

Incoming (received) data is stacked in a buffer (the line interface buffer) in the scanner, waiting for a Receive command to transfer it to the control program buffer.

Note: A "Lost Data" condition occurs if:

- the line interface buffer is filled with data and no Receive command is issued.
- the link is half duplex and a Transmit command is received before the Receive command.

The Transmit command is terminated with command rejected (line receiving) in the ending status.

The Enable command may be issued only to an even (transmit) interface. If it is issued to an odd (receive) interface, the command is rejected. It is also rejected if no Set Mode command has been previously received.

An "Enable" command must be executed before any data transfer commands may be serviced (except for lines equipped with an Option 1 modem, for which a "Reset-N" command may be issued).

Modem or internal errors occurring after completion of the Enable command are stacked in the scanner and are passed on to the control program in the ending status of the next command.

Parameter Zone

Word 1	TCC	-	-	-
Word 2	-	-	-	-
Word 3	-	-	-	-
Word 4	-	-	-	-

Status Zone (Normal Mode)

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	Modem-In	Modem-Out
Word 3	-	-	-	-

Status Zone (Character Mode)

Word 1	SCF	CCMD	-	LCS
Word 2	LCD/PCF	SDF	Modem-In	Modem-Out
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information which describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'02'.

Secondary Status (SES) Field: Contains the secondary status. This field is always X'00' for the Enable command in normal mode; in character mode, it is not used.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Modem-In and Modem-Out Fields: These fields are described in detail at the end of this chapter under the heading "Modem Control Fields".

Other Fields: The remaining fields of the status area are not used for the Change command in normal mode. In character mode, two additional fields are used: LCD/PCF and SDF. The meaning of these fields is described under the heading "Character Mode Commands".

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Disable Command (X'03')

The Disable command is used to reset a line, the associated modem, and the scanner control block information. The line is placed in the disabled state, and an Enable, Monitor Incoming Call, or a Reset-N command must be issued before it may be used to transfer data again.

The command may be issued only to an even (transmit) interface. If it is issued to an odd (receive) interface, the command is rejected.

Parameter Zone

Word 1	TCC	-	-	-
Word 2	-	-	-	-
Word 3	-	-	-	-
Word 4	-	-	-	-

Status Zone (Normal Mode)

Word 1	SCF	CCMD	-	LCS
Word 2	-	-	Modem-In	Modem-Out
Word 3	-	-	-	-

Status Zone (Character Mode)

Word 1	SCF	CCMD	-	LCS
Word 2	LCD/PCF	SDF	Modem-In	Modem-Out
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information which describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'03'.

Secondary Status (SES) Field: Contains the secondary status. This field is always X'00' for this command in normal mode; in character mode, it is not used.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Modem-In and Modem-Out Fields: These fields are described in detail at the end of this chapter under the heading "Modem Control Fields".

Other Fields: The remaining fields of the status area are not used for the Disable command in normal mode. In character mode, two additional fields are

used: LCD/PCF and SDF. The meaning of these fields is described in "Character Mode Commands".

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See under "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

1. The Set Mode data is not affected by the Disable command.

2. A Set Mode command is not required before the next Enable command.

3. The disable command should not be used on EP lines connected to modems that do not drop DSR when DTR is dropped.

Dial Command (X'05')

The Dial command is used to perform automatic dialing of a remote station via an autocall unit (ACU). It has no meaning for a manual call. The Dial command must be issued on the autocall interface (as specified by a Set Mode command) of a data line. Before issuing a Dial command on an autocall interface, the control program must issue a Raise DTR command on the associated data line in order to turn DTR on. When the Raise DTR command is complete, the Dial command must be issued. When the Dial command is complete, an Enable command must be issued on the data line. The data line interface and the ACU interface may be on different scanners. A timeout is used to monitor the different phases of the dial operation. Its value is 51.2 seconds in EP; in NCP, it is as specified in the Set Mode data.

The dial command must be issued to the even interface only; if it is issued to the odd interface, it is rejected.

The digits to be dialed must be contained in a single buffer (character mode) or a chain of buffers (normal mode). The maximum number of digits allowed is 64. The digits must be contained in bits 4 through 7 of the data byte in hexadecimal form; valid digits are X'0' through X'9', plus X'C' and X'D'. Digits X'0' through X'9' represent the value of the digit to be dialed; X'C' is the end-of-number (EON) character, used to inform the ACU that the last digit has been provided; X'D' is a separator to tell the ACU to wait for the second dial tone.

Notes:

1. The ACU must be equipped with the appropriate feature in order to use the X'C' and X'D' characters. If the ACU is not equipped to use the separator character, a dialing pause may be introduced into the dialing sequence in order to receive the second dial tone. To do this, two special characters may be introduced into the dialing buffer at the appropriate place: X'FF' introduces a 60-second pause; X'FA' introduces a one second pause. Thus, X'FF' followed by 5 X'FA' characters introduces a total delay of 65 seconds.

2. No dialing retry is done by the scanner if the call is unsuccessful.

Parameter Zone

Word 1	TCC	Modifiers	Offset	
Word 2	Byte Count	DD Buffer Pointer (Bytes X, 0, 1)		
Word 3	-	-	-	-
Word 4	-	-	-	-

Modifier Byte: Only one modifier bit is used, with the following meaning:

Bit 0 - NCP Type Buffer: This bit, if off, indicates that the data to be transferred to or from the control program is in an NCP type buffer whose address is contained in the "First Transmit Buffer Pointer". If this bit is on, the data is not in an NCP type buffer, but in a data area whose address is contained in the "First Transmit Buffer Pointer" (in this case, the buffer prefix and buffer offset are not used).

Offset: This is the number of bytes between the dialing digit buffer address contained in word 2 and the start of the dialing digits.

Byte Count: This is the number of dialing digits contained in the buffer.

Dialing Digit Buffer Address (Bytes X, 0, 1): These three bytes hold the address of the buffer containing the dialing digits.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	Modem-In	Modem-Out
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information which describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'05'.

Secondary Status (SES) Field: Contains the secondary status. This field is always X'00'.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Modem-In and Modem-Out Fields: These fields are described in detail at the end of this chapter under the heading "Modem Control Fields".

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Monitor Incoming Call Command (X'04')

The Monitor Incoming Call command is used on switched V24 interfaces to place the line in answer mode, that is, ready to accept an incoming call. Depending on the modem type, an incoming call is detected via the ring indicator (RI) signal, or via the data set ready (DSR) signal. The option is chosen via byte 3, bit 2 (Ring Indicator Mode) of the Set Mode data. If Set Mode data, byte 3, bit 0 is on, an answer tone is generated and RTS is raised.

The command may be issued only to an even (transmit) interface. If it is issued to an odd (receive) interface, it is rejected.

Parameter Zone

Word 1	TCC	-	-	-
Word 2	-	-	-	-
Word 3	-	-	-	-
Word 4	-	-	-	-

Status Zone (Normal Mode)

Word 1	SCF	CCMD	-	LCS
Word 2	-	-	Modem-In	Modem-Out
Word 3	-	-	-	-

Status Zone (Character Mode)

Word 1	SCF	CCMD	-	LCS
Word 2	LCD/PCF	SDF	Modem-In	Modem-Out
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information which describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'04'.

Secondary Status (SES) Field: Contains the secondary status. This field is always X'00' for this command in normal mode; in character mode, it is not used.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Modem-In and Modem-Out Fields: These fields are described in detail at the end of this chapter under the heading "Modem Control Fields".

Other Fields: The remaining fields of the status area are not used for the Change command in normal mode. In character mode, two additional fields are used: LCD/PCF and SDF. The meaning of these fields is described under the heading "Character Mode Commands".

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

1. After successful execution of the command (ending status = "Connection Established"), the line is set to receive status.

2. The status of the Modem-In and Modem-Out leads is available in the status area for both normal and character mode commands.

3. Modem or internal errors occurring after completion of the Monitor Incoming Call command are stacked in the scanner, and are passed on to the control program in the ending status of the next command.

Flush Data Command (X'09')

The Flush Data command takes the place of a Receive or Receive Continue command. It is used to clear out the data received on a line until an ending condition such as flag, line goes idle, or ETB/ETX/ENQ (BSC), timeout, modem error, is reached. If the line is already idle when the Flush Data command is given the command will not end.

The command must be issued only to a line in normal mode; if it is issued to a line in character mode, it is rejected.

Parameter Zone

Word 1	TCC	-	-	-
Word 2	-	-	-	-
Word 3	-	-	-	-
Word 4	-	-	-	-

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	-	-
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information which describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'09'.

Secondary Status (SES) Field: Contains the secondary status (always X'00' for this command).

Line Communication Status (LCS) field: Refer to the end of this chapter for full details of this field.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Reset-D Command (X'0B')

Reset-D means "Reset and Disable". The command is used to set the line to the disabled state. If a wrap is in progress, it is terminated. The line must have been previously initialized by means of a "Set Mode" command, and there must be no outstanding command on either interface. If this is not the case, a Level 1 interrupt occurs with an error status type 3.

This command may be issued only to an even (transmit) interface. If it is issued to an odd (receive) interface, it is rejected. This command is also rejected if a Set Mode command has not previously been issued.

Parameter Zone

Word 1	TCC	-	-	-
Word 2	-	-	-	-
Word 3	-	-	-	-
Word 4	-	-	-	-

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	Modem-In	Modem-Out
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information which describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'0B'.

Secondary Status (SES) Field: Contains the secondary status. This field is always X'00' for this command in normal mode; in character mode, it is not used.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Modem-In and Modem-Out Fields: These fields are described in detail at the end of this chapter under the heading "Modem Control Fields".

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

1. The Reset-D command should not be issued on lines equipped with Option 1 modems.

2. The Reset-D command does not have an internal wait, and does not wait for DSR to drop.

3. If the Reset-D command is issued on an autocall line, the modem-out pattern shows Call Request (CRQ) and Digit Present (DPR) both off.

Reset-N Command (X'0C')

Reset-N means "Reset and No-Op".The command is used to set the line to the enabled No-Op state. If a wrap is in progress, it is terminated. The line must have been previously initialized by means of a 'Set Mode' command, and there must be no outstanding command on either interface. If this is not the case, a Level 1 interrupt occurs with an error status type 3.

This command may be issued only to an even (transmit) interface. If it is issued to an odd (receive) interface, it is rejected. This command is also rejected if a Set Mode command has not previously been issued.

Parameter Zone

Word 1	TCC	-	-	-
Word 2	-	-	-	-
Word 3	-	-	-	-
Word 4	-	-	-	-

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	Modem-In	Modem-Out
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information which describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'0C'.

Secondary Status (SES) Field: Contains the secondary status. This field is always X'00' for this command in normal mode; in character mode, it is not used.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Modem-In and Modem-Out Fields: These fields are described in detail at the end of this chapter under the heading "Modem Control Fields".

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

1. The Reset-N command should be used instead of the Enable command on lines equipped with Option 1 modems.

2. The Reset-N command does not have an internal wait, and does not wait for DSR to rise.

Raise Data Terminal Ready Command (X'08')

The Raise Data Terminal Ready command is used to raise the DTR signal. "Data Set Ready" is not monitored. It must be issued to the data line before issuing the "Dial" command to the associated autocall interface.

This command may be issued only to an even interface. If it is issued to an odd interface, it is rejected.

Parameter Zone

Word 1	TCC	-	-	-
Word 2	-	-	-	-
Word 3	-	-	-	-
Word 4	-	-	-	-

Status Zone (Normal Mode)

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	Modem-In	Modem-Out
Word 3	-	-	-	-

Status Zone (Character Mode)

Word 1	SCF	CCMD	-	LCS
Word 2	LCD/PCF	SDF	Modem-In	Modem-Out
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information which describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'08'.

Secondary Status (SES) Field: Contains the secondary status. This field is always X'00' for the Set Mode command in normal mode; in character mode, it is not used.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred. Refer to the end of this chapter for full details.

Modem-In and Modem-Out Fields: These fields are described in detail at the end of this chapter under the heading "Modem Control Fields".

Other Fields: The remaining fields of the status area are not used for the Set Mode command in normal mode. In character mode, two additional fields are

used: LCD/PCF and SDF. The meaning of these fields is described in "Character Mode Commands".

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Halt Command (X'F0)

The Halt command requires no parameters, so that no cycle stealing occurs from the PSA area in CCU storage. It is used to terminate an outstanding command:

- If no command is active when the Halt command is issued, it is ignored; no status is returned, and no CCU level 2 interrupt occurs.
- If a command is active when the Halt command is issued, it is terminated, and the Halt bit (bit 0) and Service Request bit (bit 1) are set in the SCF. The status area is transferred to CCU storage, and a CCU level 2 interrupt request is raised. The status area shows any conditions (such as Start Bit Detected, Modem Check, etc.) that occurred before the current command was terminated by the Halt command.

  Note: If the halted command was Write ICW with PCF = X'7' and a character has been received, the new character is not stored in the PDF. The received character is however retained, and is set in the PDF for the **next** Write ICW command with PCF = X'7'.
- All outstanding commands can be halted; however, Trace and Stop Trace should not be halted, because the result is unpredictable.
- If a Halt command is issued to a previous Halt command without an intervening CCU level 2 interrupt, the scanner ignores the second Halt.
- The scanner takes no action on the line except for NCP BSC commands and the EP BSC Receive command.

Parameter Zone

The parameter zone has no meaning for the Halt command.

Status Zone (Normal Mode)

Word 1	SCF	Halted cmd.	SES	LCS
Word 2	-	-	-	-
Word 3	-	-	-	-

Status Zone (Character Mode)

Word 1	SCF	Halted cmd.	SES	LCS
Word 2	LCD/PCF	SDF	Modem-In	Modem-Out
Word 3	-	-	-	-

Important Note: The information contained in the status zone refers to the command that was halted, and not to the Halt command itself.

Status Control Field (SCF): This byte contains information which describes the progress of the command. See "Ending Status" below. Refer to the end of this chapter for full details of this field.

Halted Command Field: Contains the code for the command that was halted.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: Refer to the end of this chapter for full details of this field.

Modem-In and Modem-Out Fields: These fields are described in detail at the end of this chapter under the heading "Modem Control Fields".

Other Fields: The remaining fields of the status area are not used in the normal mode. In character mode, two additional fields are used: LCD/PCF and SDF. The meaning of these fields is described the "Character Mode Commands".

Ending Status

The ending status depends only on the command being halted.

Special Considerations

1. If the scanner has already transferred its ending status to the scanner, the Halt command has no effect.

2. If the line that has been halted is defined as duplex, two commands may be outstanding, one for the transmit interface, the other for the receive interface. The Halt command affects only one of the interfaces, as defined in the Halt command. If the commands on both interfaces must be terminated, **two** Halt commands are required.

3. The scanner takes no action on the line, except for NCP BSC commands:

   NCP BSC Control Command: the transmission of the control characters is completed, and the line is turned around to the receive state. The received data is flushed to an end-of-block or timeout condition, the line is left in the receive state, and the command is terminated with the Halt status.

   NCP BSC Transmit and Transmit Continue Commands: the scanner transmits the control characters DLE-ENQ, and the line is turned around to the receive state; the command is terminated with the Halt status.

   NCP BSC Receive and Receive Continue Commands: the received data is flushed to an end of block or timeout condition, the line is left in the receive state, and the command is terminated with the Halt status.

Effects of the Halt Command

Set Mode Command

The Halt command is ignored and Set Mode command processing is completed. The "Halt" bit is not set in the SCF.

Change Command

The Halt command is ignored and Change command processing is completed. The "Halt" bit is not set in the SCF.

Enable Command

Enable command processing is stopped. All modem-out leads are dropped, the line is set to the No-Op state, and the command is ended with a halt status.

If the line was in the receive mode, all current data and all queued data is purged; all related statuses except "Modem Check" in the SCF, and "Internal Box Error" in the LCS are also purged.

Note: Issuing a Halt command while an Enable command is pending may leave the modem interface in an unstable condition. For this reason, the next command issued to the scanner should be a Disable command to stabilize the modem interface.

Disable Command

The Halt command is ignored and Disable command processing is completed. The "Halt" bit is not set in the SCF.

Note: The disable timer is started, even if the line was already in the disable state.

Dial Command

Dial command processing is stopped. All leads to the autocall unit are dropped, and the command is ended with a halt status.

Note: The Autocall unit is reset at the next Dial command.

Monitor Incoming Call Command

Monitor Incoming Call command processing is stopped. All modem-out leads are dropped, the line is set to the No-Op state, and the command is ended with a halt status.

If the line was in the receive mode, all current data and all queued data is purged; all related statuses except "Modem Check" in the SCF and "Internal Box Error" in the LCS are also purged.

Note: Issuing a Halt command while a Monitor Incoming Call command is pending may leave the modem interface in an unstable condition. For this reason, the next command issued to the scanner should be a Disable command to stabilize the modem interface.

Flush Command

The Halt command is ignored and Flush command processing is completed. The "Halt" bit is not set in the SCF.

Reset-D Command

The Halt command is ignored and Reset-D command processing is completed. The "Halt" bit is not set in the SCF.

Reset-N Command

The Halt command is ignored and Reset-N command processing is completed. The "Halt" bit is not set in the SCF.

Raise DTR Command

Raise DTR command processing is stopped. All modem-out leads are dropped, the line is set to the No-Op state, and the command is ended with a halt status.

Note: Issuing a Halt command while a Raise DTR command is pending may leave the modem interface in an unstable condition. For this reason, the next command issued to the scanner should be a Disable command to stabilize the modem interface.

Trace command

The Halt command should not be used to terminate a Trace command because the result is unpredictable.

Stop Trace command

The Halt command should not be used to terminate a Stop Trace command because the result is unpredictable.

386X/58XX Test Command

386X/58XX Test command processing is stopped. The TC lead is dropped, then the scanner waits for the TI lead to fall, or for the TI timeout. The line is set to the No-Op state, and the command is ended with a halt status.

Wrap (Data) Command

Wrap command processing is stopped. All modem-out leads are dropped, the line is set to the No-Op state, and the command is ended with a halt status.

If the line was in the receive mode, all current data and all queued data is purged; all related statuses except "Modem Check" in the SCF and "Internal Box Error" in the LCS are also purged.

Note: Issuing a Halt command while a Wrap command is pending may leave the modem interface in an unstable condition. For this reason, the next command should be a Disable command to stabilize the modem interface.

Wrap (Control Lead) Command

Wrap command processing is stopped. All modem-out leads are dropped, the line is set to the No-Op state, and the command is ended with a halt status. The disable timer does not run.

Note: Issuing a Halt command while a Wrap command is pending may leave the modem interface in an unstable condition. For this reason, the next command issued to the scanner should be a Disable command to stabilize the modem interface.

All SDLC Transmit Type Commands (Except X.21)

Transmission is immediately stopped, but RTS does not drop. The interface is set to the No-Op state, and the command ends with the current status ORed with the halt bit.

All SDLC Receive Type Commands (Except X.21)

All current data and all queued data is purged; all related statuses except "Modem Check" in the SCF, and "Internal Box Error" in the LCS are also purged. The interface is set to the No-Op state, and the command ends with the current status ORed with the halt bit.

All SDLC Transmit Type Commands (X.21 Only)

Transmission is stopped immediately and the interface is set to the No-Op state. The command ends with a halt status.

All SDLC Receive Type Commands (X.21 Only)

All data is purged, and the interface is set to the No-Op state. The command ends with a halt status.

X.21 Call Request Command

Call Request command processing is stopped; the line is cleared and set to the X.21 No-Op state. The command ends with a halt status.

X.21 Monitor Incoming Call Command

Monitor Incoming Call command processing is stopped; the line is cleared and set to the X.21 No-Op state. The command ends with a halt status.

X.21 DTE Clear Request Command

The Halt command is ignored, and DTE Clear Request command processing is completed.

NCP BSC Transmit and Transmit Continue Commands

1. **If transmitting text**

   A DLE character followed by ENQ is sent; the line then turns around. When an answer is received, or a timeout occurs, the interface is set to the No-Op state. The command ends with the current status ORed with the halt bit.

2. **If not transmitting text**

   The Halt command is ignored and Transmit or Transmit Continue command processing is completed. The halt bit is not set.

3. **Data already in scanner buffer, but transmission not yet started**

   Transmit or Transmit Continue command processing is stopped. The interface is set to the No-Op state, and the command ends with the current status ORed with the halt bit.

All NCP BSC Commands except Transmit and Transmit Continue

The Halt command is ignored and NCP BSC command processing is completed.

EP BSC Transmit Command

A DLE character followed by ENQ is sent. If the line is half duplex, RTS is dropped. The interface is then set to the No-Op state, and the command ends with the current status ORed with the halt bit.

Note: Issuing a Halt command while an EP BSC Transmit or EP BSC Transmit Continue command is pending may leave the modem interface in an unstable condition. For this reason, the next command issued to the scanner should be a Disable command to stabilize the modem interface.

EP BSC Receive Command

All data is purged. The interface is set to the No-Op state, and the command ends with the current status ORed with the halt bit.

Write ICW Command

Write ICW command processing is stopped; if the command is Transmit, RTS is not dropped. The interface is then set to the No-Op state, and the command ends with the current status ORed with the halt bit.

Note: Issuing a Halt command while a Write ICW command is pending may leave the modem interface in an unstable condition. For this reason, the next command issued to the scanner should be a Disable command to stabilize the modem interface.

Halt Command

The second Halt command is ignored.

Note: If any other command is received while a Halt command is being processed, the Command Reject bit is set, and a Level 1 interrupt occurs.

Halt Immediate Command

The Halt command is ignored.

Command Queued because of a Halt Immediate Command

If a Halt Immediate command was issued, followed by a new command, the new command is normally executed by the scanner. If the Halt Immediate processing was not completed, the new command is queued for later execution. If a **Halt** command is now received for the new command while it is still queued, the new command is not executed, and the scanner ends the new command with a halt status.

Halt Immediate Command (X'F1)

The Halt Immediate command may be issued to the scanner at any moment. All commands may be halted. However, "Trace" and "Stop Trace" should not be halted, because the result is unpredictable.

- The Halt Immediate command requires no parameters, so that no cycle stealing occurs from the PSA area in CCU storage. It does not create an outstanding command condition in the scanner.

- If no command is outstanding to both the CCU and the scanner when the Halt Immediate command is issued, it is ignored; no status is returned and no CCU level 2 interrupt occurs.

- If a command is outstanding to both the CCU and the scanner when the Halt Immediate command is issued, the following sequence of events occurs:

1. The scanner ignores the contents of the parameter area which is considered to be available to the CCU control program to set up the parameters for the next command after the Halt Immediate.

2. Any pending status is discarded. No cycle stealing to CCU storage occurs, and no level 2 interrupt is set. At this point there is no command outstanding.

Note: In some cases, an outstanding command may already have been terminated by the scanner, and a level 2 interrupt request posted to the CCU at the moment that the Halt Immediate command was issued, but not yet treated by the control program. The line ID is still waiting in the scanner.

1. The line identification already queued in the scanner is invalidated by changing its value to X'1080'.

2. Scanner processing continues as though the Halt Immediate command has not been received. However, the status area in the scanner is locked, and no further level 2 interrupts may be requested by the scanner until the CCU control program issues another command to the scanner.

Parameter and Status Zones

The parameter and status zones have no meaning for the Halt Immediate command.

Ending Status

The ending status has no meaning for the Halt Immediate command.

Special Considerations

1. If the line that has been halted is defined as duplex, two commands may be outstanding, one for the transmit interface, the other for the receive interface. The Halt Immediate command affects only one of the interfaces, as defined in the Halt Immediate command. If the commands on both interfaces must be terminated, **two** Halt Immediate commands are required.

2. If the command that was halted by the Halt Immediate command was about to raise a Modem Check or an Internal Box Error, it is only raised on the **next** command.

   Note, however, that a Modem Check is **not** raised under these conditions if the following command is Set Mode, Enable, Disable, or X.21 Clear.

3. If the command that was halted by the Halt Immediate command was "Write ICW", and the halt was received after the scanner had stored the status, but before the level 2 interrupt request was raised to the CCU, the level 2 interrupt request is suppressed. The control program must take into account the possibility that the status area has been changed.

Effect of Halt Immediate Command

Set Mode Command

Set Mode command processing is completed, and the line remains in the No-Op state. Any pending status is ignored.

Note: The No-Op state does not apply to a line connected to an ACU.

Change Command

Change command processing is completed, and the line set to the No-Op state. Any pending status is ignored.

Enable Command

Enable command processing is stopped.

If the line was in receive mode, all current data and all queued data is purged; all related statuses except "Modem Check" in the SCF, and "Internal Box Error" in the LCS are also purged. The line is set to the No-Op state; any pending status is ignored. The modem-out leads are unchanged.

Disable Command

Disable command processing is stopped. The line is set to the No-Op state; any pending status is ignored. The modem-out leads are unchanged.

Dial Command

Dial command processing is stopped and all leads to the Autocall unit are dropped. Any pending status is ignored.

Note: The Autocall unit is reset at the next Dial command.

Monitor Incoming Call Command

Monitor Incoming Call command processing is stopped. If the line was in the receive mode, all current data and all queued data is purged; all related statuses except "Modem Check" in the SCF, and "Internal Box Error" in the LCS are also purged. The line is set to the No-Op state; any pending status is ignored. The modem-out leads are unchanged.

Flush Command

Flush command processing is stopped and the line is set to the No-Op state. Any pending status is ignored.

Reset-D Command

Reset-D command processing is completed and the line is set to the No-Op state. Any pending status is ignored.

Reset-N Command

Reset-N command processing is completed and the line is set to the No-Op state. Any pending status is ignored.

Raise DTR Command

Raise DTR command processing is completed, and the line is set to the No-Op state. Any pending status is ignored.

Trace command

The Halt Immediate command should not be used to terminate a Trace command because the result is unpredictable.

Stop Trace command

The Halt Immediate command should not be used to terminate a Stop Trace command because the result is unpredictable.

386X/58XX Test Command

386X/58XX Test command processing is stopped. The TC lead is dropped, but the scanner does not wait for the TI lead to fall. The line is set to the No-Op state.

Note: Because the fall of TI is not monitored, the command following the Halt Immediate command may find the TI lead still on, and will be rejected with "command reject" as follows:

1. Next command is another 386X/58XX Test command: the LCS indicates "TI already on".
2. Next command is any other transmit command: the LCS indicates "CTS failed to rise".

Wrap (Data) Command

Wrap command processing is stopped. If the line was in receive mode, all current data and all queued data is purged; all related statuses except "Modem Check" in the SCF, and "Internal Box Error" in the LCS are also purged. The line is set to the No-Op state; any pending status is ignored.

Wrap (Control Lead) Command

Wrap command processing is stopped. All modem-out leads are dropped, the line is set to the No-Op state, and any pending status is ignored.

All SDLC Transmit type Commands (except X.21)

Transmission is immediately stopped, but RTS does not drop. The interface is set to the No-Op state, and any pending status is ignored.

All SDLC Receive type Commands (except X.21)

If the line was in receive mode, all current data and all queued data is purged; all related statuses except "Modem Check" in the SCF, and "Internal Box Error" in the LCS are also purged. The interface is set to the No-Op state, and any pending status is ignored.

All SDLC Transmit Type Commands (X.21 Only)

Transmission is stopped immediately, the line is cleared, and the interface is set to the No-Op state. Any pending status is ignored.

All SDLC Receive Type Commands (X.21 Only)

All data is purged, and the interface is set to the No-Op state. Any pending status is ignored.

X.21 Call Request Command

Call Request command processing is stopped; the line is cleared and set to the X.21 No-Op state. Any pending status is ignored.

X.21 Monitor Incoming Call Command

Monitor Incoming Call command processing is stopped; the line is cleared and set to the X.21 No-Op state. Any pending status is ignored.

X.21 DTE Clear Request Command

DTE Clear Request command processing is completed; the line is cleared and set to the X.21 No-Op state. Any pending status is ignored.

All NCP BSC Transmit type Commands

A DLE character followed by ENQ is sent, but RTS does not drop. The interface is set to the No-Op state, and any pending status is ignored.

All NCP BSC Receive type Commands

All data is purged. The interface is set to the No-Op state, and any pending status is ignored.

EP BSC Transmit Command

A DLE character followed by ENQ is sent. If the line is half duplex, RTS is dropped. The interface is set to the No-Op state, and any pending status is ignored.

EP BSC Receive Command

All data is purged. The interface is set to the No-Op state, and any pending status is ignored.

Write ICW Command

Write ICW command processing is stopped; if the command is Transmit, RTS is not dropped. The interface is set to the No-Op state, and any pending status is ignored.

Halt Command

Halt command processing is stopped. The interface is set to the No-Op state, and any pending status is ignored.

Halt Immediate Command

The Halt Immediate command is ignored.

Note: If any other command is received while a Halt Immediate command is being processed, the command is accepted and executed normally by the scanner.

Command Queued because of a Halt Immediate Command

If a Halt Immediate command was issued, followed by a new command, the new command is normally executed by the scanner. If the Halt Immediate processing was not completed, the new command is queued for later execution. If a Halt Immediate command is now received for the new command while it is still queued, the new command is ignored by the scanner. However, the original Halt Immediate command processing is completed normally.

NCP SDLC COMMANDS

The NCP SDLC commands are used by the NCP and similar programs for data transfer and control on SDLC lines.

SDLC Transmit Control Command (X'10')

The SDLC Transmit Control command is used to transmit control information only; as no data is to be transmitted, a transmit buffer is not required, but a receive buffer is provided for the response. The command is used as follows:

1. To send supervisor frames (RR, RNR, REJ) on the even interface of a duplex or half duplex link and receive a response. The transmission may be from a primary station to a secondary station, or vice-versa.

2. To send non-sequenced frames without information fields from a primary to a secondary station.

3. To send non-sequenced responses without information fields from a secondary to a primary station.

The command may be issued only to an even (transmit) interface. If it is issued to an odd (receive) interface, it is rejected.

Parameter Zone

Word 1	TCC	Modifiers	Offset	-
Word 2	Byte Count	First Receive Buffer Pointer		
Word 3	XA1	XA2/XC1	XC1	XC2
Word 4	-	-	-	-

Modifier Byte: This byte contains command modifier bits that have the following meaning:

Bit 0 - NCP Type Buffer: This bit, if off, indicates that the data to be transferred to the control program is in an NCP type buffer whose address is contained in the "First Receive Buffer Pointer". If this bit is on, the data is not in an NCP type buffer, but in a data area whose address is contained in the "First Transmit Buffer Pointer" (in this case, the buffer prefix and buffer offset are not used).

Bit 2 - Two-Byte Address: This bit, if on, indicates that the SDLC address field (XA field) in the parameter area is two bytes long (XA1 and XA2), and that the expected address field in the receive frame is also two bytes long. If the bit is off, it indicates that the SDLC address field is one byte long (XA1 only).

Bit 3 - Two-Byte Control: This bit, if on, indicates that the SDLC control field (XC field) in the parameter area is two bytes long (XC1 and XC2), and that the expected control field in the receive frame is also two bytes long. If the bit is off, it indicates that the SDLC control field is one byte long (XC1 only).

Bit 4 - Compare Address: This bit, if off, indicates that the address field (1 or 2 bytes) in the received SDLC frame must be compared to the address in

the parameter area, if primary side, or to the address specified by the Set Mode command (data area, bytes 10 and 11), if secondary side. If a mismatch occurs, the received frame is rejected (an **All Parties** address is always accepted, however). If the bit is on, no compare occurs.

Bit 5 - Answer Requested: This bit, when on, indicates that once the frame has been transmitted, a response is expected within a given period of time.

Bit 6 - Turn Line Around/Drop RTS: For half duplex lines, this bit, when on, indicates that the line must be put into receive mode as soon as the frame has been transmitted. For duplex lines, RTS must drop as soon as the frame has been transmitted.

Bit 7 - Receive Area Assigned: This bit is valid for half duplex lines only. When on, it indicates that a chain of buffers is available to assemble the received frame.

Offset: This is the number of bytes between the buffer address contained in word 2 and the start of the data.

Byte Count: This is the number of bytes of received data to be transferred.

First Receive Buffer Pointer: This three-byte field contains the address of the buffer where the response to the transmitted control byte(s) must be stored.

SDLC Transmit Address 1 (XA1): This byte contains the SDLC station address to be used in the frame being transmitted (first byte of a two-byte station address).

SDLC Transmit Address 2 (XA2): This byte contains the second byte of the SDLC station address when a two-byte station address is used.

SDLC Transmit Control 1 (XC1): This byte contains the SDLC control byte to be used in the frame being transmitted (first byte of a two-byte control field).

SDLC Transmit Control 2 (XC2): This byte contains the second byte of the SDLC control field when a two-byte control field is used.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	Res. Count	Last Receive Buffer Used		
Word 3	RA1	RA2/RC1	RC1	RC2

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'10'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Residual Byte Count: This byte indicates the number of unused bytes remaining in the last buffer used.

Last Receive Buffer Used: This three-byte field indicates the last buffer that was used to hold the data received in response to the transmitted control byte(s).

SDLC Receive Address 1 (RA1): This byte contains the SDLC station address received in the current frame (first byte of a two-byte station address).

SDLC Receive Address 2 (RA2): This byte contains the second byte of the SDLC station address when a two-byte station address is used.

SDLC Receive Control 1 (RC1): This byte contains the SDLC control byte received in the current frame (first byte of a two-byte control field).

SDLC Receive Control 2 (RC2): This byte contains the second byte of the SDLC control field when a two-byte control field is used.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

1. On a duplex link, the Transmit Control command on the transmit side should be preceded by a Receive command on the receive side. The PSA of this command contains the address of the first buffer of a receive buffer chain. Alternatively, a Receive Monitor command may be used.

2. The way in which the command is terminated after frame transmission depends on the way in which the line is defined (duplex or half duplex), and on the command modifiers:

   - **Half Duplex and not "Turn Line Around":** The command is terminated with "transmit completed" status. The line is left in transmit mode sending continuous flags (if specified by the Set Mode command), or continuous "mark".

   - **Half Duplex and "Turn Line Around":** Ready To Send (RTS) is dropped (if "turn with RTS on" was not specified by the Set Mode command), and monitor for clear to send (CTS). The line is placed in the receive mode. Further actions depend on the "answer requested" (AR) and "receive area assigned" (RAA) bits, as shown in the following table:

AR	RAA	Action
0	-	End the command ("transmit complete")
1	0	Start the Reply timeout. When the first data comes in, end the command ("buffer request", or "response received with no data")
1	1	Start the Reply Timeout. Put the data into the buffer(s). End the command ("buffer request", or "response received with data")

   - **Duplex:** the transmit interface transmits continuous flags (if specified by the Set Mode command), or continuous "mark". The command is ended ("transmit completed"), and the Reply timeout is started on

the receive side if the "answer requested" modifier is on. If the "Drop RTS" modifier (bit 6) is specified, the transmit interface stops transmission, turning off RTS.

SDLC Transmit Data Command (X'11')

The SDLC Transmit Data command is used to transmit data from a primary station to a secondary station, and vice-versa. The command is used as follows:

1. To send non-sequenced frames with information fields from a primary station to a secondary station.

2. To send information frames from a primary station to a secondary station on the even station of a duplex link, with or without the polling bit. Alternatively, a half duplex link may be used; in this case, the response is received on the same interface.

3. To send information frames without the final bit, from a secondary to a primary station.

4. To send non-sequenced responses with information fields from a secondary to a primary station.

The command may be issued only to an even (transmit) interface. If it is issued to an odd (receive) interface, the command is rejected.

Parameter Zone

Word 1	TCC	Modifiers	Offset (T)	Offset (R)
Word 2	Byte C. (T)	First Transmit Buffer Pointer		
Word 3	XA1	XA2/XC1	XC1	XC2
Word 4	Byte C. (R)	First Receive Buffer Pointer		

Modifier Byte: This byte contains command modifier bits which have the following meaning:

Bit 0 - NCP Type Buffer: This bit, if off, indicates that the data to be transferred from the control program is in an NCP type buffer whose address is contained in the "First Transmit Buffer Pointer". If this bit is on, the data is not in an NCP type buffer, but in a data area whose address is contained in the "First Transmit Buffer Pointer" (in this case, the buffer prefix and buffer offset are not used).

Bit 1 - NCP Type Buffer: This bit, if off, indicates that the data transferred to the control program is in an NCP type buffer whose address is contained in the "First Receive Buffer Pointer". If this bit is on, the data is not in an NCP type buffer, but in a data area whose address is contained in the "First Receive Buffer Pointer" (in this case, the buffer prefix and buffer offset are not used).

Bit 2 - Two-Byte Address: This bit, if on, indicates that the SDLC address field (XA field) in the parameter area is two bytes long (XA1 and XA2), and that the expected address field in the receive frame is also two bytes long. If the bit is off, it indicates that the SDLC address field is one byte long (XA1 only).

Bit 3 - Two-Byte Control: This bit, if on, indicates that the SDLC control field (XC field) in the parameter area is two bytes long (XC1 and XC2), and that the expected control field in the receive frame is also two bytes long. If the bit is off, it indicates that the SDLC control field is one byte long (XC1 only).

Bit 4 - Compare Address: This bit, if on, indicates that the address field (1 or 2 bytes) in the received SDLC frame must be compared to the address in the parameter area, if primary side, or to the address specified by the Set Mode command (data area, bytes 10 and 11), if secondary side. If a mismatch occurs, the received frame is rejected (an **All Parties** address is always accepted, however).

Bit 5 - Answer Requested: This bit, when on, indicates that once the frame has been transmitted, a response is expected within a given period of time.

Bit 6 - Turn Line Around/Drop RTS: For half duplex lines, this bit, when on, indicates that the line must be put into receive mode as soon as the frame has been transmitted. For duplex lines, RTS must drop as soon as the frame has been transmitted.

Bit 7 - Receive Area Assigned: This bit is valid for half duplex lines only. When on, it indicates that a chain of buffers is available to assemble the received frame.

Offset (T): This is the number of bytes between the transmit buffer address contained in word 2 and the start of the data.

Offset (R): This is the number of bytes between the receive buffer address contained in word 4 and the start of the data.

Byte Count (T): This is the number of bytes of transmit data to be transferred.

First Transmit Buffer Pointer: This three-byte field contains the address of the buffer where the data to be transmitted is stored.

Note: The least significant bit of this three-byte field (byte 7, bit 7 of parameter area) is the "SDLC Data Chain" bit. See below under "Special Considerations".

SDLC Transmit Address 1 (XA1): This byte contains the SDLC station address to be used in the frame being transmitted (first byte of a two-byte station address).

SDLC Transmit Address 2 (XA2): This byte contains the second byte of the SDLC station address when a two-byte station address is used.

SDLC Transmit Control 1 (XC1): This byte contains the SDLC control byte to be used in the frame being transmitted (first byte of a two-byte control field).

SDLC Transmit Control 2 (XC2): This byte contains the second byte of the SDLC control field when a two-byte control field is used.

First Receive Buffer Pointer: This three-byte field contains the address of the buffer where the response is to be stored.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	Res. Count	Last Receive Buffer Used		
Word 3	RA1	RA2/RC1	RC1	RC2

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'11'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Residual Byte Count: This byte indicates the number of unused bytes remaining in the last buffer used.

Last Receive Buffer Used: This three-byte field indicates the last buffer that was used to hold the data received in response to the transmitted control byte(s).

SDLC Receive Address 1 (RA1): This byte contains the SDLC station address received in the current frame (first byte of a two-byte station address).

SDLC Receive Address 2 (RA2): This byte contains the second byte of the SDLC station address when a two-byte station address is used.

SDLC Receive Control 1 (RC1): This byte contains the SDLC control byte received in the current frame (first byte of a two-byte control field).

SDLC Receive Control 2 (RC2): This byte contains the second byte of the SDLC control field when a two-byte control field is used.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

1. On a duplex link, the Transmit Data command with answer requested in the modifiers on the transmit side should be preceded by a Receive command on the receive side. The PSA of this command contains the address of the first buffer of a receive buffer chain. Alternatively, a Receive Monitor command may be used.

2. How the command is terminated after frame transmission depends on how the line is defined (duplex or half duplex), and on the command modifiers:

   - **Half Duplex and not "Turn Line Around":** The command is terminated with "transmit completed" status. The line is left in transmit mode sending continuous flags (if specified by the Set Mode command), or continuous "mark".

   - **Half Duplex and "Turn Line Around":** Ready To Send (RTS) is dropped (if "turn with RTS on" was not specified by the Set Mode command), and monitor for clear to send (CTS). The line is placed in the receive mode. Further actions depend on the "answer requested" (AR) and "receive area assigned" (RAA) bits, as shown in the following table:

AR	RAA	Action
0	-	End the command ("transmit complete")
1	0	Start the Reply timeout. When the first data comes in, end the command ("buffer request", or "response received with no data")
1	1	Start the Reply Timeout. Put the data into the buffer(s). End the command ("buffer request", or "response received with data")

   - **Duplex:** The transmit interface transmits continuous flags (if specified by the Set Mode command), or continuous "mark". The command is ended ("transmit completed"), and the Reply timeout is started on the receive side if the "answer requested" modifier is on. If the "Drop RTS" modifier (bit 6) is specified, the transmit interface stops transmission, turning off RTS.

3. The rightmost bit of the "First Transmit Buffer Pointer" (byte 7, bit 7 of the parameter area) is the SDLC data chaining bit. When all the data has been transmitted, this bit is tested. If the rightmost bit is off, it means that the complete SDLC frame has been transmitted; the BCC and final flag are then sent, and the line is turned around (if specified in the modifiers). The command is terminated.

   If the bit is on, the SDLC frame has not been completely transmitted, and more data is waiting in a new buffer chain or in a data area.

SDLC Transmit Continue Command (X'1D')

The SDLC Transmit Continue command is used when a previous SDLC Transmit Data command (or a previous Transmit Continue command) was ended with a "Buffer Request (Transmit)" condition in the ending status. It provides a new chain of buffers or a new data area.

Parameter Zone

Word 1	TCC	Modifiers	Offset (T)	Offset (R)
Word 2	Byte C. (T)	First Transmit Buffer Pointer		
Word 3	XA1	XA2/XC1	XC1	XC2
Word 4	Byte C. (R)	First Receive Buffer Pointer		

Modifier Byte: This byte contains command modifier bits which have the following meaning:

Bit 0 - NCP Type Buffer: This bit, if off, indicates that the data to be transferred from the control program is in an NCP type buffer whose address is contained in the "First Transmit Buffer Pointer". If this bit is on, the data is not in an NCP type buffer, but in a data area whose address is contained in the "First Transmit Buffer Pointer" (in this case, the buffer prefix and buffer offset are not used).

Bit 1 - NCP Type Buffer: This bit, if off, indicates that the data transferred to the control program is in an NCP type buffer whose address is contained in the "First Receive Buffer Pointer". If this bit is on, the data is not in an NCP type buffer, but in a data area whose address is contained in the "First Receive Buffer Pointer" (in this case, the buffer prefix and buffer offset are not used).

Bit 2 - Two-Byte Address: This bit, if on, indicates that the SDLC address field (XA field) in the parameter area is two bytes long (XA1 and XA2), and that the expected address field in the receive frame is also two bytes long. If the bit is off, it indicates that the SDLC address field is one byte long (XA1 only).

Bit 3 - Two-Byte Control: This bit, if on, indicates that the SDLC control field (XC field) in the parameter area is two bytes long (XC1 and XC2), and that the expected control field in the receive frame is also two bytes long. If the bit is off, it indicates that the SDLC control field is one byte long (XC1 only).

Bit 4 - Compare Address: This bit, if on, indicates that the address field (1 or 2 bytes) in the received SDLC frame must be compared to the address in the parameter area, if primary side, or to the address specified by the Set Mode command (data area, bytes 10 and 11), if secondary side. If a mismatch occurs, the received frame is rejected (an **All Parties** address is always accepted, however).

Bit 5 - Answer Requested: This bit, when on, indicates that once the frame has been transmitted, a response is expected within a given period of time.

Bit 6 - Turn Line Around/Drop RTS: For half duplex lines, this bit, when on, indicates that the line must be put into receive mode as soon as the frame has been transmitted. For duplex lines, RTS must drop as soon as the frame has been transmitted.

Bit 7 - Receive Area Assigned: This bit is valid for half duplex lines only. When on, it indicates that a chain of buffers is available to assemble the received frame.

Offset (T): This is the number of bytes between the transmit buffer address contained in word 2 and the start of the data.

Offset (R): This is the number of bytes between the receive buffer address contained in word 4 and the start of the data.

Byte Count (T): This is the number of bytes of transmit data to be transferred.

First Transmit Buffer Pointer: This three-byte field contains the address of the buffer where the data to be transmitted is stored.

Note: The least significant bit of this three byte field (byte 7, bit 7 of parameter area) is the "SDLC Data Chain" bit. See below under "Special Considerations".

SDLC Transmit Address 1 (XA1): This byte contains the SDLC station address to be used in the frame being transmitted (first byte of a two-byte station address).

SDLC Transmit Address 2 (XA2): This byte contains the second byte of the SDLC station address when a two-byte station address is used.

SDLC Transmit Control 1 (XC1): This byte contains the SDLC control byte to be used in the frame being transmitted (first byte of a two-byte control field).

SDLC Transmit Control 2 (XC2): This byte contains the second byte of the SDLC control field when a two-byte control field is used.

First Receive Buffer Pointer: This three-byte field contains the address of the buffer where the response is to be stored.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	Res. Count	Last Receive Buffer Used		
Word 3	RA1	RA2/RC1	RC1	RC2

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'1D'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Residual Byte Count: This byte indicates the number of unused bytes remaining in the last buffer used.

Last Receive Buffer Used: This three-byte field indicates the last buffer that was used to hold the data received in response to the transmitted control byte(s).

SDLC Receive Address 1 (RA1): This byte contains the SDLC station address received in the current frame (first byte of a two-byte station address).

SDLC Receive Address 2 (RA2): This byte contains the second byte of the SDLC station address when a two-byte station address is used.

SDLC Receive Control 1 (RC1): This byte contains the SDLC control byte received in the current frame (first byte of a two-byte control field).

SDLC Receive Control 2 (RC2): This byte contains the second byte of the SDLC control field when a two-byte control field is used.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

Same as for the SDLC Transmit Data command.

SDLC Receive Monitor Command (X'12')

The SDLC Receive Monitor command is used to monitor a line for incoming data.

Parameter Zone

Word 1	TCC	Modifiers	-	-
Word 2	-	-	-	-
Word 3	RA1	RA2	-	-
Word 4	-	-	-	-

Modifier Byte: This byte contains command modifier bits which have the following meaning:

Bit 2 - Two-Byte Address: This bit, if on, indicates that the SDLC address field (XA field) in the parameter area is two bytes long (XA1 and XA2), and that the expected address field in the receive frame is also two bytes long. If the bit is off, it indicates that the SDLC address field is one byte long (XA1 only).

Bit 3 - Two-Byte Control: This bit, if on, indicates that the SDLC control field (XC field) in the parameter area is two bytes long (XC1 and XC2), and that the expected control field in the receive frame is also two bytes long. If the bit is off, it indicates that the SDLC control field is one byte long (XC1 only).

Bit 4 - Compare Address: This bit, if on, indicates that the address field (1 or 2 bytes) in the received SDLC frame must be compared to the address in the parameter area, if primary side, or to the address specified by the Set Mode command (data area, bytes 10 and 11), if secondary side. If a mismatch occurs, the received frame is rejected (an **All Parties** address is always accepted, however).

SDLC Receive Address 1 (RA1): This byte contains the SDLC station address to be compared against the address in the frame being received (first byte of a two-byte station address).

SDLC Receive Address 2 (RA2): This byte contains the second byte of the SDLC station address when a two-byte station address is used.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	-	-
Word 3	RA1	RA2/RC1	RC1	RC2

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'12'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

SDLC Receive Address 1 (RA1): This byte contains the SDLC station address received in the current frame (first byte of a two-byte station address).

SDLC Receive Address 2 (RA2): This byte contains the second byte of the SDLC station address when a two-byte station address is used.

SDLC Receive Control 1 (RC1): This byte contains the SDLC control byte received in the current frame (first byte of a two-byte control field).

SDLC Receive Control 2 (RC2): This byte contains the second byte of the SDLC control field when a two-byte control field is used.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

1. The XA1 and XA2 fields in the parameter area are loaded by the control program only if the controller is the primary station, and if bit 4 of the modifier field specifies "compare address". The address in the receive frame is then compared with the contents of XA1/XA2. If the controller is a secondary station, and "compare address" is specified, the received address is compared with the address specified by the Set Mode command. In this case, the XA1 and XA2 fields are not specified in the command.

2. Start of message: depending on the length of the address and control fields (as specified in the modifiers), the command is ended when the 6th, 7th, or 8th character of the frame has been assembled. If this character is a flag, the ending status indicates "frame received - no data"; if the character is not a flag, the ending status indicates "buffer request".

3. On a duplex link, the reply timeout is started on the receive side when a Transmit Control or Transmit Data command with "answer requested" in the modifier byte has been issued on the transmit side. This timeout is the maximum time allowed before a response must arrive.

   Once reception has started, the receive text timeout is started. This timeout is reset to zero at the end of a frame, or at buffer request. In this latter case, the receive text timeout is restarted by a Receive Continue command.

4. In duplex receive mode, as soon as the receive side (odd interface) has been set to the receive mode, the receive mode continues, even between the "SDLC Receive Monitor", "SDLC Receive", and "SDLC Receive Continue" commands, until a "Halt", "Halt Immediate", or "Disable" command is issued by the control program.

   In half duplex receive mode, as soon as the line has been set to the receive mode, the receive mode continues between receive commands, as for duplex, until a "Halt", "Halt Immediate", "SDLC Transmit Control", "SDLC Transmit Data" or "Disable" command is issued by the control program.

SDLC Receive Command (X'13')

The SDLC Receive command is used to pass the address of the first buffer in a receive data buffer chain to the scanner, and to place the scanner in the ready to receive mode. It is used as follows:

1. On the receive side of a duplex line, before sending a frame with the poll bit on in the transmit side.

2. On a duplex or half duplex line, whenever an "I" frame is received with the "final" bit off, to prepare for the reception of the next frame.

Parameter Zone

Word 1	TCC	Modifiers	Offset	-
Word 2	Byte Count	First Receive Buffer Pointer		
Word 3	RA1	RA2	-	-
Word 4	-	-	-	-

Modifier Byte: This byte contains command modifier bits which have the following meaning:

Bit 1 - NCP Type Buffer: This bit, if off, indicates that the data transferred to the control program is in an NCP type buffer whose address is contained in the "First Receive Buffer Pointer". If this bit is on, the data is not in an NCP type buffer, but in a data area whose address is contained in the "First Receive Buffer Pointer" (in this case, the buffer prefix and buffer offset are not used).

Bit 2 - Two-Byte Address: This bit, if on, indicates that the SDLC address field (XA field) in the parameter area is two bytes long (XA1 and XA2), and that the expected address field in the receive frame is also two bytes long. If the bit is off, it indicates that the SDLC address field is one byte long (XA1 only).

Bit 3 - Two-Byte Control: This bit, if on, indicates that the SDLC control field (XC field) in the parameter area is two bytes long (XC1 and XC2), and that the expected control field in the receive frame is also two bytes long. If the bit is off, it indicates that the SDLC control field is one byte long (XC1 only).

Bit 4 - Compare Address: This bit, if off, indicates that the address field (1 or 2 bytes) in the received SDLC frame must be compared to the address in the parameter area, if primary side, or to the address specified by the Set Mode command (data area, bytes 10 and 11), if secondary side. If a mismatch occurs, the received frame is rejected (an **All Parties** address is always accepted, however).

Bit 5 - Answer Requested: This bit, when on, indicates that once the frame has been transmitted, a response is expected within a given period of time.

Offset: This is the number of bytes between the receive buffer address contained in word 2 and the start of the data.

Byte Count: This is the number of bytes actually available in the first buffer of a chain for storing data.

First Receive Buffer Pointer: This three-byte field contains the address of the buffer where the response is to be stored.

SDLC Receive Address 1 (RA1): This byte contains the SDLC station address to be compared against the address in the frame being received (first byte of a two-byte station address).

SDLC Receive Address 2 (RA2): This byte contains the second byte of the SDLC station address when a two-byte station address is used.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	Res. Count	Last Receive Buffer Used		
Word 3	RA1	RA2/RC1	RC1	RC2

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'13'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Residual Byte Count: This byte indicates the number of unused bytes remaining in the last buffer used.

Last Receive Buffer Used: This three-byte field indicates the last buffer that was used to hold the data received in response to the transmitted control byte(s).

SDLC Receive Address 1 (RA1): This byte contains the SDLC station address received in the current frame (first byte of a two-byte station address).

SDLC Receive Address 2 (RA2): This byte contains the second byte of the SDLC station address when a two-byte station address is used.

SDLC Receive Control 1 (RC1): This byte contains the SDLC control byte received in the current frame (first byte of a two-byte control field).

SDLC Receive Control 2 (RC2): This byte contains the second byte of the SDLC control field when a two-byte control field is used.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

1. The XA1 and XA2 fields in the parameter area are loaded by the control program only if the controller is the primary station, and if bit 4 of the modifier field specifies "compare address". The address in the receive frame is then compared with the contents of XA1/XA2. If the controller is a secondary station, and "compare address" is specified, the received address is compared with the address specified by the Set Mode command.

2. On a duplex link, the reply timeout is started on the receive side when a Transmit Control or Transmit Data command with "answer requested" in the modifier byte has been issued on the transmit side. This timeout is the maximum time allowed before a response must arrive.

   Once reception has started, the receive text timeout is started. This timeout is reset to zero at the end of a frame, or at buffer request. In this latter case, the receive text timeout is restarted by a Receive Continue command.

3. In duplex receive mode, as soon as the receive side (odd interface) has been set to the receive mode, the receive mode continues, even between the "SDLC Receive Monitor", "SDLC Receive", and "SDLC Receive Continue" commands, until a "Halt", "Halt Immediate", or "Disable" command is issued by the control program.

   In half duplex receive mode, as soon as the line has been set to the receive mode, the receive mode continues between receive commands, as for duplex, until a "Halt", "Halt Immediate", "SDLC Transmit Control", "SDLC Transmit Data" or "Disable" command is issued by the control program.

SDLC Receive Continue Command (X'14')

The SDLC Receive Continue command is used to assign additional buffers, when the scanner informs the CCU that data is being received for which no buffer is currently available.

Parameter Zone

Word 1	TCC	Modifiers	Offset	-
Word 2	Byte Count	First Receive Buffer Pointer		
Word 3	-	-	-	-
Word 4	-	-	-	-

Modifier Byte: This byte contains a single modifier bit which has the following meaning:

Bit 0 - NCP Type Buffer: This bit, if off, indicates that the data to be transferred from the control program is in an NCP type buffer whose address is contained in the "First Transmit Buffer Pointer". If this bit is on, the data is not in an NCP type buffer, but in a data area whose address is contained in the "First Transmit Buffer Pointer" (in this case, the buffer prefix and buffer offset are not used).

Offset: This is the number of bytes between the receive buffer address contained in word 2 and the start of the data.

Byte Count: This is the number of bytes actually available in the first buffer of a chain for storing data.

First Receive Buffer Pointer: This three-byte field contains the address of the buffer where the response is to be stored.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	Res. Count	Last Receive Buffer Used		
Word 3	RA1	RA2/RC1	RC1	RC2

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'14'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Residual Byte Count: This byte indicates the number of unused bytes remaining in the last buffer used.

Last Receive Buffer Used: This three-byte field indicates the last buffer that was used to hold the data received in response to the transmitted control byte(s).

SDLC Receive Address 1 (RA1): This byte contains the SDLC station address received in the current frame (first byte of a two-byte station address).

SDLC Receive Address 2 (RA2): This byte contains the second byte of the SDLC station address when a two-byte station address is used.

SDLC Receive Control 1 (RC1): This byte contains the SDLC control byte received in the current frame (first byte of a two-byte control field).

SDLC Receive Control 2 (RC2): This byte contains the second byte of the SDLC control field when a two-byte control field is used.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

1. Every time a buffer is filled, the receive text timeout (see SDLC Receive command) is restarted. This timeout is reset to zero at the end of a frame, or at buffer request. In this latter case, the receive text timeout is restarted by a Receive Continue command.

2. In duplex receive mode, as soon as the receive side (odd interface) has been set to the receive mode, the receive mode continues, even between the "SDLC Receive Monitor", "SDLC Receive", and "SDLC Receive Continue" commands, until a "Halt", "Halt Immediate", or "Disable" command is issued by the control program.

   In half duplex receive mode, as soon as the line has been set to the receive mode, the receive mode continues between receive commands, as for duplex, until a "Halt", "Halt Immediate", "SDLC Transmit Control", "SDLC Transmit Data" or "Disable" command is issued by the control program.

NCP X.21 COMMANDS

The NCP X.21 commands are used by the NCP and similar programs for controlling X.21 lines on public data networks. The data transmission itself is done using the SDLC commands.

X.21 Call Request Command (X'15')

The X.21 Call Request command is used to make an outgoing call. Before receiving this command, the line must be in the "controlled not ready" state, as set by the Set Mode command, or at the end of the clear process. If it is not, the command is rejected. The scanner then monitors for "proceed to select"; when this signal rises, it sends the selection signals (dialing digits), contained in an NCP type buffer. The end of the selection signals must be marked by a "+" sign, also in the buffer. After the transmission of the selection signals, the scanner monitors for possible call progress signals (CPSs); if any are received, the last one is transferred into byte 10 of the status area. If the command is completed without error, the line is set to the "ready for data" state, and the command ends.

If a negative CPS is received, the scanner automatically retries all calls except those that are designated as not retriable.

If the modifier bits specify "direct call", no selection signals are required.

Parameter Zone

Word 1	TCC	Modifiers	Offset	-
Word 2	Byte Count	First Selection Signal Buffer Pointer		
Word 3	Retry Timer Value		-	Retry Count
Word 4	-	-	-	-

Modifier Byte: This byte contains command modifier bits which have the following meaning:

Bit 0 - NCP Type Buffer: This bit, if off, indicates that the data to be transferred from the control program is in an NCP type buffer whose address is contained in the "First Transmit Buffer Pointer". If this bit is on, the data is not in an NCP type buffer, but in a data area whose address is contained in the "First Transmit Buffer Pointer" (in this case, the buffer prefix and buffer offset are not used).

Bit 2 - Retry Requested: This bit, if on, indicates that the X.21 call request is to be retried after a delay specified by the Retry Timer field.

Bit 6 - Direct Call: This bit, when on, indicates that on an X.21 call request, no selection signals are to sent.

Offset: This is the number of bytes between the first select character buffer pointer contained in word 2 and the start of the data.

Byte Count: This is the number of select characters to be transmitted.

First Selection Signal Buffer Pointer: This three-byte field contains the address of the buffer where the selection (dialing) characters are stored.

Retry Timer Value: This two-byte field contains the timer value after which the command is retried automatically.

Retry Count: This byte indicates the number of times the command is to be retried.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	-	-
Word 3	ILCS	Res. Retry Ct	Last CPS	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'15'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: This field contains the last status of the network; bit 7 indicates the result of a DTE Clear or DCE Clear (bit 7 = 0 means the Clear was successful; bit 7 = 1 means the Clear failed). If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Initial Line Communication Status (ILCS): This byte contains the LCS of the first error in an error recovery sequence (retriable Call Progress Signal, timeout, CPS error).

Residual Retry Count: This byte contains the residual value of the retry count from the parameter area.

Last Call Progress Signal: This byte contains the last CPS character received from the X.21 interface.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

X.21 Monitor Incoming Call Command (X'16')

The X.21 Monitor Incoming Call command is sent to the scanner to allow incoming calls on the line. Before receiving this command, the line must be in the "controlled not ready" state, as set by the Set Mode command, or at the end of the clear process. If it is not, the command is rejected. The scanner presents a "ready" state to the network, monitors for incoming calls, and receives the incoming line identification. After receiving the command, the line is set to the "ready for data" state, and the command ends.

The following errors are not reported to the CCU:

- Timeout on ready for data (T4 elapsed)
- DCE clear received

The line is cleared and set back to the monitor incoming call state. If an error occurs during the clearing phase (timeout during clear), the command is ended. The interface is then set back to the controlled not ready state, to be ready to monitor for incoming calls.

Parameter Zone

Word 1	TCC	-	-	-
Word 2	-	-	-	-
Word 3	-	-	-	-
Word 4	-	-	-	-

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	-	-
Word 3	ILCS	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'16'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: This field contains the last status of the network; bit 7 indicates the result of a DTE Clear or DCE Clear (bit 7 = 0 means the Clear was successful; bit 7 = 1 means the Clear failed). If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Initial Line Communication Status (ILCS): This byte contains the LCS of the first error in an error recovery sequence.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

X.21 DTE Clear Request Command (X'17')

The DTE Clear Request command is used to inform the scanner to clear the line, irrespective of the state of the line. The scanner clears the line and sets it to the "controlled not ready" state.

Parameter Zone

Word 1	TCC	-	-	-
Word 2	-	-	-	-
Word 3	-	-	-	-
Word 4	-	-	-	-

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	-	-
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'17'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: This field contains the last status of the network; bit 7 indicates the result of a DTE Clear or DCE Clear (bit 7 = 0 means the Clear was successful; bit 7 = 1 means the Clear failed). If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

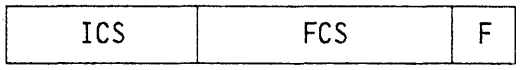
NCP BSC COMMANDS

The NCP BSC commands are used by the NCP and similar programs for data transfer over and control of BSC lines operating in the normal mode.

Transmit Control Byte

All of the NCP BSC commands (with the single exception of NCP BSC Receive Continue) use a "Transmit Control" byte. To avoid repetition, this byte is described fully here.

The transmit control byte contains coded instructions to the scanner. These instructions specify the initial and final control characters to be used in a transmission. It also contains an indicator which specifies whether leading graphics are to be sent. The transmit control byte has the following format:



ICS = Initial control sequence

FCS = Final control sequence

F = Leading graphics flag

Initial Control Sequence (ICS)

The initial control sequence field specifies the control sequence to be used at the beginning of a transmission. The ICS is decoded as follows:

Bits 0 1 2	Meaning
0 0 0	Control
0 0 1	Start of text (STX)
0 1 0	Transparent start of text (DLE-STX)
0 1 1	Start of header (SOH)
1 0 0	Special

Note: If the leading graphics flag is on, the leading graphics are transmitted **in front** of the ICS.

Final Control Sequence (FCS)

The final control sequence field specifies the control sequence to be used at the end of the transmission. Its meaning is determined in conjunction with the ICS as follows:

ICS	FCS	LGF	Meaning
000	0000	N	Turn line round and monitor
000	0011	Y	Send ENQ, turn around, and receive response. ENQ may be in a data stream of leading graphics (see note 2)
000	0110	Y	Send ACK-0, turn around and receive
000	0111	Y	Send NAK, turn around and receive
000	1101	N	Send RVI, turn around and receive
000	1110	Y	Send ACK-1, turn around and receive
000	1111	N	Send WACK, turn around and receive
001	0011	N	Send STX-ENQ (TTD), turn around and receive
001	1001	N	Send STX-data-ETX, turn around and receive
001	1010	N	Send STX-data-ETB, turn around and receive
010	0011	N	Send DLE-STX-data-DLE-ENQ, turn around and receive
010	0100	N	Send DLE-STX-data-DLE-ITB
010	1001	N	Send DLE-STX-data-DLE-ETX, turn around and receive
010	1010	N	Send DLE-STX-data-DLE-ETB, turn around and receive
011	0011	N	Send SOH-data-ENQ, turn around and receive
011	1001	N	Send SOH-data-ETX, turn around and receive
011	1010	N	Send SOH-data-ETB, turn around and receive
100	0000	N	Send EOT, turn around and monitor
100	0111	N	Send EOT, leading graphics, ENQ
100	1100	N	Send EOT, turn around and L2 interrupt
100	1110	N	Send DLE-EOT, turn around and L2 interrupt

Notes:

1. A Y in the column headed LGF indicates that leading graphics are possible with this bit configuration; an N indicates that leading graphics are not possible.

2. When a Receive or Control command is issued with polling or selection, it may be necessary to send EOT and put the line in control mode before sending the polling or selection characters. In this case, the ICS must be 100 instead of 000. This tells the scanner to send EOT before doing anything else.

NCP BSC Control Command (X'18')

The NCP BSC Control command is used for selection, control character transmission (such as TTD, WACK, or RVI), or to monitor for incoming data/control characters on point-to-point lines. The operation then depends on the bits of the transmit control byte:

- On point-to-point lines, if the transmit control byte contains X'80', the scanner sends EOT, turns the line around, and monitors for incoming data/control characters.
- On point-to-point lines, if the transmit control byte is X'00', the scanner monitors only incoming data/control characters.

Note: On point-to-point lines, if the transmit control byte is X'00' or X'80', and modifier bit 5 (Start Reply Timer) is on, the reply timer is started when monitoring.

On switched lines, a timeout is always started when monitoring.

The command must be issued to the even interface; it is rejected if issued to the odd interface.

Parameter Zone

Word 1	TCC	Modifiers	-	-
Word 2	Byte Count	Leading Graphics Data Address		
Word 3	-	Xmit Ctrl	-	-
Word 4	-	-	-	-

Modifier Byte: The bits of this byte have the following meaning:

Bit 0 - NCP Type Buffer: This bit should be set to 1 if a Leading Graphics Data Address is specified.

Bit 4 - ITB Mode: In transmit operations, this bit, if on, indicates that the byte following the ITB must be skipped.

In receive operations, the BCC character following the ITB is checked, and the EIB is built and stored in the buffer along with the ITB.

Bit 5 - Start Reply Timer: This bit indicates that the Reply Timer is to be started when monitoring a point-to-point line and the transmit control byte is X'00' or X'80'.

Bit 7 - Acknowledgment Expected: This bit indicates the type of acknowledgment expected: 0 = ACK0, 1 = ACK1.

Byte Count: This is the number of bytes of leading graphics data to be transferred.

Leading Graphics Data Address: This three-byte field contains the data address where the leading graphics (selection characters) are stored.

Transmit Control Byte: This byte contains control information for the scanner. See "Transmit Control Byte" at the beginning of the section "NCP BSC Commands" for a full description of this byte.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	-	-
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'18'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

NCP BSC Transmit Command (X'19')

The NCP BSC Transmit command is used to transmit the contents of a chain of buffers, turn the line around, and receive a reply. The operation then depends on the bits of the transmit control byte. At the end of the transmission (count = 0 and no data chaining), the line is turned around. When the expected response is received, the status is placed in the LCS byte of the status area.

This command may also be used for the Online Terminal Test (OLTT) by setting on modifier bit 3 (OLTT). In this case, the contents of the buffer (data and control characters) is transmitted in a similar way to the EP Transmit commands.

When the response is received, the status is placed in the LCS byte of the status area. If in addition, modifier bit 2 (second transparent write on OLTT) is on, the first two characters in the buffer are DLE followed by either ETB, ETX, or ITB. If the second character is ETB or ETX, the line is turned round and the response is received. If the second character is ITB, the BCC is sent, and data transmission continues.

The command must be issued to the even interface; it is rejected if issued to the odd interface.

Parameter Zone

Word 1	TCC	Modifiers	Offset (T)	-
Word 2	Byte Ct. (T)	First Transmit Buffer Pointer		
Word 3	-	Xmit Ctrl	-	-
Word 4	Byte Ct. (I)	Insert Data Address		

T = transmit buffer pointer; I = insert data address

Modifier Byte: This byte contains command modifier bits that have the following meaning:

Bit 0 - NCP Type Buffer: This bit, if off, indicates that the data to be transferred from the control program is in an NCP type buffer whose address is contained in the "First Transmit Buffer Pointer". If this bit is on, the data is not in an NCP type buffer, but in a data area whose address is contained in the "First Transmit Buffer Pointer" (in this case, the buffer prefix and buffer offset are not used).

Bit 1 - NCP Type Buffer: This bit, if off, indicates that the data transferred to the control program is in an NCP type buffer whose address is contained in the "Insert Data Address Pointer". If this bit is on, the data is not in an NCP type buffer, but in a data area whose address is contained in the "Insert Data Address Pointer" (in this case, the buffer prefix and buffer offset are not used).

Bit 2 - Second Transparent Write for OLTT: This bit is ignored if modifier bit 3 (OLTT) is off. If modifier bit 3 is on, it indicates that the first two characters in the transmit buffer are DLE followed by either ETB, ETX, or ITB.

Bit 3 - Online Terminal Test Mode: This bit, when on, indicates that the Online Terminal Test is running, and that the transmit buffers contain data and control characters.

Bit 4 - ITB Mode: In transmit operations, this bit, if on, indicates that the byte following the ITB must be skipped.

In receive operations, the BCC character following the ITB is checked, and the EIB is built and stored in the buffer along with the ITB.

Bit 5 - Data Chain: For Transmit commands, this bit indicates that more data is available in another buffer chain when the current chain has been completely transmitted and end-of-chain is reached ("subblock" is set in the ending status LCS).

Bit 6 - Insert Data: This bit, when on, indicates that data, taken from a data area, must be transmitted after the initial control sequence, but before the data in the NCP type buffer.

Bit 7 - Acknowledgment Expected: This bit indicates the type of acknowledgment expected: 0 = ACK0, 1 = ACK1.

Offset (T): This is the number of bytes between the first transmit buffer pointer contained in word 2 and the start of the data to be transmitted.

Byte Count (T): This is the number of bytes of data contained in the first transmit buffer.

First Transmit Buffer Pointer: This three-byte field contains the address of the first transmit buffer.

Transmit Control Byte: This byte contains control information for the scanner. See "Transmit Control Byte" at the beginning of the section "NCP BSC Commands" for a full description of this byte.

Byte Count (I): This is the number of bytes of insert data contained in the insert data address pointer.

Insert Data Address Pointer: This three-byte field contains the address of the data area containing the insert data in word 4.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	Res. Count	Last Tx Buffer Pointer (if length check)		
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'19'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Residual Byte Count: This byte indicates the number of unused bytes in the last transmit buffer used.

Last Transmit Buffer Pointer Used: This three-byte field indicates the last buffer that was used to hold the transmitted data. This field is applicable only if the operation ended with a length check in the ending status.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

NCP BSC Transmit Continue Command (X'1A')

The NCP BSC Transmit Continue command is used to provide a chain of buffers when transmit data chaining. The transmit control byte must be exactly the same as the one specified in the preceding NCP BSC Transmit command.

This command must be issued to the even interface; it is rejected if issued to the odd interface.

Parameter Zone

Word 1	TCC	Modifiers	Offset	-
Word 2	Byte Count	First Transmit Buffer Pointer		
Word 3	-	Xmit Ctrl	-	-
Word 4	-	-	-	-

Modifier Byte: This byte contains command modifier bits that have the following meaning:

Bit 0 - NCP Type Buffer: This bit, if off, indicates that the data to be transferred from the control program is in an NCP type buffer whose address is contained in the "First Transmit Buffer Pointer". If this bit is on, the data is not in an NCP type buffer, but in a data area whose address is contained in the "First Transmit Buffer Pointer" (in this case, the buffer prefix and buffer offset are not used).

Bit 2 - Second Transparent Write for OLTT: This bit is ignored if modifier bit 3 (OLTT) is off. If modifier bit 3 is on, it indicates that the first two characters in the transmit buffer are DLE followed by either ETB, ETX, or ITB.

Bit 3 - Online Terminal Test Mode: This bit, when on, indicates that the Online Terminal Test is running, and that the transmit buffers contain data and control characters.

Bit 5 - Data Chain: For Transmit commands, this bit indicates that more data is available in another buffer chain when the current chain has been completely transmitted and end-of-chain is reached ("subblock" is set in the ending status LCS).

Bit 7 - Acknowledgment Expected: This bit indicates the type of acknowledgment expected: 0 = ACK0, 1 = ACK1.

Offset: This is the number of bytes between the first transmit buffer pointer contained in word 2 and the start of the data to be transmitted.

Byte Count: This is the number of bytes of data contained in the first buffer pointer.

First Transmit Buffer Pointer: This three-byte field contains the address of the first transmit buffer.

Transmit Control Byte: This byte contains control information for the scanner. See "Transmit Control Byte" at the beginning of the section "NCP BSC Commands" for a full description of this byte.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	Res. Count	Last Tx Buffer Pointer (if length check)		
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'1A'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Residual Byte Count: This byte indicates the number of unused bytes in the last transmit buffer pointer used.

Last Transmit Buffer Pointer Used: This three-byte field indicates the last buffer that was used to hold the transmitted data. This field is applicable only if the operation ended with a length check in the ending status.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

NCP BSC Receive Command (X'1B')

The NCP BSC Receive command is used to poll terminals or to send a response to a received block, and to receive either text or another reply.

This command must be issued to the even interface; it is rejected if issued to the odd interface.

Parameter Zone

Word 1	TCC	Modifiers	Offset (R)	-
Word 2	Byte Ct. (R)	First Receive Buffer Pointer		
Word 3	-	Xmit Ctrl	-	-
Word 4	Byte Ct. (P)	Poll Characters or LG Data Address		

R = receive buffer pointer; P = poll/LG data address

Modifier Byte: This byte contains command modifier bits that have the following meaning:

Bit 0 - NCP Type Buffer: This bit, if off, indicates that the data transferred to the control program is in an NCP type buffer whose address is contained in the "First Receive Buffer Pointer". If this bit is on, the data is not in an NCP type buffer, but in a data area whose address is contained in the "First Receive Buffer Pointer" (in this case, the buffer prefix and buffer offset are not used).

Bit 1 - NCP Type Buffer: This bit should be set to 1 if Poll Characters or a Leading Graphics Data Address is specified.

Bit 4 - End of Intermediate Transmission Block (ITB) Mode: In transmit operations, this bit, if on, indicates that the byte following the ITB must be skipped.

In receive operations, the BCC character following the ITB is checked and the EIB is built and stored in the buffer along with the ITB.

Offset (R): This is the number of bytes between the first receive buffer pointer contained in word 2 and the start of the data.

Byte Count (R): This is the number of bytes of data contained in the first receive buffer.

First Receive Buffer Pointer: This three-byte field contains the address of the first receive buffer.

Transmit Control Byte: This byte contains control information for the scanner. See "Transmit Control Byte" at the beginning of the section "NCP BSC Commands" for a full description of this byte.

Byte Count (P): This is the number of poll character bytes or bytes of leading graphics data contained in the poll characters or leading graphics data address pointer.

Poll Characters or Leading Graphics Data Address Pointer: This three-byte field contains the address of the data area containing the poll characters or leading graphics data in word 4.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	Res. Count	Last Receive Buffer Used		
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'1B'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Residual Byte Count: This byte indicates the number of unused bytes in the last receive buffer used.

Last Receive Buffer Used: This three-byte field indicates the last buffer that was used to hold the received data.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

NCP BSC Receive Continue Command (X'1C')

The NCP BSC Receive Continue command is used to provide a chain of receive data buffers to the scanner.

This command must be issued to the even interface; it is rejected if issued to the odd interface.

Parameter Zone

Word 1	TCC	Modifiers	Offset	-
Word 2	Byte Count	First Receive Buffer Pointer		
Word 3	-	-	-	-
Word 4	-	-	-	-

Modifier Byte: This byte contains one command modifier bit with the following meaning:

Bit 0 - NCP Type Buffer: This bit, if off, indicates that the data transferred to the control program is in an NCP type buffer whose address is contained in the "First Receive Buffer Pointer". If this bit is on, the data is not in an NCP type buffer, but in a data area whose address is contained in the "First Receive Buffer Pointer" (in this case, the buffer prefix and buffer offset are not used).

Offset: This is the number of bytes between the first receive buffer pointer contained in word 2 and the start of the data.

Byte Count: This is the number of bytes of data contained in the first receive buffer.

First Receive Buffer Pointer: This three-byte field contains the address of the first receive buffer.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	Res. Count	Last Receive Buffer Used		
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'1C'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred.

Residual Byte Count: This byte indicates the number of unused bytes in the last receive buffer used.

Last Receive Buffer Used: This three-byte field indicates the last buffer that was used to hold the received data.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

EP BSC COMMANDS

The EP BSC commands are used by the EP and similar programs for data transfer and control on BSC lines operating in the normal mode.

EP BSC Transmit Initial Command (X'20')

The EP BSC Transmit Initial command is used to place a line in the transmit state, first checking the status of the line. If the line is receiving (in phase), or if it has received data since the last CCU level 2 interrupt, the scanner ends the command with an "in phase" final status and takes no other action. If the line is not receiving, the scanner ends the command with a "transmit initial accepted" final status and prepares the line for transmission.

The command must be issued to the even interface; it is rejected if issued to the odd interface.

Parameter Zone

Word 1	TCC	-	-	-
Word 2	-	-	-	-
Word 3	-	-	-	-
Word 4	-	-	-	-

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	-	-
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'20'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

1. Any errors detected during the execution of this command (after status presentation) are presented to the next command received.

2. The scanner assumes data chaining. The line remains in the transmit state, and SYN characters are transmitted (SYN fill) until the next command is received.

EP BSC Transmit SYN Command (X'21')

The EP BSC Transmit SYN command is used to provide a variable time delay. The delay depends on the line speed. The scanner does this by transmitting a specified number of SYN characters before presenting the ending status. The Transmit SYN command must be preceded by a Transmit Initial, by a Transmit Data, or by another Transmit SYN command.

Parameter Zone

Word 1	TCC	-	-	-
Word 2	SYN count	-	-	-
Word 3	-	-	-	-
Word 4	-	-	-	-

SYN Count: This byte contains the number of SYN characters to be transmitted before presenting ending status.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	-	-
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'21'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

EP BSC Transmit Data Command (X'22')

The EP BSC Transmit Data command is used to transmit the contents of one data buffer on the line. The Transmit Data command must be preceded by a Transmit Initial, Transmit SYN, or another Transmit Data command, otherwise it is rejected.

The command must be issued to the even interface; it is rejected if issued to the odd interface.

EIB/ITB Handling

The handling of ITB characters in the transmitted data depends on bit 6 (ITB is data) and bit 7 (EIB mode) of the set mode data as specified by the Set Mode command. They work together as shown in the following table:

EIB Mode	ITB= Data	Meaning
0	0	Compute BCC, send it after ITB, ETB, and ETX
0	1	Treat ITB as data
1	-	Ignore ITB = Data bit, compute BCC, send it after ITB, ETB, and ETX.

Transparent Mode

At the start of transmission, the non-transparent mode is assumed. The transparent mode is entered when the DLE-STX sequence is detected in the data to be transmitted. Once the transparent mode is entered, the scanner automatically inserts a second DLE each time the DLE bit combination (data) is detected. This second DLE is not included in the CRC computation.

The transparent mode is ended when the control program issues a Transmit Data command in which the "second transparent write" bit (modifier bit 2) is on. The control program must provide the exact byte count:

- **DLE-ETB, DLE-ETX, DLE-ENQ:** The byte count = 2, and the scanner leaves the transparent mode.
- **DLE-ITB:** The byte count is variable. If additional data follows the DLE-ITB combination, the scanner leaves the transparent mode after sending the DLE-ITB, and then continues sending the remainder of the data in non-transparent mode.

  Note: A new DLE-STX sequence in the data puts the scanner back into the transparent mode. However, the transparent ending sequence must not be used again in the data transmitted of this second transparent write. An additional second transparent write must be used to end the transparent mode.

Parameter Zone

Word 1	TCC	Modifiers	-	-
Word 2	Byte Count	Transmit Data Pointer		
Word 3	-	-	-	-
Word 4	-	-	-	-

Modifier Byte: This byte contains command modifier bits that have the following meaning:

Bit 2 - Second Transparent Write: This bit, when on, indicates that transparent mode must end when the first two characters of the buffer have been transmitted. These two characters must be one of the following pairs:

- DLE-ETB
- DLE-ETX
- DLE-ENQ
- DLE-ITB

Bit 5 - Data Chain: On Transmit commands, this bit indicates that more data is available in another buffer chain when the current chain has been transmitted. Buffer requested is set in the ending status.

Byte Count: This is the number of bytes of data contained in the first transmit buffer.

Transmit Data Pointer: This three-byte field contains the address of the data area containing the data to be transmitted.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	-	-
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'22'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Consideration:

1. If the data count is zero and data chaining is not specified, nothing is transmitted, the line is turned around, and the command is ended.

2. If the data count is zero and data chaining is specified, nothing is transmitted and the command is ended.

EP BSC Poll Command (X'23')

The EP BSC Poll command is used to poll terminals on a multipoint line. The scanner cycle steals the polling information from CCU storage, initiates polling, and handles negative responses. A level 2 interrupt occurs when a positive response is received to polling, or when the end of the polling list is reached. The command is then ended.

The Poll command must be preceded by a Transmit Initial or by another Poll command, otherwise it is rejected.

The command must be issued to the even interface; it is rejected if issued to the odd interface.

Parameter Zone

Word 1	TCC	Modifiers	-	-
Word 2	Byte Count	Poll Data Pointer		
Word 3	-	-	-	-
Word 4	-	-	-	-

Modifier Byte: This byte contains command modifier bits that have the following meaning:

Bit 5 - Data Chain: On Transmit commands, this bit indicates that more data is available in another buffer chain when the current chain has been transmitted. Buffer requested is set in the ending status.

Byte Count: This is the number of bytes of poll data bytes to be used for polling.

Poll Data Pointer: This three-byte field contains the data address where the poll data is stored.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	-	-
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'23'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Consideration:

1. If a timeout occurs, a timeout ending status will also be presented to the Receive command which follows the Poll command.

2. The index byte of the last polled entry will always be the first data byte presented to the Receive command which follows the Poll command.

EP BSC Receive Command (X'24')

The EP BSC Receive command is used to transfer one buffer of data into main storage. If the line has already started to receive (pseudo-read), the scanner transfers the data to the CCU.

A "lost data" condition occurs if all the line interface buffers in the scanner has been filled with received data and no Receive command has been received by the scanner. All subsequent received data is flushed. The "lost data" condition is set in the ending status when the scanner finally receives a Receive command.

If the line has not started to receive when the Receive command is issued, the scanner starts a three-second timeout; if a control character (other than SYN) is not received within this three-second period, the command is ended.

EIB/ITB Handling

The handling of ITB characters in the received data depends on bit 6 (ITB is data) and bit 7 (EIB mode) of the set mode data as specified by the Set Mode command. They work together as shown in the following table:

EIB Mode	ITB= Data	Meaning
0	0	Treat ITB as a control character
0	1	Treat ITB as data
1	-	Ignore ITB = Data bit, check BCC, generate and insert EIB character, treat ITB as a control character.

When an overrun occurs, the data is flushed either to an ITB (if ITB is a control character) or to an ending condition (ETB, ETX, ENQ or timeout). The overrun bit (SCF bit 2) is set in the status for the command to be processed (or being processed), and if "EIB Mode" is set, the overrun bit (bit 5) is set in the EIB character.

When an ITB (if ITB is a control character), an ETB, or an ETX is detected, the BCC (next character in the received data) is compared with the computed BCC. If there is any difference, there has been an error in the received data. The data check bit (SES bit 4) is set in the status for the command being processed, and if EIB Mode is on, the data check bit (bit 4) is set in the EIB character. Data check is set only in the command in which it occurred.

When overrun and data check occur together, if the error occurs and an ITB is detected, normal processing continues for the rest of the data after the ITB. If the error occurs and an ending condition is detected, EOM (SCF bit 5) is also set.

Transparent Mode

The scanner enters transparent mode when a DLE-STX sequence is received. From this point on, the scanner automatically deletes the second DLE of a DLE-DLE sequence; the deleted DLEs are not computed in the BCC. The scanner leaves transparent mode when a DLE-ETB, DLE-ETX, DLE-ENQ, or DLE-ITB is detected.

Parameter Zone

Word 1	TCC	-	-	-
Word 2	Byte Count	Receive Buffer Address		
Word 3	-	-	-	-
Word 4	-	-	-	-

Byte Count: This is the length of the receive buffer.

Receive Buffer Address: This three-byte field contains the data address where the received data is to be stored.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	Res. Count	-	-	-
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'24'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred.

Residual Byte Count: This byte indicates the number of unused bytes in the last receive buffer used.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

All received characters, other than SYN, DLE-SYN, and the second DLE of a DLE-DLE sequence are transferred to the buffer.

EP BSC Receive Continue Command (X'25')

The EP BSC Receive Continue command is used to provide a new buffer as requested when ending a previous Receive or Receive Continue command with a buffer request.

An overrun condition occurs if the line interface buffer in the scanner has been filled with received data and a Receive Continue command has not been received. Data subsequently received is flushed to an ITB or to an ending condition as described above under "EP BSC Receive Command". The overrun condition is set in the ending status when the Receive Continue command is finally received.

If more than three seconds delay occurs between SYN and non-SYN characters, a timeout condition occurs.

The command must be issued to the even interface; it is rejected if issued to the odd interface.

EIB/ITB Handling

The handling of ITB characters is the same as for the EP BSC Receive command.

Transparent Mode

Transparent mode operation is the same as for the EP BSC Receive command.
Parameter Zone

Word 1	TCC	-	-	-
Word 2	Byte Count	Receive Buffer Address		
Word 3	-	-	-	-
Word 4	-	-	-	-

Byte Count: This is the length of the receive buffer.

Receive Buffer Address: This three-byte field contains the data address where the received data is to be stored.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	Res. Count	-	-	-
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'25'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred.

Residual Byte Count: This byte indicates the number of unused bytes in the last receive buffer used.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

The special considerations are as described above under "EP BSC Receive Command".

EP BSC Prepare Command (X'26')

The EP BSC Prepare command is used to monitor for the "in phase" condition. The command is rejected if the line has not been previously enabled.

The command must be issued to the even interface; it is rejected if issued to the odd interface.

Parameter Zone

Word 1	TCC	-	-	-
Word 2	-	-	-	-
Word 3	-	-	-	-
Word 4	-	-	-	-

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	-	-
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'26'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

EP BSC Monitor for Phase Command (X'27')

The EP BSC Monitor for Phase command is used to set the line in receive mode, monitoring for phase (SYN-SYN). The command is ended without waiting for the phase condition. When issued in place of a Transmit Continue command, it may be used to end the transmission and set the line back into the receive mode. The command is rejected if the line has not been previously enabled.

The command must be issued to the even interface; it is rejected if issued to the odd interface.

Parameter Zone

Word 1	TCC	-	-	-
Word 2	-	-	-	-
Word 3	-	-	-	-
Word 4	-	-	-	-

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	-	-
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'27'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

EP BSC Address Prepare Command (X'28')

The EP BSC Address Prepare command is used when the controller is defined as a tributary station in a non-centralized multipoint network. The Tributary Support bit (byte 2, bit 1) of the Set Mode data must be on. The command is used to monitor received data, looking for a match between the receive data and the Selection, Group, or Poll address as defined in the Set Mode data, bytes 7 through 9. When a match occurs, the status is set into the status area, and a level 2 interrupt is raised. It is similar to a Receive command, receiving data in the same way, but discarding all data until an EOT character is received.

When an EOT character is received, the line is synchronized, and the next character received is checked against the three addresses in the set mode data. The SCF in the ending status is set to X'4C' for a Selection or Group Address compare, and to X'47' for a Poll Address compare.

Once issued, line traffic is monitored continuously until one of the following conditions is detected:

- An address match occurs.
- A Halt or Halt Immediate command is issued.
- An error condition occurs.

Parameter Zone

The parameter zone is not used by the Address Prepare command.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	-	-
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'27'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

1. The command is rejected if the line has not been previously enabled.

2. If the Selection Address or the Group Address matches, a Receive command should be the next command issued to the line.

3. If the Poll Address matches, a Transmit command should be the next command issued to the line.

4. After an EOT character is detected, the effect of the Ignore Bad Pad bit is as follows:

   - If the pad is good, check the next character for an address compare.
   - If the pad is bad and the Ignore Bad Pad bit is on, check the next character for an address compare.
   - If the pad is bad and the Ignore Bad Pad bit is off, set a Bad Pad status in the SES, then continue the search for EOT and a good pad.

5. The Address Prepare command may remain active for long periods of time, depending on the traffic in the network. If the master station fails to poll, or there is no traffic, the scanner continues searching for character phase on the receive side.

EP BSC Search Command (X'29')

The EP BSC Search command is a receive type command, used when the controller is defined as a master station in a non-centralized multipoint network. The Tributary Support bit (byte 2, bit 1) of the Set Mode data must be on. The command is used to monitor line traffic, looking for data intended for this station. The following situations may occur:

- The line data is intended for this station; the control program should issue a receive command to receive the data.
- The line data traffic has ended; the control program should issue a new Poll command to resume polling.

Note: The command is rejected if the command does not follow a Poll, Transmit, or another Search command.

Operation after Poll with Data Intended for this Station

The scanner prepares an "Index Byte" and transmits it to the CCU. The next character is a Selection Address or a Group Address; if the data is intended for this station, the first character matches the Set Mode data byte 7 (Selection Address) or byte 8 (Group Address). All the received data, up to and including the ENQ character, is cycle stolen to the CCU buffer. A status with the SCF set to X'4C' is transferred to the status area, and a level 2 interrupt is raised. A Receive command should follow to transfer the data received **after** the ENQ to the CCU.

Note: If a bad pad is received after the ENQ character, and the Ignore Bad Pad bit is off, the scanner continues to search for an ENQ character followed by a good pad.

Operation after Poll with Data Not Intended for this Station

The scanner prepares an "Index Byte" and transmits it to the CCU. The next character is a Selection Address or a Group Address; if the data is not intended for this station, the first character does not match the Set Mode data byte 7 (Selection Address) or byte 8 (Group Address).

The scanner therefore starts searching for SOH or STX (indicating the end of traffic on the line), and transfers all the data, from the "Index Byte" up to but not including the SOH/STX, to the CCU. If the buffer is filled before all the data is transferred or SOH/STX is detected, the command ends with an SCF of X'48'; another Search command should be issued to transfer the remaining data and continue the search for SOH/STX (this is the only occasion in which a Search command should follow another Search command).

When SOH/STX is received, the scanner ends data transfer, but continues to search for an EOT character. It then ends the command with SCF = X'0C' and SES = X'40'.

Note: If an EOT character is detected **before** SOH/STX, the command is ended with SCF = X'04' and SES = X'40'.

Operation after Poll (via Transmit Command) with EOT Received

This is the normal negative response to a poll done using a Transmit Command. When the EOT is detected, the scanner ends the command with SCF = X'04' and SES = X'40', and raises a level 2 interrupt request.

Note: If the pad character is bad, and the Ignore Bad Pad bit is off, the EOT character is abandoned, and the scanner continues to search the data stream for a valid selection sequence. Parameter Zone

Word 1	TCC	-	-	-
Word 2	Byte Count	Receive Buffer Address		
Word 3	-	-	-	-
Word 4	-	-	-	-

Byte Count: This is the length of the receive buffer.

Receive Buffer Address: This three-byte field contains the data address where the received data is to be stored.

Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	Res. count	-	-	-
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'27'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred.

Residual Byte Count: This byte indicates the number of unused bytes in the last receive buffer used.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

CHARACTER MODE COMMANDS

The "character mode" emulates a 3704/5 Communications Scanner Type 2. It uses an interface control word (ICW) to control each line as in the 3704/5. This ICW is located in the scanner storage, and is loaded from the parameter area of the PSA.

The character mode commands are used by the control program (NCP or EP) to control BSC and start/stop lines. IOH and IOHI instructions that contain character mode commands have the character bit (bit 14 of the second halfword) set to 1.

Character Mode Write ICW Command (X'40')

The Character Mode Write ICW command is used to load the ICW (located in the scanner) from the parameter area of the PSA. The fields that are to be loaded depend on the bits of the modifier field. On completion of the command, the contents of the ICW are stored in the status area of the PSA and a Level 2 interrupt occurs.

Note: A Write ICW command may be overridden by another Write ICW command.

Parameter Zone

Word 1	TCC	Modifiers	SCF	PDF
Word 2	LCD/PCF	SDF	Quiet count	-
Word 3	-	-	-	-
Word 4	-	-	-	-

Note: In the text that follows, because the meaning of certain fields depends on the mode of operation (BSC or start/stop) of the line, they are not described strictly in the order in which they appear in the PSA.

Modifier Byte: This byte contains command modifier bits that have the following meaning:

Bit 0 - Set SCF and PDF

Note: If the PCF = 7 (receive/receive in phase), only the SCF is set.

Bit 1 - Set SDF

Bit 2 - Set PCF

Bit 5 - Line Quiet Test: This bit indicates that when a start/stop receive operation ends, a delay must occur to allow the line to stabilize before attempting any other operation on the line. The delay is determined via the "Quiet Count" field of the parameter area. Each unit represents one character delay time. Thus, a count of three introduces a delay of three characters; the operation ends at the end of the third character time. During this last character time, the scanner assembles a dummy character using the input from the line, and stores it in the PDF field (if more than 8 bit start/stop transmission is used, the high order bits are stored in the LCD part of the LCD/PCF field in the status area). The PDF/LCD fields then contains an indication of the activity on the line; "all ones" indicates a quiet line.

Bit 6 - Set SCF Only

Parallel Data Field (PDF): This field is used as a character buffer. For transmit operations, the character to be sent is loaded into the PDF from the parameter area using a Write ICW command with modifier bit 0 set to 1. The scanner then transfers the character to the serial data field (SDF) and transmits it to the interface.

For receive operations, the scanner assembles the character into the SDF, transfers it into the PDF, and sets a level 2 interrupt. The character may be recovered for the use of the control program by means of a another Write ICW command. The format of the PDF depends on the type of transmission control employed as defined by the line control definer field (LCD), as in the table below:

		PDF bit positions							
Type of control	LCD	0	1	2	3	4	5	6	7
Start/stop 9/6	'0'	0	0	X6	X5	X4	X3	X2	X1
Start/stop 8/5	'2'	0	0	0	X5	X4	X3	X2	X1
Start/stop 9/7	'4'	0	X7	X6	X5	X4	X3	X2	X1
Start/stop 10/7	'5'	0	X7	X6	X5	X4	X3	X2	X1
Start/stop 10/8	'6'	X8	X7	X6	X5	X4	X3	X2	X1
Start/stop 11/8	'7'	X8	X7	X6	X5	X4	X3	X2	X1
BSC EBCDIC	'C'	X8	X7	X6	X5	X4	X3	X2	X1
BSC ASCII	'D'	X8	X7	X6	X5	X4	X3	X2	X1

Note: The bit marked X1 is always the first to be transmitted (and received).

Serial Data Field (SDF): The SDF is used as a character deserializer/serializer field. On transmit operations, a character from the PDF is transferred to the SDF by the scanner, and then sent, one bit at a time, to the line.

On receive operations, the bits coming from the line are assembled bit by bit into the SDF by the scanner. When a character has been assembled, it is transferred into the PDF by the scanner.

Line Control Definer (LCD) Field: This field comprises bits 0 through 3 of the LCD/PCF field, and is used during normal transmit and receive operations to define the type of line control. Unlike the other fields of the ICW, this field is not set by the Write ICW command, but by the Set Mode command.

The four bits are decoded as one hexadecimal digit having the following meaning:

Hex	Meaning
0	Start/stop 9/6
1	(not used)
2	Start/stop 8/5
3	(not used)
4	Start/stop 9/7
5	Start/stop 10/7
6	Start/stop 10/8
7	Start/stop 11/8
8	(not used)
9	(not used)
A	(not used)
B	(not used)
C	BSC (EBCDIC)
D	BSC (ASCII)
E	(not used)
F	(not used)

The meaning of each of the different LCD states is described below. In the descriptions which follow, the first **information** bit of a transmitted or received character is designated as X1. In the case of start/stop operation, start and stop bits are inserted or deleted automatically by the scanner.

LCD State X'0' - Start/Stop 9/6 Bit Control: This state indicates a start/stop transmission with a 9/6 format, that is, one start bit, six data bits, and two stop bits. When a character is sent to the interface, the six data bits must be situated in bits 2 through 7 of the PDF as shown in the table below. Bits received from the interface will be found in the same positions in the PDF when the scanner requests a character service interrupt.

PDF bit positions	0	1	2	3	4	5	6	7
Character bits	0	0	X6	X5	X4	X3	X2	X1

LCD State X'2' - Start/Stop 8/5 Bit Control: This state indicates a start/stop transmission with a 8/5 format, that is, one start bit, five data bits, and two stop bits. When a character is sent to the interface, the five data bits must be situated in bits 3 through 7 of the PDF as shown in the table below. Bits received from the interface will be found in the same positions in the PDF when the scanner requests a character service interrupt.

PDF bit positions	0	1	2	3	4	5	6	7
Character bits	0	0	0	X5	X4	X3	X2	X1

LCD State X'4' - Start/Stop 9/7 Bit Control: This state indicates a start/stop transmission with a 9/7 format, that is, one start bit, seven data bits, and one stop bit. When a character is sent to the interface, the seven data bits must be situated in bits 1 through 7 of the PDF as shown in the table below.

Bits received from the interface will be found in the same positions in the PDF when the scanner requests a character service interrupt.

PDF bit positions	0	1	2	3	4	5	6	7
Character bits	0	X7	X6	X5	X4	X3	X2	X1

LCD State X'5' - Start/Stop 10/7 Bit Control: This state indicates a start/stop transmission with a 10/7 format, that is, one start bit, seven data bits, and two stop bits. When a character is sent to the interface, the seven data bits must be situated in bits 1 through 7 of the PDF as shown in the table below. Bits received from the interface will be found in the same positions in the PDF when the scanner requests a character service interrupt.

PDF bit positions	0	1	2	3	4	5	6	7
Character bits	0	X7	X6	X5	X4	X3	X2	X1

LCD State X'6' - Start/Stop 10/8 Bit Control: This state indicates a start/stop transmission with a 10/8 format, that is, one start bit, eight data bits, and one stop bit. When a character is sent to the interface, the eight data bits must be situated in bits 0 through 7 of the PDF as shown in the table below. Bits received from the interface will be found in the same positions in the PDF when the scanner requests a character service interrupt.

PDF bit positions	0	1	2	3	4	5	6	7
Character bits	X8	X7	X6	X5	X4	X3	X2	X1

LCD State X'7' - Start/Stop 11/8 Bit Control: This state indicates a start/stop transmission with a 11/8 format, that is, one start bit, eight data bits, and two stop bits. When a character is sent to the interface, the eight data bits must be situated in bits 0 through 7 of the PDF as shown in the table below. Bits received from the interface will be found in the same positions in the PDF when the scanner requests a character service interrupt.

PDF bit positions	0	1	2	3	4	5	6	7
Character bits	X8	X7	X6	X5	X4	X3	X2	X1

LCD State X'C' - BSC EBCDIC Line Control: This state indicates a binary synchronous transmission using the EBCDIC SYN character. When a character is sent to the interface, the eight data bits must be situated in bits 0 through 7 of the PDF as shown in the table below. Bits received from the interface will be found in the same positions in the PDF when the scanner requests a character service interrupt. The SYN character (X'32') provides for automatic detection of the first phase character during a receive operation.

PDF bit positions	0	1	2	3	4	5	6	7
Character bits	X8	X7	X6	X5	X4	X3	X2	X1

LCD State X'D' - BSC ASCII Line Control: This state indicates a binary synchronous transmission using the USASCII SYN character. When a character is sent to the interface, the eight data bits must be situated in bits 0 through 7 of the PDF as shown in the table below. Bits received from the interface will be found in the same positions in the PDF when the scanner requests a character service interrupt. The SYN character (X'16') provides for automatic detection of the first phase character during a receive operation.

PDF bit positions	0	1	2	3	4	5	6	7
Character bits	X8	X7	X6	X5	X4	X3	X2	X1

Other Parameter Fields: The meaning of the bits of the remaining parameter fields (SCF and PCF) depends on whether the line is operating in start/stop or in BSC mode, as defined by the line control definer (LCD) field. They are discussed separately below.

Start/Stop Operation

There are six different types of start/stop operation, defined by the LCD as follows:

		PDF bit positions							
Type of control	LCD	0	1	2	3	4	5	6	7
Start/stop 9/6	'0'	0	0	X6	X5	X4	X3	X2	X1
Start/stop 8/5	'2'	0	0	0	X5	X4	X3	X2	X1
Start/stop 9/7	'4'	0	X7	X6	X5	X4	X3	X2	X1
Start/stop 10/7	'5'	0	X7	X6	X5	X4	X3	X2	X1
Start/stop 10/8	'6'	X8	X7	X6	X5	X4	X3	X2	X1
Start/stop 11/8	'7'	X8	X7	X6	X5	X4	X3	X2	X1

Note: The bit marked X1 is always the first to be transmitted. **Secondary Control Field (SCF):** This field is used as a sense, status, and operation modifier field between the control program and the scanner. The bits of the SCF have the following meaning:

Bit	Meaning
0	Stop bit check/receive break/line halted
1	Service request interlock
2	Character overrun/underrun
3	Modem check
4	Receive line signal detector
5	Start bit detected
6	Program flag
7	Pad flag

Bit 0 - Stop Bit Check/Receive Break/Line Halted: On receive operations (PCF state X'7'), the stop bit is checked after each character is received. If the stop bit is a "space" instead of a "mark", the scanner signals this condition by setting bit 0.

On transmit operations (PCF state X'9'), the "receive data" line is checked for a space condition (0). If a space condition is detected, bit 0 is set to one. When the control program detects that this bit is on for two successive characters, it must interpret it as a receive break signal.

This bit is also set if the line has been halted; in this case, the service request interlock bit (SCF bit 1) is also on.

When bit 0 is 1, a Level 2 interrupt occurs.

Bit 1 - Service Request Interlock: This bit is set if the scanner detects that data transfer or control servicing is required between the control program and the PDF. The bit is also set if the line has been halted. A Level 2 interrupt request is set. The control program must reset this bit via a Write ICW command after the interrupt has been accepted.

If bit 1 is set, bits 2 and 3 are set to zero.

Bit 2 - Character Overrun/Underrun:

- Overrun occurs in the receive state. The bit is set by the scanner if three characters have been received and there is no outstanding Write ICW command. This error is normally caused by an instantaneous peak overload situation. Errors of this type should not normally occur in the average installation, and only infrequently in high throughput installations. When a character overrun occurs, the characters that follow are lost.

- Underrun occurs in the transmit state. The line is put into mark status until the control program resumes transmission and changes the PDF field to another character or until the primary control field is changed from the transmit state.

When this bit is set, the service request interlock bit (SCF bit 1) is set to zero, and a CCU level 2 interrupt request is set.

Bit 3 - Modem Check: This bit indicates that one of the following conditions has occurred:

1. The data set ready line is inactive when the PCF field is in states X'7', X'8', X'9', X'A', X'B', X'C', or X'D'.

2. The clear to send line is inactive when the PCF field is in states X'9', X'A', X'B', or X'D'.

3. The line receive line signal detect is inactive, the pad flag bit (SCF bit 7) is on, and the PCF is in state X'7' (receive).

If the modem check bit is on, the service request interlock (SCF bit 1) is set to zero, the PCF is set to X'0' (No-Op), and a Level 2 interrupt occurs. In addition, if the line is designated as "Secure", data terminal ready is dropped.

Bit 4 - Received Line Signal Detector: This bit is set when the line interface indicates that the data communication equipment is receiving a carrier signal. The scanner resets this bit to zero when the signal becomes inactive; no interrupt is raised to the CCU, and the SCF is not transferred to CCU storage. The control program may read the bit by issuing a Halt command to terminate the outstanding Write ICW.

Bit 5 - Start Bit Detected: This bit is set to 1 when the start bit for the first character is received after the line has been placed in the receive state (PCF = X'7'). No interrupt is raised to the CCU, and the SCF is not

transferred to CCU storage. The control program may read the bit by issuing a Halt command to terminate the outstanding Write ICW. The bit is reset to zero when the character has been completely assembled; a service request is then set.

Bit 6 - Program Flag: This bit provides a flag in the ICW that can be used by the program.

Bit 7 - Pad Flag: For transmit operations, this bit is turned on by the control program when it requires the scanner to hold the transmit data line in a mark condition for one complete line transmission character time. This operation employs normal transmit character serialization except that the start bit is sent as a mark instead of the normal space. The remainder of the character is serialized as usual. The control program must ensure that the PDF is loaded with X'FF', and the modifiers must specify "Set SCF and PDF". Any number of pad characters may be sent by leaving the pad flag on and the PDF set to X'FF'. When pad transmission is to end, the control program must turn off the pad flag and resume placing normal characters in the PDF.

For receive operations, the bit is turned on by the control program to force the scanner set the "modem check" bit (SCF bit 3) when the "receive line signal detect" is inactive. This use of the pad flag provides a higher level of security on switched lines than can be obtained by monitoring only "data set ready". This method should only be used on lines with duplex facilities.

Primary Control Field (PCF): This field comprises bits 4 through 7 of the LCD/PCF field, and is used during normal transmit and receive operations to define the state of the interface at any particular time. The interpretation of the PCF depends on the LCD field. The four bits are decoded as one hexadecimal digit having the following meaning:

Hex	Meaning
0	No-op
1	(not used)
2	(not used)
3	(not used)
4	(not used)
5	(not used)
6	(not used)
7	Receive
8	Transmit initial
9	Transmit data
A	Transmit break
B	Prepare to turn
C	Transmit turnaround - request to send off
D	Transmit turnaround - request to send on
E	(not used)
F	(not used)

The meaning of each of the different PCF states is described below.

PCF State X'0' - No-op: This state causes the scanner to take no action, either active or passive, on subsequent scans for this particular line. The no-op state is set by the control program; no interrupt is generated.

PCF State X'7' - Receive: In this state, the scanner monitors for start bits according to the type of line operation as defined by the setting of the LCD. This state is set by the scanner after the completion of a transmit turnaround (PCF states X'C' or X'D'), and remains in effect until changed by the control program.

PCF State X'8' - Transmit Initial: This state is set by the control program. Transmit initial sets the interface hardware to the transmit state.

To start a transmit initial sequence, the control program must set the following conditions in the parameter area **before** issuing the Write ICW command:

1. Set modifier bits 0 (Set SCF/PDF), 1 (Set SDF), and 2 (Set PCF).

2. Set the SCF.

3. Store the first character to be transmitted (normally X'FF') in the SDF.

4. Store the second character to be transmitted in the PDF.

5. Set PCF state X'8' (transmit initial).

When the "clear to send" lead from the modem rises, the PCF state changes to X'9' (transmit normal), and transmission of the SDF character is started.

PCF State X'9' - Transmit Normal: This PCF state is set by the scanner after completion of PCF state X'8' (transmit initial); it is used for transmitting data. The first character in the SDF (X'FF') is transmitted twice automatically. The scanner transmits it with the start bit forced to mark. When the SDF has been transmitted for the second time, the character in the PDF (the first data character) is transferred to the SDF and transmitted in its turn. The service request bit (SCF bit 1) is then set, and a level 2 interrupt occurs.

The character in the SDF is always interpreted as a full character, right justified, with as many data bits as defined by the LCD. The scanner automatically supplies the start and stop bits. The stop bit(s) is at the mark level; the start bit is at mark level for the SDF and at space level for the PDF. The scanner stays in the transmit normal state until one of the transmit turnaround states (PCF state X'B', X'C', or X'D') is set by the control program. The scanner detects and signals underruns, but the control program must take corrective action.

After all information characters have been transmitted in the transmit data state (PCF X'9'), the control program must complete the transmit operation by setting one of the transmit turnaround states (PCF states X'B', X'C', or X'D').

Notes:

1. All control and non-information characters must be supplied by the control program; this is because the scanner does not perform character encoding, decoding, or insertion of any kind during a transmit operation.

2. It may be desirable in certain applications (contention) to test the PCF state in order to determine if a transmit operation should be started. For example, a line may have just set PCF state X'7' (receive), and its subsequent interrupt has not been handled by the control program.

PCF State X'A' - Transmit Break: This state is set by the control program instead of PCF state X'9' ("transmit data") when transmitting a break signal to the remote location. The stop bits for the character (X'00') are not transmitted as a mark so that the break signal is continuous spacing.

Note to PCF states X'9' and X'A': After all information characters have been transmitted using PCF state X'9' or X'A', the control program must complete the transmit operation by setting one of the transmit turnaround states, X'B', X'C', or X'D'.

PCF State X'B' - Prepare to Turn:

The Write ICW issued by the control program must specify:

1. Set modifier bits 0 (Set SCF/PDF) and 2 (Set PCF).

2. Set the SCF, PDF, and PCF state X'B' (prepare to turn).

It is not mandatory to set the PCF in the same Write ICW command which sets the SCF/PDF. The PCF may be set in the next Write ICW command.

While bits are being transmitted, this state is identical to PCF state X'9' ("transmit data"). When the character has been completely transmitted, PCF state X'C' (transmit turnaround - request to send off) is set by the scanner (unpredictable results may occur if PCF state X'C' is set by the control program). The SDF is set to X'00', and the interface is left in the "mark" state. This action delays the completion of the transmit operation to ensure that the stop bit remains on the interface transmit data output for at least one bit time before "request to send" can be turned off. At the next bit interval, if "clear to send" is off, the line is placed in an interrupt pending condition as the final interrupt of the transmit operation. PCF state X'7' ("receive") is set by the scanner, and the SDF is left at X'00'. If "clear to send" is on, there is no change in the PCF state and no interrupt is generated until "clear to send" drops.

PCF State X'C' - Transmit Turnaround - Request to Send Off: This state is entered immediately after PCF state X'B' ("prepare to turn"). When the last character has been completely transmitted, PCF state X'C' is set by the scanner. The SDF is set to X'00', and the interface is left in the "mark" state. This action delays the completion of the transmit operation to ensure that the stop bit remains on the interface transmit data output for at least one bit time before "request to send" can be turned off. At the next bit interval, if "clear to send" is off, the line is placed in an interrupt pending condition as the final interrupt of the transmit operation. PCF state X'7' ("receive") is set by the scanner, and the SDF is left at X'00'. If "clear to send" is on, the scanner loops, and there is no change in the PCF state and no interrupt is generated until "clear to send" drops.

When the control program wants to close a line that normally transmits with "request to send" on, it must inform the scanner that "request to send" is to be turned off by PCF state X'C'. This must be done by sending a pad message using PCF state X'B' ("prepare to turn") instead of PCF state X'D' ("transmit turnaround with request to send on"). The pad message should result in a continuous mark condition on the line.

Programming Note: Some modems do not turn off "clear to send" under the above conditions. The control program must therefore test this condition and it may be necessary to set the PCF state to X'D' and operate with "request to send" on.

PCF State X'D' - Transmit Turnaround/Request to Send On: This state is set by the control program.

The Write ICW issued by the control program must specify:

1. Set modifier bits 0 (Set SCF/PDF) and 2 (Set PCF).

2. Set the SDF, the PDF, and PCF state X'D' (transmit turnaround - request to send on).

It is not mandatory to set the PCF in the same Write ICW command which sets the SCF/PDF. The PCF may be set in the next Write ICW command.

While bits are being transmitted, this state is identical to PCF state X'9' ("transmit data").

When the character has been completely serialized, the interface transmit control (not including "request to send") is reset and the final interrupt request for the transmit operation is set. The PCF is set by the scanner to state X'7' (receive).

Note: In wrap mode only, the PCF is set to X'0' (No-Op) at this point.

The state "transmit turnaround with request to send on" must be used with all DCEs that provide duplex facilities and for IBM line adapter/modem equipment on duplex communication facilities.

Note: A start/stop local attachment is considered to be equivalent to a 4-wire point-to-point communication facility.

BSC Operation

There are two different types of BSC operation, defined by the LCD as follows:

		PDF bit positions							
Type of control	LCD	0	1	2	3	4	5	6	7
BSC EBCDIC	'C'	X8	X7	X6	X5	X4	X3	X2	X1
BSC ASCII	'D'	X8	X7	X6	X5	X4	X3	X2	X1

Note: The bit marked X1 is always the first to be transmitted.

Secondary Control Field (SCF): This field is used as a sense, status, and operation modifier field between the control control program and the scanner. The bits of the SCF have the following meaning:

Bit	Meaning
0	(not used)
1	Service request interlock
2	Character overrun/underrun
3	Modem check
4	Receive line signal detector
5	Phase detection
6	Program flag
7	(not used)

Bit 1 - Service Request Interlock: This bit is set if the scanner detects that data transfer or control servicing is required between the control program and the "parallel data field". A Level 2 interrupt request is set. The control program must reset this bit via a Write ICW command after the interrupt has been accepted. If this bit is already set when the scanner is prepared to set it on, and the PCF state is X'7' through X'A', the character overrun/underrun flag (SCF bit 2) is set.

Bit 2 - Character Overrun/Underrun:

- Overrun occurs in the receive state. The bit is set by the scanner if three characters have been received and there is no outstanding Write ICW command. This error is normally caused by an instantaneous peak overload situation. Errors of this type should not normally occur in the average installation, and only infrequently in high throughput installations. When a character overrun occurs, the characters that follow are lost.

- Underrun occurs in the transmit state. SYN characters are transmitted until the control program resumes transmission and changes the PDF field to another character, or until the primary control field is changed from the transmit state.

When this bit is set, the service request interlock bit (SCF bit 1) is set to zero, and a CCU level 2 interrupt request is set.

Bit 3 - Modem Check: This bit indicates that one of the following conditions has occurred:

1. The "data set ready" line is inactive when the PCF field is in states X'5', X'7', X'8', X'9', X'A', X'C', or X'D'.

2. The "clear to send" line is inactive when the PCF field is in states X'9', X'A', or X'D'.

If the "modem check" bit is on, the "service request interlock" (SCF bit 1) is set to zero, and a Level 2 interrupt request is set.

Bit 4 - Received Line Signal Detector: This bit is set when the line interface indicates that the data communication equipment is receiving a carrier signal. The scanner resets this bit to zero when the signal becomes inactive; no interrupt is raised to the CCU, but the SCF is transferred into CCU storage. The control program may read the bit by issuing a Halt command to terminate the outstanding Write ICW.

Bit 5 - Phase Detection: This bit is set to one when the scanner detects the 16-bit SYN character in the received data. No interrupt request is raised to the CCU, and the SCF is not transferred to the PSA status area. The control program may read the bit by issuing a Halt command to terminate the outstanding Write ICW.

Bit 6 - Program Flag: This bit provides a flag in the ICW that can be used by the program.

Primary Control Field (PCF): This field comprises bits 4 through 7 of the LCD/PCF field, and is used during normal transmit and receive operations to define the state of the interface at any particular time. The interpretation of the PCF depends on the LCD field.
The four bits are decoded as one hexadecimal digit having the following meaning:

Hex	Meaning
0	No-op
1	(not used)
2	(not used)
3	(not used)
4	Monitor phase - data set ready check off
5	Monitor phase - data set ready check on
6	(not used)
7	Receive in phase
8	Transmit initial
9	Transmit data
A	Transmit data with new sync
B	(not used)
C	Transmit turnaround - request to send off
D	Transmit turnaround - request to send on
E	(not used)
F	(not used)

The meaning of each of the different PCF states is described below.

PCF State X'0' - No-op: This state causes the scanner to take no action, either active or passive, on subsequent scans for this particular line. The no-op state is set by the control program; no interrupt is generated.

PCF State X'4' - Monitor Phase with Data Set Ready Check Off: This PCF state places a BSC line into a search for phase condition (looking for the bit configuration of the SYN-SYN characters). If the SYN configuration is found, PCF state X'7' (receive) is set and the phase detection bit is set on in the SCF. A level 2 interrupt request, however, is **not** generated at this time. When the next character is received, the character is moved into the PDF, a service request is set in the SCF, a level 2 interrupt is set, and the scanner changes to PCF state X'7' (receive).

PCF state X'4' initializes the first receive operation after a switched network call connection has been established. The inactive state of "data set ready" does not signal a check condition.

PCF State X'5' - Monitor Phase with Data Set Ready Check On: This PCF state is identical to PCF state X'4' except that the inactive state of "data set ready" signals a check condition.

PCF state X'5' can also be set by the scanner after completing a transmit turnaround (PCF state X'C' or X'D').

PCF State X'7' - Receive: In this state, the scanner assembles successive 8-bit characters according to the type of line control as defined by the setting of the LCD (X'C' = BSC EBCDIC line control; X'D' = BSC USASCII line control). This state is set by the scanner on the first character received after synchronization (SYN-SYN characters detected) when in either of PCF states X'4' or X'5' ("monitor phase with data set ready check on/off"). This state remains in effect until changed by the control program.

PCF State X'8' - Transmit Initial: This state is set by the control program. Transmit initial sets the interface hardware to the transmit state. To start a transmit initial sequence, the control program must set the following conditions in the parameter area **before** issuing the Write ICW command:

1. Set modifier bits 0 (Set SCF/PDF), 1 (Set SDF), and 2 (Set PCF).
2. Set the SCF.
3. Store the first character to be transmitted (normally an X'55' pad character) in the SDF.
4. Store the second character to be transmitted (normally a SYN character) in the PDF.
5. Set PCF state X'8' (transmit initial).

When the "clear to send" lead from the modem rises, the PCF state changes to X'9' (transmit normal), and transmission of the SDF character is started.

PCF State X'9' - Transmit Normal: This PCF state is set by the scanner after completion of PCF state X'8' (transmit initial); it is used for transmitting data. The first character in the SDF (normally an X'55' pad character) is transmitted twice automatically. When the SDF has been transmitted for the second time, the character in the PDF (normally a SYN character) is transferred to the SDF and transmitted in its turn. The service request bit (SCF bit 1) is then set, and a level 2 interrupt occurs. The next character to be transmitted should normally be a second SYN character.

The scanner stays in the transmit normal state until one of the transmit turnaround states (PCF state X'B', X'C', or X'D') is set by the control program. The scanner detects and signals underruns, but the control program must take corrective action (for example, by transmitting a BSC end sequence).

After all information characters (EOB, EOT, ENQ, ACK, check characters, etc.) have been transmitted under one of the transmit data states (PCF states X'9' or X'A'), the control program must complete the transmit operation by setting one of the transmit turnaround states (PCF states X'B', X'C', or X'D').

Notes:

1. The sequence PAD-PAD-SYN-SYN described above may be changed as required to suit the application.
2. All control and non-information characters must be supplied by the control program; this is because the scanner does not perform character encoding, decoding, or insertion of any kind during a transmit operation.
3. It may be desirable in certain applications (contention) to test the PCF state in order to determine if a transmit operation should be started. For example, a line may have just set PCF state X'7' (receive), and its subsequent interrupt has not been handled by the control program.

4. For synchronous modem equipment containing a "new sync" lead, PCF state X'A' (transmit data with new sync) should be used instead of X'9'.

PCF State X'A' - Transmit Data with New Sync: This state is identical to PCF state X'9' ("transmit normal"), except that the "new sync" line to the modem is active. It must be used only on 4-wire duplex, multipoint leased-line modems where the associated interface is designated as a master station. The control program must change the PCF state from X'A' to X'9' ("transmit normal") in the character service routine that places the last character to be transmitted in the PDF.

PCF State X'C' - Transmit Turnaround - Request to Send Off: This state is set by the control program.

The control program must set the following conditions in the parameter area before issuing the Write ICW command:

1. Set modifier bits 0 (Set SCF/PDF) and 2 (Set PCF).

2. Set the SCF and PDF in the parameter area.

3. Set PCF state X'C' (transmit turnaround - request to send off).

It is not mandatory to set the PCF in the same Write ICW command which sets the SCF/PDF. The PCF may be set in the next Write ICW command.

While bits are being transmitted, this state is identical to PCF state X'9' ("transmit data"). When the character has been completely transmitted, "request to send" is reset in the interface hardware. This state is not changed until "clear to send" drops. After "clear to send" goes off, the scanner sets PCF state X'5' ("monitor phase - data set ready check on"), and the line is placed in the interrupt pending state. The control program must ensure that "clear to send" is off before the background time-out elapses.

When the control program wishes to close a line that normally transmits with "request to send" on, it must inform the scanner that "request to send" is to be turned off by a PCF state X'C'. This must be done by sending a pad message using PCF state X'C' instead of PCF state X'D' ("transmit turnaround with request to send on"). The pad message should result in a continuous marking condition on the line. An alternative is to ensure that final outgoing transmissions use PCF state X'C'.

Programming Note: Some modems do not turn off "clear to send" under the above conditions. The control program must therefore test this condition and it may be necessary to set the PCF state to X'D' and operate with "request to send" on.

PCF State X'D' - Transmit Turnaround - Request to Send On: This state is set by the control program.

The control program must set the following conditions in the parameter area before issuing the Write ICW command:

1. Set modifier bits 0 (Set SCF/PDF) and 2 (Set PCF).

2. Set the SDF and PDF in the parameter area.

3. Set PCF state X'D' (transmit turnaround - request to send on).

It is not mandatory to set the PCF in the same Write ICW command which sets the SCF/PDF. The PCF may be set in the next Write ICW command.

While bits are being transmitted, this state is identical to PCF state X'9' ("transmit data").

When the character has been completely serialized, the interface transmit control (not including "request to send") is reset and the final interrupt request for the transmit operation is set. The PCF is set by the scanner to state X'5' (monitor phase - data set ready check on).

Note: In wrap mode only, the PCF is set to X'0' (No-Op) at this point.

The state "transmit turnaround with request to send on" must be used on point-to-point 4-wire duplex and on multipoint 4-wire duplex communication facilities where the controller serves as the master station. All BSC switched network communication facilities are half duplex.

Note: A BSC local attachment is considered to be equivalent to a 4-wire point-to-point communication facility.

Ending Status

Word 1	SCF	PDF	-	LCS
Word 2	LCD/PCF	SDF	Modem-In	Modem-Out
Word 3	-	-	-	-

The ending status consists essentially of the current state of the SCF, PDF, LCD/PCF, and the LCF. In addition, a line communication status (LCS) field is used to indicate error conditions, and two further fields, Modem-In and Modem-Out indicate the current state of the control lines from/to the modem. The tables below show the contents of these fields.

Line Communication Status (LCS) Field

LCS	Meaning
C0	AIO error
C2	Adapter check
C4	Scanner error
C6	Scanner failed to answer
C8	Scanner internal error
CA	LIC driver check or internal (BM) clock error
CC	Line interface coupler error
CE	Line interface coupler/internal (BM) clock error
D2	Command rejected
D6	Scanner error reporting patch check
D8	Invalid Level 2 interrupt
E2	CTS dropped/modem retrain
EE	DSR dropped

Modem-In Field

Bit	Meaning
0	Data set ready (DSR)
1	Clear to send (CTS)
2	Ring indicator (RI)
3	Receive line signal detector (RLSD)
4	Test indicator (TI)
5	Receive data (RVDT)
6	(not used)
7	(not used)

Modem-Out Field

Bit	Meaning
0	Data terminal ready (DTR)
1	Request to send (RTS)
2	New sync
3	Data rate select
4	Modem test
5	(not used)
6	(not used)
7	(not used)

Special Considerations

1. On duplex lines, the command is rejected with an LCS = D2 if the PCF value specified in the command is incorrect for the interface on which the command was issued. For example: PCF state X'8' (transmit initial) is set for the receive interface, or state X'D' (Transmit Turnaround) is set for the transmit interface.

2. A Write ICW command remains outstanding until a condition occurs that raises a CCU level 2 interrupt, or until a Halt or Halt Immediate command is received.

3. To reactivate a Write ICW (for example, after a Halt command in order to examine the status), the control program must issue a new Write ICW command with modifier bit 6 (Set SCF) set to one, and with an SCF taken from the SCF in the status area when the previous Write ICW was halted.

4. Transmission is normally started using PCF X'8' (Transmit Initial). However, in the case of a duplex facility, transmission may be started via PCF X'9' (Transmit Data). In this case, the SDF and PDF must contain the first two characters to be transmitted.

5. While a PCF X'7' (Receive) is active, the control program may switch the interface to transmit by issuing a new Write ICW with PCF X'8' (Transmit Initial). Any received data is discarded, but modem check and internal hardware error are retained.

6. The line status after an internal hardware error has occurred is PCF X'0' (No-Op). The scanner takes no action regarding the modem.

7. The line state after a modem check depends on the type of check:

   - NCP and CTS dropped: the modem retrain procedure is entered.
   - NCP and DTR dropped: the line is set to PCF X'0' (No-Op).
   - EP with "Secure Line" in the Set Mode parameters: DTR and RTS are turned off.
   - EP without "Secure Line" in the Set Mode parameters: the line is set to PCF X'0' (No-Op).

Start/Stop Transfer Command (X'41')

The Start/Stop Transfer command is used to transfer "bursts" of 4 bytes between the CCU and the scanner by cycle stealing. Short bursts (less than 4 characters) may be transmitted by using the PCF = X'E' or PCF = X'F' option. In addition, the receive end condition is detected in the scanner by comparing each received character with a set of up to 8 possible ending characters. It is the responsibility of the control program to supply the list of ending characters to the scanner via the Set Mode or Change commands. See "End of Reception Detection" below for details.

Note: Character translation, shift insertion/deletion, and VRC/LRC checking must be done by the control program.

Parameter Zone

Word 1	TCC	Modifiers	SCF	PDF
Word 2	LCD/PCF	SDF	Quiet count	SCF Ext/Ch Ct
Word 3	PDF 1	PDF 2	PDF 3	PDF 4
Word 4	EOR 5*	EOR 6*	EOR 7*	EOR 8*

* Not set by this command, but by the Set Mode or Change commands.

Modifier Byte: This byte contains command modifier bits that have the following meaning:

Bit 0 - Set SCF and PDF

Note: If the PCF = X'7' (receive), only the SCF is set.

Bit 1 - Set SDF

Bit 2 - Set PCF

Bit 5 - Line Quiet Test: This bit indicates that when a start/stop receive operation ends, a delay must occur to allow the line to stabilize before attempting any other operation on the line. The delay is determined via the "Quiet Count" field of the parameter area. Each unit represents one character delay time. Thus, a count of 3 introduces a delay of three characters, the operation ending at the end of the third character time. During this last character time, the scanner assembles a dummy character using the input from the line, and stores it in the PDF field (if more than 8 bit start/stop transmission is used, the high order bits are stored in the LCD part of the LCD/PCF field in the status area). The PDF/LCD fields then contains an indication of the activity on the line; "all ones" indicates a quiet line.

Bit 6 - Set SCF Only

Secondary Control Field (SCF): This field is used to control the secondary control field that is set into the status area at the end of the operation.

The first 6 bits can only reset the corresponding bit; the remaining two can set or reset the bit.

Bit	Meaning
0	Reset stop bit check/receive break
1	Reset service request interlock
2	Reset character overrun
3	Reset modem check
4	Reset receive line signal detector
5	Reset start bit detected
6	Set/reset program flag
7	Set/reset pad flag

These bits have the following meaning:

Bit 0 - Stop Bit Check: On receive operations (PCF state X'7'), the stop bit is checked after each character is received. If the stop bit is a "space" instead of a "mark", the scanner signals this condition by setting bit 0. The command ends as soon as the stop bit check occurs; the last character in the status PDFs is the one on which the check occurred.

When bit 0 is 1, a Level 2 interrupt occurs.

Bit 0 - Receive Break: A receive break can only occur on transmit operations (PCF state = X'9'). If bit 2 (Perform Receive Break Detection) is on in the SCF extension, the "receive data" line is checked for a space condition (0). If a space condition is detected for at least two character times, bit 0 is set to 1; the command ends immediately, even if there are remaining characters to be transmitted.

When bit 0 is 1, a Level 2 interrupt occurs.

Note: The modem leads remain unchanged. The next command should be a Start Stop Transfer command with line turnaround (PCF = X'B' or X'D'), or Halt Immediate. If Halt Immediate is used, request to send remains on.

Bit 1 - Service Request Interlock: This bit is set if the scanner detects that data transfer or control servicing is required between the control program and the PDF. The bit is also set if the line has been halted by a Halt or Read ICW command (this is the only case in which bits 0 and 1 can be on at the same time). A Level 2 interrupt request is set. The control program must reset this bit via a Write ICW command after the interrupt has been accepted.

If bit 1 is set, bits 2, 3, 4, and 5 are forced to zero.

Bit 2 - Character Overrun: Overrun occurs in the receive state when the line interface buffer in the scanner has been filled, and there is no outstanding Start/Stop Transfer command to transfer the received characters to the control program. This bit is set in the status of the **next** Start/Stop Transfer command. If this bit is on, bit 1 (service request interlock) is off.

Bit 3 - Modem Check: This bit indicates that one of the following conditions has occurred:

1. The data set ready line is inactive when the PCF field is in the transmit state (PCF = X'8', X'9', X'A', X'B', X'E', or X'F'), or in the receive state (PCF = X'7').

2. The clear to send line is inactive when the PCF field is in the transmit state (PCF = X'8', X'9', X'A', X'B', X'E', or X'F').

3. The receive line signal detect line is inactive, the pad flag bit (SCF bit 7) is on, and the PCF is in state X'7' (receive).

If the modem check bit is on, the service request interlock (SCF bit 1) is set to zero, the PCF is set to X'0' (No-Op), and a Level 2 interrupt occurs. In addition, if the line is designated as "Secure", data terminal ready is dropped.

Bit 4 - Received Line Signal Detector: This bit is set when the line interface indicates that the data communication equipment is receiving a carrier signal. It is meaningful if the command has been halted (bits 0 and 1 both on), or if a Read ICW is executed.

Bit 5 - Start Bit Detected: This bit is set to 1 when the start bit for the first character is received and indicates that assembly of a character has started. It is meaningful if the command has been halted (bits 0 and 1 both on).

Bit 6 - Program Flag: This bit provides a flag in the ICW that can be used by the program.

Bit 7 - Pad Flag: On transmit operations, this bit indicates that the first character in the PDFs must be transmitted with the start bit in the mark state. If the SCF Extension bit 1 (Multiple Pad) is on, **all** the PDFs must be transmitted with the start bit in the mark state.

Parallel Data Field (PDF): This field is used on transmit operations only. If the character count is non-zero, this field is a duplicate of the PDF 1 character; if the count is zero, it contains the only field to be transmitted.

Line Control Definer (LCD) Field: This field comprises bits 0 through 3 of the LCD/PCF field, and is used during normal transmit and receive operations to define the type of line control. This field is not set by the Start/Stop Transfer command, but by the Set Mode command; it can be modified by the Change command.

The four bits are decoded as one hexadecimal digit having the following meaning:

Hex	Meaning
0	Start/stop 9/6
1	(not used)
2	Start/stop 8/5
3	(not used)
4	Start/stop 9/7
5	Start/stop 10/7
6	Start/stop 10/8
7	Start/stop 11/8
8	(not used)
9	(not used)
A	(not used)
B	(not used)
C	(not used)
D	(not used)
E	(not used)
F	(not used)

Primary Control Field (PCF): This field comprises bits 4 through 7 of the LCD/PCF field, and is used during normal transmit and receive operations to define the state of the interface at any particular time. The LCD field is always zero. The field is meaningful only if Modifier bit 2 (Set PCF) is on. The four bits are decoded as one hexadecimal digit having the following meaning:

Hex	Meaning
0	No-op
1	(not used)
2	(not used)
3	(not used)
4	(not used)
5	(not used)
6	(not used)
7	Receive
8	Transmit initial
9	Transmit data
A	Transmit break
B	Transmit turnaround, RTS off
C	(not used)
D	Transmit turnaround, RTS on
E	Transmit initial, turnaround, RTS off
F	Transmit initial, turnaround, RTS on
F	(not used)

The meaning of each of the different PCF states is described below.

PCF State X'0' - No-op: This state causes the scanner to take no action on the line; modem-in monitoring is also stopped. The no-op state is set by the control program; no interrupt level 2 is generated. The only way to terminate this state is via a Halt or Halt Immediate command.

Note: A Halt Immediate terminates the command, but no status is cycle stolen to the control program, and a CCU level 2 interrupt is not raised.

PCF State X'7' - Receive: The scanner is set to the receive mode and monitors for start bits. PCF state X'7' is ended when:

- The expected number of characters is received.
- An EOR (end of reception) character is received.

PCF State X'8' - Transmit Initial: Transmit initial sets the interface hardware to the transmit state; Ready To Send is turned on. When the "clear to send" lead from the modem rises, the character in the SDF is transmitted with its start bit at the mark level. The PDFs are then transmitted, the character count field indicating how many characters are to be transmitted.

Note: Unlike the Write ICW command, the SDF character is **not** transmitted twice.

The command ends when all characters have been transmitted by the scanner. The line stays in the transmit state waiting for another Start/Stop Transfer command (modifier bit 0 on to set the SCF/PDF again) to transmit more data, or to turn the line around (PCF = B and modifier bit 2 on to set the PCF).

PCF State X'9' - Transmit Data: This PCF state is set by the control program to transmit the PDF characters on a line which has RTS permanently on. It is the responsibility of the control program to ensure that RTS is already on.

PCF State X'A' - Transmit Break: This state transmits all-zero characters with the stop bit at the space level. The control program must place the characters to be transmitted in the PDFs; the count specifies the number of characters to be transmitted.

PCF State X'B' - Transmit Turnaround, RTS Off: The line is switched from the transmit state to the receive state. When the last character has been transmitted, RTS is turned off. When clear to send drops, the line is set to receive mode, and the command is terminated.

Note: If modifier bit 0 (set SCF/PDF) is on, the PDFs are transmitted before turning the line around.

PCF State X'D' - Transmit Turnaround - RTS On: The line is switched from the transmit state to the receive state. When the last character has been transmitted, RTS is **not** turned off. When clear to send drops, the line is set to receive mode, and the command is terminated.

Note: If modifier bit 0 (set SCF/PDF) is on, the PDFs are transmitted before turning the line around.

PCF State X'E' - Transmit Initial and Turnaround with RTS Off: This state combines PCF states X'8' (Transmit Initial) and X'B' (Transmit Turnaround with RTS Off). This state can be used when the message to be transmitted is less than four characters.

PCF State X'F' - Transmit Initial and Turnaround with RTS On: This state combines PCF states X'8' (Transmit Initial) and X'D' (Transmit Turnaround with RTS On). This state can be used when the message to be transmitted is less than four characters.

Serial Data Field (SDF): This field is used for Transmit Initial (PCF = X'8') only. It contains the first character that must be transmitted after CTS (Clear To Send) rises. The character is transmitted with the start bit at the "Mark" level. The SDF character is not duplicated.

Quiet Count: This field is used in conjunction with modifier bit 5; refer to this bit for the use and meaning of the Quiet Count field. **Secondary Control Field Extension (SCF Ext.):** This field comprises the first four bits of the SCF Extension/Character Count field. The bits have the following meaning:

Bit	Meaning
0	End Of Recption (EOR) checking
1	Multiple pads
2	Perform receive break detection
3	(not used)

End Of Reception Checking: This bit is used on receive only. It indicates that the scanner must perform End Of Reception checking.

Multiple Pads: This bit is used on transmit only. It indicates that all the characters in the PDFs must be sent with their start bit in the mark state. Bit 7 (Pad Flag) of the SCF must also be set.

Perform Receive Break Detection: This bit is used on transmit only. It indicates that the scanner must terminate the command if the receive lead stays at the space state for at least two character times. See also under bit 0 of the SCF.

Character Count: This field comprises the last four bits of the SCF Extension/Character Count field.

Transmit: If this field contains a count of 1 through 4, this is the count of characters in PDFs 1 through 4 in the parameter area to be transmitted. If the count is 0, the only character to be transmitted is in the PDF.

Receive: If this field contains a count of 1 through 4, this is the expected count of characters to be put into PDFs 1 through 4 in the status area. If the count is 0, the only expected character must be put into the PDF.

Parallel Data Fields (PDFs): These fields contain the characters to be transmitted, the count being indicated by the Character Count field. The PDF 1 character is also available in the PDF. When transmission is complete, the transmitted characters are copied into the status area.

End Of Reception Fields: These fields are not set by the Start/Stop Transfer command, but by the Set Mode command; they can be modified by the Change command. If the number of End Of Reception characters exceeds 4, the additional characters are stored in word 4 of the parameter zone. See "End Of Reception Detection" for full details.

Ending Status

Word 1	SCF	PDF	-	LCS
Word 2	LCD/PCF	Char. Count	Modem-In	Modem-Out
Word 3	PDF 1	PDF 2	PDF 3	PDF 4

Status Control Field (SCF): This field contains the current state of the SCF at termination.

Parallel Data Field (PDF): On transmit, the scanner moves the first transmitted character into this field. On receive, this field contains a copy of the first received character (from PDF 1).

Line Communication Status (LCS) Field

LCS	Meaning
82	End of reception encountered
C0	AIO error
C2	Adapter check
C4	Scanner error
C6	Scanner failed to answer
C8	Scanner internal error
CA	LIC driver check or internal (BM) clock error
CC	Line interface coupler error
CE	Line interface coupler/internal (BM) clock error
D2	Command rejected
D6	Scanner error reporting patch check
D8	Invalid Level 2 interrupt
E2	CTS dropped/modem retrain
EE	DSR dropped

Line Control Definer (LCD) Field: This field contains the current state of the LCD at termination.

Primary Control Field (PCF): This field contains the current state of the PCF at termination.

Character Count: On transmit, this field indicates the residual count of untransmitted characters. On receive, it indicates the true count of received characters.

Modem-In Field

Bit	Meaning
0	Data set ready (DSR)
1	Clear to send (CTS)
2	Ring indicator (RI)
3	Receive line signal detector (RLSD)
4	Test indicator (TI)
5	Receive data (RVDT)
6	(not used)
7	(not used)

Modem-Out Field

Bit	Meaning
0	Data terminal ready (DTR)
1	Request to send (RTS)
2	New sync
3	Data rate select
4	Modem test
5	(not used)
6	(not used)
7	(not used)

Parallel Data Fields: On receive, these fields contain the received characters; this first received character (in PDF 1) is also copied into the PDF field.

On transmit, the transmitted characters are copied from the parameter area into the status area as the transmission proceeds.

End Of Reception Detection: Up to eight possible ending characters may be used to signal to the scanner that transmission is ended. If the "EOR Checking" bit is on in the SCF extension (bit 0), the received characters are continually compared with the list of ending characters in the scanner; if a match occurs, the command is ended with LCS = X'82' and the Service Request Interlock is set in the SCF. The received character is available to the control program as the last character placed in the PDFs in the status area. Use the status area character count field to determine the last PDF used.

Note: Bit 0 of the received character is ignored when checking for EOR.

The list of End Of Reception characters is loaded into the scanner at Set Mode time. Byte 7 of the Set Mode data area contains the count of EOR characters, bytes 8 through 11 of the Set Mode data contain the first four EOR characters, and bytes 12 through 15 of the parameter zone contain the remainder. The Change command can be used to change the contents and the count of the EOR characters.

Read ICW Command (X'F2)

The Read ICW command is used to terminate an outstanding Write ICW or Start/Stop transfer command. It obtains the status area and raises a CCU level 2 interrupt. The Read ICW command does not affect the line status.

Parameter Zone

The parameter zone is not used by this command.

Status Zone

Word 1	SCF	X'F2'	SES	LCS
Word 2	LCD/PCF	SDF/CC	Modem-In	Modem-Out
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Halted Command Field: contains the code for the Write ICW command (X'40').

Secondary Status (SES) Field: contains the secondary status. This field is not used.

Line Communication Status (LCS) field: See "Miscellaneous Status Fields" at the end of this chapter for a full description of this field.

Character Count: this contains the character count for a status following a transfer command.

Modem-In and Modem-Out Fields: these fields are described in detail at the end of this chapter under the heading "Modem Control Fields".

LCD/PCF and SDF Fields: The meaning of these fields is described under the heading "Write ICW Command".

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

386X/58XX Modems Test Request Command (X'2B')

The 386X and 58XX families of modems provide improved maintenance facilities to allow better network problem determination. These modems provide the means of collecting information about line quality and modem status using tests issued via the 386X/58XX Modems Test Request command. The parameter and status zones are different for each modem family, but the rest of this description is the same for both. You should use modifier bit 6 to indicate which modem the command is for.

The command is used to request test execution on a specific line. The following tests are available:

- Local modem status report
- Local modem self-test
- Local/remote modem status report
- Remote modem self-test
- Remote DTE interface status report

Note: Local Loop Back and Remote Loop Back tests are not supported.

The tests are handled completely by the 386X/58XX Modems Test Request command. As soon as the response frame(s) is received, the command is ended (two response frames are received in the case of Local/Remote Modem Status test).

The 386X/58XX Modems Test command must be issued to the even interface only; if issued to the odd interface, it is rejected. It is also rejected if the primary modem is already in the test mode.

Parameter Zone (386X)

Word 1	TCC	Modifiers	Offset	Long timeout
Word 2	Byte Count	Receive Buffer Pointer		
Word 3	XA1	XA2/XC1	XC1	XC2
Word 4	XTST	-	-	-

Modifier Byte: This byte contains command modifier bits that have the following meaning:

Bit 2 - Two Byte Address: This bit, if on, indicates that the SDLC address field (XA field) in the parameter area is two bytes long (XA1 and XA2), and that the expected address field in the receive frame is also two bytes long. If the bit is off, it indicates that the SDLC address field is one byte long (XA1 only).

Bit 3 - Two Byte Control: This bit, if on, indicates that the SDLC control field (XC field) in the parameter area is two bytes long (XC1 and XC2), and that the expected control field in the receive frame is also two bytes long. If the bit is off, it indicates that the SDLC control field is one byte long (XC1 only).

Bit 5 - Long Reply Timeout on Modem Operations: This bit must be set for all modem operations except "local modem status report". It causes the timer defined by byte 3 of the parameter zone to be substituted for the 0.5-second timer used for local modem operations.

Bit 6 - 386X/58XX Operation: This bit should be off to indicate a 386X modem. (When it is on it indicates a 58XX modem). Remember to use the correct Parameter Zone for the modem, see above.

Bit 7 - Tailed Modem: This bit, if on, indicates that a "tailed" modem (3863/3864 equipped with an LPDA Tailing card, or similar) is to be tested.

Offset: This is the number of bytes between the receive buffer address contained in word 2 and the start of the data.

Long Timeout: This timeout value is used for modem operations in place of the normal 0.5-second timeout value. Modifier bit 5 (Long Timeout on Modem Operations) must be set to 1.

Byte Count: This is the number of bytes actually available in the buffer provided for storing the received data.

Receive Buffer Pointer: This three-byte field contains the address of the buffer where the resulting test frames returned by the modem are to be stored.

SDLC Transmit Address 1 (XA1): This byte contains the SDLC station address to be used in the frame being transmitted (first byte of a two-byte station address).

SDLC Transmit Address 2 (XA2): This byte contains the second byte of the SDLC station address when a two-byte station address is used.

SDLC Transmit Control 1 (XC1): This byte contains the SDLC control byte to be used in the frame being transmitted (first byte of a two-byte control field).

SDLC Transmit Control 2 (XC2): This byte contains the second byte of the SDLC control field when a two-byte control field is used.

XTST: This is the test frame. Its format is described in the modem documentation.

Parameter Zone (58XX)

Word 1	TCC	Modifiers	Rcv Offset	-
Word 2	Byte Count	Receive Buffer Pointer		
Word 3	Reply Timeout		XMIT Offset	-
Word 4	XMIT Count	Transmit Buffer Address		

Modifier Byte: This byte contains command modifier bits that have the following meaning:

Bits 0-4: These are not used for 58XX.

Bit 5 - Timer: This bit should always be on to indicate that the timer value to be used is the one provided in word 3.

Bit 6 - 386X/58XX Operation: This bit should be on to indicate a 58XX modem. (When it is off it indicates a 386X modem). Remember to use the correct Parameter Zone for the modem, see above.

Bit 7: This bit should always be off for a 58XX modem.

Receive Offset: This must always be zero as no offset is used.

Reply Timeout: The timeout value is given in multiples of 100 milliseconds.

Maximum Receive Count: The maximum length, in bytes, available for response data in the receive buffer.

Receive Buffer Address: The address of the receive buffer to be used for the modem response data frame.

Transmit Buffer Offset: This should always be zero as no offset is used.

Transmit Count: The length, in bytes, of the modem command data passed by NCP in the transmit buffer, from A-field to D-field inclusive.

Transmit Buffer Address: The address of the buffer containing the modem command frame.

Status Zone (386X)

The status area contains the ending status of the first returned frame in bytes 0 through 3. In the case of a Local/Remote Modem Status Report operation, bytes 8 through 11 contain the ending status of the second returned frame.

Word 1	SCF(1)	CCMD(1)	SES(1)	LCS(1)
Word 2	Res. Count	Last Receive Buffer Used		
Word 3	SCF(2)	CCMD(2)	SES(2)	LCS(2)

Status Zone (58XX)

The status area contains the ending status of the first returned frame in bytes 0 through 3.

Word 1	SCF	CCMD	SES	LCS
Word 2	Res. Count	Last Receive Buffer Used		
Word 3	A-Field	N-Field	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'2B'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a

hardware error has occurred; refer to the end of this chapter for full details.

Residual Byte Count: This byte indicates the number of unused bytes in the last receive buffer used.

Last Receive Buffer Used: This three-byte field indicates the last buffer that was used to hold the received data.

A-Field (58XX only): SDLC address in the 58XX response SDLC frame. It should be X'FD'.

N-Field (58XX only): SDLC command in the 58XX response SDLC frame. It should be X'1B'.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

1. The receive buffer always contains the **complete** returned frame(s) with the exception of the leading/trailing flags and the FCS. Each returned frame is preceded by a one-byte field containing the byte count of the frame. The resulting format is as follows:

   Buffer Prefix (eight bytes)
   Buffer Header (eight bytes, if present)
   Byte count of first returned frame (one byte)
   First returned frame
   Byte count of second returned frame (one byte)
   Second returned frame

2. The returned frames have the following format:
   RA1
   RA2/XC1
   RC1
   RC2
   RTST
   I1
   I2
   I3

3. The contents of the XTST/RTST field and of the returned frame(s) is as described in the 386X documentation.

4. If an abnormal ending is detected on the second received frame, the Service Request bit is also set off on the **first** received frame.

Trace Command (X'2C')

The Trace command is used to start the scanner interface trace (SIT). The SIT is a line trace, stored by the scanner into buffers or data areas in CCU storage; it may be called either independently, or along with the control program line trace. The SIT records the following information:

- The IOH/IOHI instruction.
- The parameter area of the PSA issued by the control program.
- The status area of the PSA returned by the scanner.
- The data (if any).
- Checkpoint data (if Checkpoint Trace has been activated from the MOSS), comprising the interface control block (ICB) control and status information. This information is traced at critical entry points in the scanner microcode for each command, and is added to the normal SIT.

The Trace command may be issued to either interface; a separate Trace command may be issued to each interface of a half duplex or duplex line. This command is also used to provide a fresh buffer string to the scanner to accommodate additional trace information. Each scanner limits the number of simultaneous traces on its own lines to four; if this limit is exceeded, the scanner raises a level 2 interrupt to the CCU, and the command is rejected. In addition, only one Trace command at a time may be issued to a given interface. If a second trace is issued to the same interface, it is rejected.

The Trace command uses a special vector table called the Trace Line Vector Table (TLVT). This table is located immediately in front of the Line Vector Table, and contains 32 4-byte (full-word) entries. Each entry contains the address of an associated trace parameter/status area (TPSA).

Note: The default starting address of the TLVT is X'0800' corresponding to the normal LVT at X'0880'. If the LVT is relocated using the Set LVT High/Low instructions, the TLVT address may be calculated by subtracting X'80' from the starting address of the LVT.

During trace operations, the scanner uses a **slot** number specified in the instruction to point to one of the 32 entries of the Trace Line Vector Table, in order to access the associated Trace Parameter/Status Area (TPSA).

Trace Initialization

The first Trace command on a given interface is used to start the trace, and must be issued via a Start Line Initial instruction. Subsequent Trace commands (used to provide fresh buffers) must use the Start Line instruction.

Trace Termination

Trace is normally ended by the Stop Trace command (there is one exception, as described below under **Special Considerations).**

Trace Processing

Trace information is stored into CCU buffer areas. Every IOH or IOHI instruction processed for the interface being traced causes a Trace Record Unit (TRU) to be stored. Each TRU contains the IOH/IOHI, the parameter area of the PSA, the data (if any) as exchanged between the scanner and the line interface, and the status area of the PSA, in that order. If Checkpoint Trace

has been activated from the MOSS, checkpoint data, comprising the interface control block (ICB) control and status information, is also stored.

The scanner moves the above data into the current buffer (or buffer chain) until it is completely filled. TRU recording may be continued by providing a new buffer (or buffer chain) via a second Trace command.

TRU Formats

The TRU field formats are as follows:

IOH/IOHI Field

1. Byte 1 contains the character "I" identifying an IOH/IOHI field.
2. Bytes 2 and 3 contain the byte count (always five).
3. Byte 4 contains an X'00' pad byte.
4. Bytes 5 and 6 contain the first halfword of the IOH/IOHI instruction.
5. Bytes 7 and 8 contain the second halfword of the IOH/IOHI instruction.

Parameter Field

1. Byte 1 contains the character "P" identifying a parameter field.
2. Bytes 2 and 3 contain the data count (equal to 16 for a normal command or nine for a character mode command).
3. Byte 4 contains an X'00' pad byte.
4. Bytes 5 through 20 (normal mode) or 5 through 12 (character mode) contain the parameter area of the PSA.

Data Field

1. Byte 1 contains the character "R" for received data, or "X" for transmitted data.
2. Bytes 2 and 3 contain the data count (depends on the length of the data burst).
3. Byte 4 contains an X'00' pad byte.
4. Bytes 5 through "n" contain the data burst. The data burst is a maximum of eight bytes long, rounded to the next even (halfword) count. The true burst count may be found in the scanner status.
5. The remaining bytes contain the scanner status.

Status Field

1. Byte 1 contains the character "S" identifying a status field.
2. Bytes 2 and 3 contain the data count (equal to 12 for a normal mode command or nine for a character mode command).
3. Byte 4 contains an X'00' pad byte.
4. Bytes 5 through 16 (normal mode) or 5 through 12 (character mode) contain the status area of the PSA.

Checkpoint Data Field

1. Byte 1 contains the character "C" identifying a checkpoint data field.
2. Bytes 2 and 3 contain the byte count (always five).
3. Byte 4 contains an X'00' pad byte.
4. Bytes 5 and 6 contain the scanner microcode checkpoint entry address.
5. Byte 7 contains the ICB status byte.
6. Byte 8 contains the ICB control byte.

Overrun Field

1. Byte 1 contains one of the characters "I", "P", "R", "X", "S", or "C" identifying the type of TRU that the scanner was trying to store when the overrun occurred.
2. Bytes 2 and 3 contain the byte count (always one).
3. Byte 4 contains an X'00' pad byte.

Trace PSA Parameter Zone

Word 1	-	Modifiers	Offset	Timer
Word 2	Byte Count	First Buffer Pointer		
Word 3	Slot Identifier		Data Count	Interface
Word 4	Buffer Prefix	Buffer Size	-	-

Modifier Byte: This byte contains command modifier bits having the following meaning:

Bit 0 - NCP Type Buffer: This bit, if off, indicates that the trace is in an NCP buffer whose address is contained in the "First Buffer Pointer". If this bit is on, the data is not in an NCP buffer, but in a data area whose address is contained in the "First Buffer Pointer" (in this case, the buffer prefix and buffer offset are not used).

Bit 1 - Single Interface (Duplex Line): This bit, if off, indicates that both the even and odd interfaces of the line (as defined by the "Interface" field of the parameter zone) are to be traced into the same NCP buffer or data area.

If on, it indicates that either the even or the odd interface of the line is to be traced (only one interface per buffer).

Offset: This is the number of bytes between the buffer address contained in the first buffer pointer and the start of the trace data. It is only used if modifier bit 0 (NCP Type Buffer) is 0. This offset is only used for the first buffer of a chain; the remaining buffers of this chain are assumed to have a zero offset.

Timer: This is a timeout value. If it expires, the scanner sets a buffer service level 2 interrupt request for buffer service. The basic unit is 0.1 second. If the timer field contains all zeros, the timer does not run.

Byte Count: This specifies the effective size of the data area of the first buffer of a chain (if modifier bit 0 = 0), or of the data area (if modifier bit 0 = 1).

First Buffer Pointer: This three-byte field contains the address where the trace data is to be stored. If modifier bit 0 = 0, it indicates the address of the first NCP type buffer of a chain; if modifier bit 0 = 1, it indicates the address of the data area itself.

Slot Identifier: This two-byte field contains the identifier provided by the scanner to the CCU via the Get Line Identification instruction.

Data Count: This is the number of bytes of data to be traced for the interface. It is reinitialized every time that a turnaround occurs on the line, or a new SDLC frame is transmitted or received. The data includes scanner control information, but the count indicates only the number of true data characters. The scanner rounds the data count to the next half-word boundary.

- If the data count is X'00', no data is traced.
- If the data count is X'FF', all data is traced.
- If the data count is X'nn', nn bytes of data are traced.

The data count does not apply to the character mode as all the data is automatically included in the scan interface trace since the PDF field is part of the parameter/status area.

Interface: This byte contains the interface address of the line to be traced. If modifier bit 1 is off, the interface address must be even.

Buffer Prefix: This field is only used if modifier bit 0 (NCP Type Buffer) is zero. It specifies the size of the prefix area in the NCP buffer, and contains the link pointer to the next buffer in the chain, the offset, and the data count.

Buffer Size: This field is only used if modifier bit 0 (NCP Type Buffer) is zero. It specifies the effective data area size within all the buffers of an NCP buffer chain except the first. The effective data area size of the first buffer of the chain is specified in the Byte Count field (buffer length minus prefix size).

Trace PSA Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	Res. Count	Pointer to Last Buffer Used		
Word 3	-	Res. Timer	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'2C'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: This field contains an indication of hardware errors.

Residual Byte Count: This byte indicates the number of unused bytes remaining in the last buffer used.

Pointer to Last Buffer Used: This three-byte field indicates the last buffer that was used to hold trace information.

Residual Timer: if the current interrupt is a request for buffer service due to a timeout, this field contains zero. If the current interrupt is not due to a timeout, this field contains the current value of the timer.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

1. A TRU may be split between two buffers of the same chain, or between two buffers of different chains. In the second case, a new Trace command is required for the second chain.

2. The trace operation is not stopped if the data line is disabled, or if an error condition is detected. The only way to turn off the trace is via a Stop Trace command. The only exception to this rule is if an internal hardware error is detected during the status transfer of the Trace command; however, no level 2 interrupt occurs.

3. If an internal hardware error occurs during the transfer of the trace data from the scanner to the CCU buffer(s), the Trace command is terminated, but the trace **function** remains active. Data transfer is resumed as soon as the scanner receives a new Trace command.

4. If an overrun occurs, data recording inside the scanner is stopped. However, the trace function remains active, and data recording is restarted as soon as the overrun condition disappears. The TRUs lost because of the overrun are replaced by overrun TRUs, with the first byte indicating the type of TRU on which the overrun occurred, and with a byte count of one.

5. For an ending condition with X'D2' or X'D4' in the LCS, the slot identifier returned by the Get Line Identification instruction is the identifier provided in the parameter zone of the Trace PSA.

6. The scanner cycle steals in bursts of 64 bytes as the data is accumulated. If the scanner timeout expires, any remaining data bytes are flushed out by cycle stealing into the CCU buffer; this may mean that a burst of less than 64 bytes has occurred.

Stop Trace Command (X'2D')

The Stop Trace command is used to stop the scanner internal trace (SIT) on a specific interface (even or odd) or on both interfaces of a line.

Parameter Zone

The parameter zone is not used by the Stop Trace command.

TPSA Status Zone

Word 1	SCF	CCMD	SES	LCS
Word 2	-	-	-	-
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'2D'.

Secondary Status (SES) Field: Contains the secondary status.

Line Communication Status (LCS) field: This field contains an indication of hardware errors.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

1. If an internal hardware error is detected during the status transfer of the Stop Trace command, the trace function is stopped; however, no level 2 interrupt occurs.

2. The command is rejected in the normal way if no SIT was active on this interface.

   The command is rejected with command reject, error status type 3, and a level 1 interrupt request if no SIT was active for the specified slot, or if an outstanding trace command was already active for the slot.

Wrap Command (X'2E')

The Wrap command is used to turn on the wrap function. It is always accepted if issued to the even interface, providing that a Set Mode command has been previously issued to the same even interface. If the Wrap command is issued to the odd interface, or if a Set Mode command has not been previously issued, it is rejected. Several different wrap tests may be performed, depending on the setting of the modifier bits.

The diagnostic wrap mode remains in effect until the next Reset-N or Reset-D command for that line; the line then returns to normal operation.

Parameter Zone (Normal Mode or Control Lead Wrap)

Word 1	TCC	Modifiers	Offset T*	Offset R*
Word 2	Tx Count*	First Transmit Buffer (Modem-Out)*		
Word 3	-	-	Receive Interface ID**	
Word 4	Rx Count*	First Receive Buffer (Modem-In)*		

* Control lead wrap only
** Data wrap only

Parameter Zone (Character Mode)

Word 1	TCC	Modifiers	-	-
Word 2	-	-	-	-
Word 3	-	-	Receive Interface ID	
Word 4	-	-	-	-

Note: For a **control lead** wrap of a Character Mode line, the PSA as defined for normal mode lines must be used.

Modifier Byte: This byte contains command modifier bits that have the following meaning:

Bit 0 - NCP Type Buffer: This bit, if off, indicates that the data to be transferred from the control program is in an NCP type buffer whose address is contained in the "First Transmit Buffer Pointer". If this bit is on, the data is not in an NCP type buffer, but in a data area whose address is contained in the "First Transmit Buffer Pointer" (in this case, the buffer prefix and buffer offset are not used).

Bit 1 - NCP Type Buffer: This bit, if off, indicates that the data transferred to the control program is in an NCP type buffer whose address is contained in the "First Receive Buffer Pointer". If this bit is on, the data is not in an NCP type buffer, but in a data area whose address is contained in the "First Receive Buffer Pointer" (in this case, the buffer prefix and buffer offset are not used).

Bit 4 - Data/Control Leads Wrap: This bit, when off, indicates that a data wrap is required; if on, it indicates a control lead wrap.

Bit 5 - LIC/External:

3720/3721 only: This bit, when off, indicates that the wrap occurs at the LIC itself.

3725 only: If on, it indicates that the wrap must be set up externally via a special plug on the cable, or at the modem via the modem switches.

Bit 6 - Cable/Modem Wrap (3725 only): This bit is used when modifier bit 5 (LIC/External Wrap) is set to 1 to indicate an external wrap. This bit, when off, indicates that the wrap must be set up via the special plug on the modem cable. When on, it indicates that the wrap must be set up using the switches on the modem.

Note: If the bit is on, and external clocking was specified in the Set Mode command, the clocking comes from the modem. If the bit is off, a 480 Hz clock is used.

Transmit Count: This field applies to a Control Lead wrap only. It contains the transmit byte count.

First Transmit Buffer (Modem-Out): This field applies to a Control Lead wrap only. It contains the address of the buffer area that contains the sequence of Modem-Out test patterns.

Receive Interface Identification: This field is an LVT address containing the PSA address of the receive interface.

Receive Count: This field applies to a Control Lead wrap only. It contains the receive byte count.

First Receive Buffer (Modem-In): This field applies to a Control Lead wrap only. It contains the address of the buffer area in which the sequence of Modem-In test patterns are to be stored.

Status Zone (Normal Mode)

Word 1	SCF	CCMD	SES	LCS
Word 2	Res. Count	Pointer to Last Receive Buffer Used		
Word 3	-	-	-	-

Status Zone (Character Mode)

Word 1	SCF	CCMD	-	LCS
Word 2	LCD/PCF	SDF	Modem-In	Modem-Out
Word 3	-	-	-	-

Status Control Field (SCF): This byte contains information that describes the progress of the command.

Current Command (CCMD) Field: Contains the current command, in this case, X'2E'.

Secondary Status (SES) Field: Contains the secondary status. This field is always X'00' for this command in normal mode; in character mode, it is not used.

Line Communication Status (LCS) field: If this field contains X'D2', the command was rejected. If it contains any other combination of bits, a hardware error has occurred; refer to the end of this chapter for full details.

Residual Byte Count: This byte indicates the number of unused bytes remaining in the last buffer used. It is only used for normal mode and for a control lead wrap in character mode.

Pointer to Last Receive Buffer Used: This three-byte field indicates the last buffer that was used to hold the last Modem-In pattern received. It is only used for normal mode and for a control lead wrap in character mode.

Other Fields: the remaining fields of the status area are not used for the Wrap command in normal mode. In character mode, two additional fields are used. They are LCD/PCF and SDF. The full meaning of these fields is described at the end of this chapter.

Ending Status

The ending status is contained in the SCF, the SES, and the LCS fields. See "Miscellaneous Status Fields" at the end of this chapter for a full description of these fields.

Special Considerations

1. In the case of an external wrap, the wrap must be set up physically (via the special cable plug or the modem switches) **before** the Wrap command is issued.

2. For Japanese PTT, the external wrap test must be set up at the cable level for V24 type interfaces, and at the modem level for V35 type interfaces (Test 1 switch).

3. For data wrap, if a new PSA is to be used, or if the line was operating in half duplex mode, the Wrap command must be issued via a Start Line Initial instruction. The PSA must contain the identification of the receive interface. When in wrap mode, the line is handled as a duplex line; the control program may then issue commands on both interfaces.

Modem Control Fields

Modem-In Field

This byte contains the status of the incoming leads from the modem.

1. EIA/CCITT V24 (LIC 1), Bell 303 (LIC 2), and CCITT V35 (LIC 3).

	Bit	Meaning
	0	Data set ready (DSR)
	1	Clear to send (CTS)
	2	Ring indicator (RI)
	3	Receive line signal detector (RLSD)
	4	Test indicator (TI)
	5	Received data (RD)
	6	(not used)
	7	(not used)

2. EIA/CCITT V25 (LIC 1)

	Bit	Meaning
	0	Power indicator (PWI)
	1	Data line occupied (DLO)
	2	Present next digit (PND)
	3	Abandon call and retry (ACR)
	4	Call originator status (COS)
	5	(not used)
	6	(not used)
	7	(not used)

3. X.21 (LIC 4)

	Bit	Meaning
	0	(not used)
	1	Indication (I)
	2	(not used)
	3	(not used)
	4	(not used)
	5	Receive data (R)
	6	(not used)
	7	(not used)

Modem-Out Field

This field contains the status of the leads going to the modem.

1. EIA/CCITT V24 (LIC 1), Bell 303 (LIC 2), and CCITT V35 (LIC 3).

	Bit	Meaning
	0	Data terminal ready (DTR) (0 = LIC wrap)
	1	Request to send (RTS)
	2	New sync (NSYNC)
	3	Data rate select
	4	Modem test
	5	(not used)
	6	(not used)
	7	(not used)

2. EIA/CCITT V25 (LIC 1)

	Bit	Meaning
	0	Digit signal 8
	1	Digit signal 4
	2	Digit signal 2
	3	Digit signal 1
	4	Call request (CRQ)
	5	Digit present (DPR)
	6	(not used)
	7	(not used)

3. X.21 (LIC 4)

	Bit	Meaning
	0	No LIC wrap
	1	Control (C)
	2	(not used)
	3	T.EN (= T.Enable)
	4	(not used)
	5	(not used)
	6	(not used)
	7	(not used)

Miscellaneous Status Fields

Three status fields, the Status Control Field (SCF), the Secondary Status Field (SES), and the Line Communication Status Field (LCS), have many different meanings, dependant on the type of line control used, so they are grouped here for easy reference.

Status Control Field

This byte contains information describing the progress of the operation being performed. The bits of the status control field have the following meanings:

Bit	Meaning
0	Halt/abort
1	Service request
2	Scanner underrun/overrun
3	Modem check
4	Data stored
5	End of message (EOM)
6	Data transmission occurred
7	Receive sequence

Bit 0 - Halt/Abort: This bit indicates that the command was ended prematurely for one of the following reasons:

1. On SDLC lines only, an abort sequence has been detected during a receive operation. The frame is flushed up to an ending condition (idle), and the Service Request bit (bit 1) is set off.

2. A Halt command has been received from the CCU. The Service Request bit (bit 1) is also set on.

Bit 1 - Service Request: This bit indicates that the command ended normally. Buffer service may or may not be required, depending on the state of the "end of message" (EOM) bit (SCF bit 5): if the EOM bit is off, buffer service is required; if the EOM bit is on, the operation is complete and buffer service is not required.

This bit is also set if a Halt command has been executed. In this case, the Halt bit (bit 0) is also set.

Note: After an abort sequence on an SDLC line, this bit is always off.

Bit 2 - Scanner Underrun/Overrun:

1. An underrun can only occur on SDLC lines during a transmit operation. This situation occurs when no byte is available to transmit because the the cycle steal request for new data has not yet been satisfied. An abort sequence is transmitted.

2. An overrun can only occur during a receive operation. This situation occurs when the data coming from the interface has nowhere to go because the byte buffer associated with that interface is already full. The action taken by the scanner depends on the type of line, as follows:

   SDLC: the frame is flushed up to an ending condition (flag or idle).

NCP BSC: the data is flushed either up to an ending condition (ETB, ETX, ENQ or timeout), or if in ITB mode, until an ITB is received. The overrun bit is set in the EIB; End of Message (EOM) is also set.

EP BSC: if either "EIB Mode" is set or "ITB is Data" is not set, the data is flushed either to an ITB or to an ending condition (ETB, ETX, ENQ or timeout); if "EIB Mode" is set, the overrun bit is set in the EIB; if the data was flushed to an ending condition, EOM is set, but not for ITB.

If "EIB Mode" is not set and "ITB is Data" is set, the data is flushed to an ending condition, EOM is set. The overrun bit is set in the EIB; End of Message (EOM) is also set.

Bit 3 - Modem Check: This bit indicates that a modem check has been detected.

Bit 4 - Data Stored: This bit is valid only for a receive sequence. It indicates that information has been placed in the buffer or data area specified for this command.

Bit 5 - End of Message (EOM): This bit indicates that the operation initiated by the control program is complete. For 270X EP lines working in normal mode, EOM is always set unless the command ends with a buffer request.

Bit 7 - Receive Sequence: This bit indicates that a line turnaround has occurred during execution of the command, and that the information contained in the PSA status area applies to that part of the command that was executed after the turnaround.

Programming Note: If SCF bits 0 through 3 are all zero, the required status information is contained in the Secondary Status Field (SES). If the SES is also zero, then the required status information is in the Line Communication Status (LCS) field.

Secondary Status Field

This byte identifies errors encountered during the execution of the command. When any bit in this byte is on, the "service request" bit (SCF bit 1) must be off, except for "modem retrain" in NCP BSC. The bits of the secondary status field have the following meanings:

Bit	Meaning
0	Modem retrain
1	Idle detection (SDLC)/format exception (NCP BSC, EP BSC)
2	(not used)
3	Data check (SDLC, NCP BSC, EP BSC)
4	Flag off boundary (SDLC)/bad pad (NCP BSC)
5	In phase (EP BSC)/58XX: TI on (NCP)
6	DLE error (NCP BSC)
7	Early flag (SDLC)/length check (NCP BSC)

Bit 0 - Modem Retrain: This bit only applies to NCP operations on SDLC and BSC lines. Modem retrain indicates that a loss of CTS occurred during a transmit operation, but that CTS was recovered before the Enable timeout expired, indicating that the loss was only temporary. If CTS is not recovered before the timeout expires, Modem Retrain is not set; Modem Check is set instead.

Bit 1 - Idle Detect/Format Exception: The meaning of this bit depends on the type of line control:

1. SDLC: The bit signifies "Idle Detect". It indicates that at least 15 consecutive 1 bits have been received. When this bit is on, the "abort" bit (SCF bit 1) must also be on.

   Note: The idle detect bit without an abort bit may occur in the case of an overrun condition (SCF bit 2) and the line is found to be at the "mark" level.

2. NCP BSC: The bit signifies "Format Exception". On a transmit sequence, this bit indicates that the "transmit control" field did not contain a valid final control sequence. The scanner has forced an ENQ character before turning the line around.

   On a receive sequence, this bit indicates that the scanner has received a control character in an invalid sequence. The possible causes are as follows:

   a. SOH was not the first character received.

   b. A control character was received while receiving leading graphics.

   c. The message started with ITB, ETB, ETX, DLE-ITB, DLE-ETB, or DLE-ETX.

   d. A bad pad was received.

3. EP BSC: The bit signifies "Format Exception". It indicates that EOT followed by a good pad (or by a bad pad if the "ignore bad pad" option is o') has been received.

Bit 3 - Data Check: This bit applies to SDLC, NCP BSC, and EP BSC lines on receive only. It indicates that a CRC or LRC/VRC (ASCII) check has been detected.

Bit 4 - Flag Off Boundary/Bad Pad: The meaning of this bit depends on the type of line control:

1. SDLC: The bit signifies "Flag Off Boundary". It indicates that an SDLC flag character has been received, but not on a correct character boundary.

2. NCP BSC: The bit signifies "Bad Pad". It indicates that the character following EOT was not X'xF'.

Bit 5 - In Phase (EP BSC)/58XX: TI on (NCP):

1. In Phase: This is valid only for EP BSC lines.

   On a Transmit Initial command, it indicates that the command was issued to a line that was receiving; the command is rejected.

   On a Receive command, it is set to indicate that the In Phase condition has been detected and a Halt command has been received. It is reset on any condition that causes EOM to be set.

2. 58XX: TI on (NCP): This is valid only for 58XX modems. When on, this indicates that the TI lead was found to be on. When off, this indicates that the TI lead was found to be off.

Bit 6 - DLE Error: This bit is valid only for NCP BSC lines. It indicates that a DLE character was followed by an invalid character.

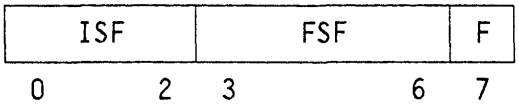
Bit 7 - Early Flag (SDLC)/Length Check (NCP BSC):

SDLC: the received frame was too short, i.e. less than four bytes long.

NCP BSC: While transmitting, an embedded ENQ, ETB, or ETX character was found before the byte count reached zero. This is **not** an error. When this bit is set, the "end of message" bit (SCF bit 5) is also set and the command ends as required for the specific embedded character.

Line Communication Status Byte (LCS)

The line communication status byte contains status information relative to the line being serviced. It contains fields indicating an initial and a final status. It also contains an indicator which specifies whether or not leading graphics were the first characters received. The line communication status byte has the following format:



ISF = Initial status field

FSF = Final status field

F = Leading graphics flag/timeout during X.21 clear

Initial Status Field (ISF)

The initial status field indicates essentially the type of line control that is used. The initial status field is decoded as follows:

NCP BSC Receive only

Bits	
0 1 2	Meaning
0 0 0	Control mode - no text received
0 0 1	Text mode - STX is first character
0 1 0	Transparent text mode - DLE-STX are first characters
0 1 1	Header mode - SOH is first character

Special

Bits	
0 1 2	Meaning
1 0 0	Special status

Errors

Bits	
0 1 2	Meaning
1 1 0	Internal box error
1 1 1	Hardware error

Final Status Field (FSF)

The final status field gives further status information. Its interpretation depends on the ISF, as follows:

1. **Initial Status = 0xx**

FSF	Meaning
0000	Timeout occurred after reception has begun and initial status is not 000
0011	ENQ received
0100	EOT received
0101	DLE/xxx was received (xxx = any valid second character)
0110	Wrong ACK received
0111	NAK received
1001	ETX received
1010	ETB received
1101	RVI received
1110	Positive ACK (0 or 1) received
1111	WACK received

2. **Initial Status = 100 (Special)**

FSF	Meaning
0000	Timeout (nothing received); ACR, COS, DLO, or PND failed to drop; X.21 timeout on ready for data
0010	X.21 timeout during clear
0011	386X test control active/X.21 timeout on proceed to select
0100	DLE-EOT disconnect sequence received
0101	Lost data
0110	Poll entry too long
1100	EOT transmitted
1101	X.21 call progress signal (CPS) error
1110	Disconnected/X.21 DCE clear received
1111	Connected

Notes:

1. When a special status occurs, the line is set to the disable state.

2. For X.21, if "DTE Clear" or "DCE Clear" confirmation is required, the "X.21 Timeout During Clear" flag (bit 7) is added to the final status to indicate the result of the clear operation. Bit 7 is set to zero if the clear was successful, and to one if it was not.

0000 - Timeout: This condition occurs when a reply to a transmission is expected, and nothing is received within the timeout period.

0000 - ACR, COS, DLO, or PND Failed to Drop: This condition occurs on an ACU interface.

0000 - X.21 Timeout on Ready for Data: This condition occurs on an X.21 call request command if timer T2 or T3 (as defined by the CCITT) has timed out.

0010 - X.21 Timeout During Clear: This condition occurs when the DTE has initiated a clear operation, but the R = 0, I = OFF condition has not been received from the DCE within 2 seconds.

0011 - 386X/58XX Test Control Active: This condition occurs if the 386X/58XX modem is already in the test mode (TI lead is active).

0011 - X.21 Timeout on Proceed to Select: This condition occurs if the "Proceed to Select" signal is not received within 3 seconds of the "Call Request" signal being sent.

0100 - DLE-EOT disconnect sequence: This condition occurs when a DLE/EOT (disconnect) sequence has been received.

0101 - Lost Data: This condition occurs when the line interface buffer has been filled with received data, and no receive command has been issued to accept it. This condition is presented as status to the next command issued.

0110 - Poll Entry Too Long: on an EP BSC Poll command, a poll entry was longer than 56 bytes.

1100 - EOT Transmitted: EOT has been transmitted as requested in the transmit control byte.

1101 - X.21 Call Progress Signal (CPS) error: This condition occurs when a non-retriable call progress signal has been received, or if one or more unsuccessful retries have occurred. The last CPS received is available in the status area.

1110 - Disconnected/X.21 DCE Clear Received During Call Request: This is the normal ending status for a "Disable" or "X.21 Clear" command. It also occurs if a DCE Clear indication is received while processing an "X.21 Call Request" command or an SDLC command on an X.21 interface. It is also set when DLE-EOT has been transmitted as requested in the transmit control byte.

1111 - Connected: This is the normal ending status for the "Enable", "Dial", "Monitor Incoming Call", "X.21 Call Request", and "X.21 Monitor Incoming Call" commands.

3. Initial Status = 110 (Internal Box Error)

FSF	Meaning
0000	AIO error
0001	Adapter check
0010	Scanner error
0011	Scanner failed to answer
0100	Scanner internal error
0101	LIC driver check/internal (BM) clock error
0110	LIC internal error
0111	LIC/clock error
1000	No interrupt from scanner/receive text timeout
1001	Command rejected
1010	Trace already active
1011	Scanner error reporting path check
1100	Invalid level 2 interrupt
1101	Modem already in test mode

Notes:

1. For NCP operations, the line is set to the disable state.

2. For NCP operations on duplex lines, the command on the failed interface is ended; the LCS indicates the cause of the error, and a level 2 interrupt is raised to the CCU. The command on the other interface is cleared; no ending status is set, and no interrupt is raised to the CCU.

3. For EP operations, the line is set to the No-Op state.

4. After a hardware error, the only commands accepted for that line are "Enable", "Monitor Incoming Call", or "Dial" (if autocall interface).

0000 - AIO Error: This condition occurs if a hardware error is detected during an adapter initiated operation (cycle steal).

0001 - Adapter Check: This condition occurs if a hardware error is detected in the scanner hardware.

0010 - Scanner Error: This condition occurs if a hardware error is detected at the scanner interconnection.

0011 - Scanner Failed to Answer: This condition occurs if there was no reply to an operation initiated by the scanner microcode.

0100 - Scanner Internal Error: This condition occurs if a hardware error is detected in the scanner hardware.

0101 - LIC Driver Check/Internal (BM) Clock Error: This condition occurs if a hardware error is detected in the modem-out drivers or if a parity error is detected in the interface clock.

0110 - LIC Internal Error: This condition occurs if a parity check occurred in the LIC.

0111 - LIC/Clock Error: This condition occurs if a parity check is detected in the interface clock, or if the LIC is not physically present, for an operation initiated by the scanner microcode.

1000 - No Interrupt From Scanner: This condition applies to SDLC lines only. Once reception has started, an NCP type buffer must be filled within 6 seconds. This error indicates that the scanner has stopped transfer of the received data to the CCU, or that the receive text timer timed out before the current buffer was filled.

1001 - Command Rejected: This condition may occur for three different reasons:

1. The command was issued on the wrong interface of the line (transmit instead of receive, or vice-versa).

2. The command is of the "Transmit" or "Receive" type, and the line had not been previously enabled.

3. The command could not be accepted in the current state of the line.

1010 - Trace Already Active: This condition occurs if a scanner internal trace (SIT) is already running for this line.

1011 - Scanner Error Reporting Path Check: This condition occurs if a the scanner cannot complete an operation initiated by the scanner microcode, because of an error that cannot be reported.

1100 - Invalid Level 2 Interrupt: This condition occurs if the scanner requested an unexpected level 2 interrupt for this interface.

1101 - Modem Already in Test Mode: This condition occurs if the interface is already in the Modem Test mode and a modem test is requested.

4. Initial Status = 111 (Hardware Error)

FSF	Meaning
0001	CTS dropped during command/modem retrain
0011	RLSD failed to drop on Disable command (not used by NCP)
0111	DSR dropped during command
1001	CTS failed to come up/X.21 DCE not ready
1010	DSR failed to come up
1011	No cable installed
1100	DSR and/or CTS failed to drop (on Disable and Transmit Data commands on half duplex lines without duplex facilities
1110	Autocall check

Notes:

1. For NCP operations, the line is set to the disable state.

2. For NCP operations on duplex lines, the command on the failed interface is ended; the LCS indicates the cause of the error, and a level 2 interrupt is raised to the CCU. The command on the other interface is cleared; no ending status is set, and no interrupt is raised to the CCU.

3. For EP operations, the line is set to the No-Op state.

4. After a hardware error, the only commands accepted for that line are "Enable", "Monitor Incoming Call", or "Dial" (if autocall interface).

0001 - CTS dropped during command: This condition occurs if Clear To Send failed during the transmission.

0011 - RLSD failed to drop (on Disable command): This condition occurs if Receive Line Signal Detector failed to drop during a disable command.

0111 - DSR dropped during command: This condition occurs if Data Set Ready failed during the command.

1001 - CTS failed to come up: This condition occurs if Clear To Send failed to rise after Request To Send was set.

1001 - X.21 DCE not ready: During enabling of a leased X.21 line, the DCE was found to be not ready.

1010 - DSR failed to come up: This condition occurs if Data Set Ready failed to rise after Data Terminal Ready was set.

1011 - No cable installed: This condition occurs if there is no cable connected to the modem.

1100 - DSR and/or CTS failed to drop: This condition occurs only for Disable and Transmit Data commands on half duplex lines without duplex facilities. It indicates that Data Set Ready and/or Clear To Send failed to fall after Data Terminal Ready was turned off. **1110 - Autocall check:** This condition indicates that the autocall unit was unable to accept a dial operation because PWI was down when starting to dial, or COS was received before all dial digits were sent. The status of the interface leads is available to the control program in the status area.

Leading Graphics Flag

The leading graphics flag is used only for NCP BSC Receive and NCP BSC Control commands. It indicates that a non-control character was the first character received. For the "Hardware Error" status configuration, the scanner automatically sets the leading graphics flag to zero. For the "Special" status, the leading graphics flag indicates the result of the DTE Clear or DCE Clear; flag off indicates a successful clear, and flag on an unsuccessful clear.

Wrap Testing

Wrap testing loops the data (or modem control leads) from the transmit interface back into the receive interface. The transmitted and received signals are then compared. This allows the detection of errors in the transmission path to the modem.

Wrap testing is for a specified line and is initiated from the MOSS. Before issuing the wrap, the line must be deactivated from the host for leased lines; switched lines must be disabled.

When the MOSS receives a wrap request from the MOSS operator, it raises a Mailbox In Wrap Test Initialize Request to obtain the line characteristics from the control program. The MOSS then uses the data provided by the MOSS operator to create a Mailbox In Start Wrap Request to the control program. If the MOSS operator has not provided any data, the MOSS constructs the request from information in its own storage.

For BSC wrapping (NCP and EP) the scanner hardware cannot handle the SYN insert timer on the transmit interface and the SYN detection timer on the

receive interface at the same time. Therefore, during wrap, only the SYN insert timer on the transmit interface runs.

Wrapping is started by the Wrap command, and stopped by a Reset-N or Reset-D command. The line must first have been initialized correctly by means of a Set Mode command.

The wrap may be set up at the LIC, at the modem cable, or at the modem. Two types of wrap are possible:

- Data wrap. The data is wrapped back from the transmit to the receive interface.
- Control lead wrap. The modem control leads are wrapped back from the transmit to the receive interface.

SDLC Data Wrap

The following sequence of commands should be used:

1. Wrap on the transmit interface, specifying "data" and "LIC" or "external". This also forces the line to the duplex mode.
2. SDLC Receive on the receive interface.
3. SDLC Transmit on the transmit interface. Modifier bits 4 (Compare Address) and 6 (Turn Line Around) must both be off. The XA1, XA2, XC1, and XC2 of the SDLC Transmit parameter area are used by the command; the number of address and control fields must be specified in the normal way using modifier bits 2 and 3.
4. Repeat steps two and three as often as required.
5. End the wrap with a Reset-D or a Reset-N command.

NCP BSC Data Wrap

The following sequence of commands should be used:

1. Wrap on the transmit interface, specifying "data" and "LIC" or "external".
2. NCP BSC Receive on the receive interface.
3. NCP BSC Transmit on the transmit interface. Modifier bits 6 (Turn Line Around) and 7 (Acknowledgment Expected) must both be off.
4. Repeat steps two and three as often as required.
5. End the wrap with a Reset-D or a Reset-N command.

Notes:

1. Line control characters are not provided in the buffer chain.
2. The transmit control byte is the first character provided by the MOSS in the buffer chain. It must not be transmitted, rather it must be passed to the scanner in the parameter zone of the PSA.
3. ITB/EIB mode is not supported when in wrap mode.

EP BSC Data Wrap

The following sequence of commands should be used:

1. Wrap on the transmit interface, specifying "data" and "LIC" or "external".
2. EP BSC Receive on the receive interface.
3. EP BSC Transmit Initial on the transmit interface.
4. EP BSC Transmit Data on the transmit interface. If modifier bit 5 (Data Chain) is set to zero, the scanner does not turn the line around.
5. If there is only one buffer, repeat steps two through four as often as required, then end the wrap with a Reset-D or a Reset-N command. If there is more than one buffer, continue as follows:
6. EP BSC Receive Continue on the receive interface.
7. EP BSC Transmit Data on the transmit interface.
8. Repeat steps 6 and 7 as often as required.
9. End the wrap with a Reset-D or a Reset-N command.

Notes:

1. Line control characters must be provided by the MOSS in the buffer chain to the control program. They are returned to the control program as received by the scanner, with the exception of SYN, DLE-SYN, and inserted DLEs.
2. The transmit control byte is the first character provided by the MOSS in the buffer chain. It must not be transmitted, rather it must be passed to the scanner in the parameter zone of the PSA.
3. If the transmit control byte specifies transparent wrap, the last buffer of the pattern must be sent as second transparent write. Transparent ITB is not supported; non transparent ITB is supported.
4. If no ending character is provided, or if is not recognized, the Flush command must be used to recover from the wrap test.

Character Mode Data Wrap.

The following sequence of commands should be used:

1. Wrap on the transmit interface, specifying "data" and "LIC" or "external".
2. Write ICW on the receive interface with PCF state X'7' (Receive) for start/stop operation, or X'5' (Monitor for Phase) for BSC operation. In the case of BSC, the PCF state will change automatically to X'7' when synchronization occurs. PCF state X'0' must be used to stop reception when an EOM character is recognized, or the count is exhausted.
3. Write ICW on the transmit interface with PCF state X'9' (Transmit Data).

   Note: In the case of external wrap test only, the first byte should be transmitted using PCF state X'8' (Transmit Initial). This is not mandatory, but if PCF state X'8' is not used, the first Write command with PCF state X'9' must supply both the PDF and the SDF.

PCF state X'D' must be used to stop transmission. This state, in wrap mode, only ends the transmission and returns the PCF state to X'0' (No-Op). A level 2 interrupt request is raised, and the interface is left without an outstanding command. The wrap test, however, remains in force.

4. Repeat steps two and three until all the characters in the buffer chain have been transmitted and received.

5. Repeat as often as required with a new buffer chain.

6. End the wrap with a Reset-D or a Reset-N command.

Notes:

1. Line control characters must be provided by the MOSS in the buffer chain to the control program. They are returned to the control program as received by the scanner. CRC and/or LRC checking should not be done.

2. Leading pad and synchronization characters must be included by the MOSS in the buffer chain.

3. For start/stop and character mode BSC lines, the control program must not send the first character in the buffer chain, but must use it instead to define the EOM character that must be used to complete reception.

Control Lead Wrap.

The one-byte test patterns are stored as a sequence of bytes in NCP buffers or data areas. The scanner alternatively presents one modem-out pattern and then reads it in as a modem-in pattern until the entire buffer chain or data area has been exhausted. The operation is completely automatic; no Write or Read commands are required.

The following sequence of commands should be used:

1. Wrap test on the transmit interface, specifying "control lead" and "LIC" or "external"

2. Repeat as often as required with a new buffer chain.

3. End the wrap test with a Reset-D or a Reset-N command.

Timeout Values Used

Many different timeout values are used for the different line protocols; some are fixed, and some are set by the Set Mode command.

Timeouts for SDLC

Type of timeout	Value
Enable timeout	Set Mode value
Disable timeout	Set Mode value
Reply timeout	Set Mode value
Receive text timeout	Set Mode value
Monitor rise of CTS	Same as enable
Monitor fall of CTS	3 seconds
Modem retrain	Same as enable

Timeouts for NCP BSC

Type of timeout	Value
Enable timeout	Set Mode value
Disable timeout	Set Mode value
Reply timeout	Set Mode value
Monitor rise of CTS	Same as enable
Monitor fall of CTS	3 seconds
Modem retrain	Same as enable
Ensure SYNs received during text	3 seconds
Ensure no more than 3 seconds of SYNs	3 seconds
SYN insertion during transmission	1 second

Timeouts for EP BSC

Type of timeout	Value
Enable timeout	Fixed (see Enable Cmd)
Disable timeout	Fixed and set mode values
Reply timeout (no control character received on a Receive command or after polling)	3 seconds
Monitor rise of CTS	25.6 seconds*
Monitor fall of CTS	3 seconds
Monitor fall of CTS on 2nd transparent write	1 second
Ensure SYNs received during text	3 seconds
Ensure no more than 3 seconds of SYNs	3 seconds

* two or one seconds if raised at enable time.

Timeouts for Character Mode

Type of timeout	Value
Enable timeout	Set Mode (NCP) Fixed (EP)
Disable timeout	Set Mode (NCP) Fixed (EP)

Timeouts for Autocall Interface

Type of timeout	Value
Dial timeout (NCP) Dial timeout (EP)	Set Mode value 51.2 seconds

Scanner Program/Hardware Checks causing a Level 1 Interrupt

In addition to the program and hardware errors reported in the command status via level 2 interrupts (as described under the heading "Ending Status" for each command), the CCU may be notified that a program check or a hardware check has occurred via a CCU level 1 interrupt request. To obtain information about the check, the control program must issue a Get Error Status instruction, which transfers a two-byte error status to the CCU. The control program must also reset the check and the interrupt, and perform any necessary recovery actions. The following conditions may cause a level 1 program or hardware check:

1. CCU/scanner problems:

   - IOH/IOHI op-code not supported.
   - IOH/IOHI rejected because there is an outstanding command for that interface. For example, a second Transmit command has been sent while a Transmit command is already outstanding.

     Note: Certain conditions do not cause a program check; the command is simply rejected by setting "command rejected" in the status area. For example, a Transmit Continue command has been issued without a previous Transmit command.

   - IOH/IOHI rejected because a Set Mode command has not yet been received for that interface.

2. Scanner Problems:

   - Scanner internal errors are usually detected as parity checks, but in some cases, the scanner detects program errors during its own processing.
   - Some hardware errors may occur only when handling a given line interface. For example, a parity check may occur when accessing a hardware register associated with a particular line interface. In this case, the error status contains the five-bit line interface address in addition to information identifying the type of error.

3. Abnormal conditions detected during I/O operations on the CCU to scanner bus. These errors may be detected by the CCU or by the scanner. Errors are related to CCU storage and address checks, invalid sequences, and invalid or timed out IOH/IOHI instructions.

If the check is related to a particular line interface, the scanner freezes operations on the line interface in error, creates the error status, raises a level 1 interrupt, and waits for the control program to get the error status with a Get Error Status instruction. The scanner continues to operate all other line interfaces normally.

If the check is not related to a specific line, a scanner hardstop occurs. The scanner freezes all line interface operations, creates the error status, raises a level 1 interrupt, and waits for the control program to get the error status; IOH/IOHI instructions other than Get Error Status are ignored. All communications on the lines attached to this scanner are blocked. When the CCU has obtained the error status, the scanner informs the MOSS that an error has occurred and ignores all further IOH/IOHI instructions, except those coming from the MOSS (bit 12 of the second halfword of the instruction is a one to indicate that the instruction comes from the MOSS). The MOSS may then use Display and Dump commands to collect further information about the error.

Note: To return to the operational state, it is necessary to re-IPL the scanner.

Error Status Bytes

The error status is two bytes long, and contains the following information:

- Type of IOH/IOHI check.
- Type of scanner check.
- Type of CCU/scanner bus check.
- The five-bit line interface address, if the error is associated with a specific line address.

Note: These 16 bits indicate hardware errors and are used by maintenance personnel for fault isolation. Refer to maintenance documentation for full details.

CHAPTER 6. TOKEN-RING SUBSYSTEM

The Token-Ring Subsystem (TRSS) provides the controller with physical access to a token-ring. The structure and components of the Token-Ring Subsystem are described in detail in the 3720/3721 Introduction Manual, GA33-0060. The TRSS does not perform any Data Link Control (DLC) or higher level networking functions, it is simply a data transport between the controller and the ring. The TRSS will attempt to deliver all data that the controller requests it to deliver, and will pass to the controller all user data it receives. This chapter describes those functions of the TRSS that are accessible for programming. These are the functions of the Token-Ring Multiplexor (TRM) part of the TRSS.

Communication between the TRM and the controller is provided by:

- Interrupts
- MMIO instructions to read/write information from/to the TRM RAM and registers without direct CCU control
- TRM commands that transfer user data frames between the controller and the ring using Direct Memory Access (DMA).

Two types of frame are handled by the TRM:

- User frames, also called 'frame format 1'.
- Medium Access Control (MAC) frames that are handled at the physical transmission level.

This chapter describes the format and handling of user frames, but some types of MAC frame report errors and ring problems in the status reports of some TRM commands.

This chapter is organized in the following way:

- Section 1 contains basic information about the TRM.
- Section 2 describes the procedure necessary to initialize the TRM.
- Section 3 describes the MMIO instructions.
- Section 4 describes the TRM commands.

Read all sections of this chapter before you use any commands as both MMIO instructions and TRM commands are used together.

SECTION 1. BASIC INFORMATION

The TRSS acts as a data transfer device between the controller and the ring. Data is transferred in the form of frames and under the control of TRM commands issued by the program in CCU.

The Token - Free and Captured

The token controls the transmission and, partially, the reception of frames. No frame can be transmitted to the frame without a token. When the ring is in operation, a "free" token is transmitted around the ring. If a station on the ring wants to transmit a frame, it waits until a free token arrives then "captures" it. A bit is changed in the token to show that it is busy, the station appends the frame to the token, and both are transmitted to the destination station. Each station between the source and destination receives the frame, regenerates it, and transmits it onto the ring.

When the frame arrives at the destination, the receiving station copies the frame, changes a bit in the trailer to confirm reception, and transmits the frame back to the source station. The source changes a bit in the token to show that it is free again, strips the frame from the token, and transmits the free token back onto the ring for another station to capture it. Control of the token is, however, transparent to the user and is not discussed further in this manual.

Token Format

The token is part of the frame header, described below, and consists of:

1. A starting delimiter field (one byte).
2. A Physical Control Field 0 (PCF0) (one byte).
3. An ending delimiter (one byte).

Bit 3 of the PCF0 field is used to indicate whether the token is "free" or "captured". The PCF0 field is described in detail below under "Frame Structure".

Frame Structure

A 'frame' is the unit of transmission on the IBM Token-Ring Network. It includes delimiters, control characters, user data, and frame checking characters. It consists of three components:

1. A frame header
2. The user data
3. A frame check (CRC) field.

Its structure is as follows:

Header

Field	Size
PCF0 | PCF1	2 bytes
To address	6 bytes
From address	6 bytes
Routing Field	18 bytes maximum

User data

Data	user defined length

Frame CRC

Frame CRC	4 bytes

The content of these fields is described below.

PCF0 (PHYSICAL CONTROL FIELD 0): The content of PCF0 is:

Bit	Meaning
0	Access priority
1	Access priority
2	Access priority
3	Token indicator
4 to 7	Reserved

The bits of PCF0 have the following significance:

Bits 0 to 2 - Access priority: these bits select the Access priority for the frame. This value (0-3) must be less than or equal to the Authorized Access Priority for the station.

Bit 3 - Token indicator: when on, this bit indicates that this is a busy token attached to a frame, that is, a captured token. When off, this bit indicates a free token.

Bits 4 to 7 - Reserved: these bits will be reset to '0000' by the TRM.

PCF1 (PHYSICAL CONTROL FIELD 1): This field is transmitted as specfied by the controller.

TO ADDRESS: This field specifies the destination of the transmitted frame.

FROM ADDRESS: This field specifies the source of the frame. The TRM stores the Node Address for this controller into the six bytes of the From Address field with the exception of byte 0, bit 0.

ROUTING FIELD: This must be included if bit 0 of the From Address field is one.

DATA: The data field is the user specified data from the controller.

FRAME CRC: Each frame is terminated by Cyclic Redundancy Check characters and a frame delimiter. These are appended automatically by the TRM.

TRM Direct Memory Access

There is one Direct Memory Access (DMA) channel for a TRM that performs transfers in one direction at a time between the TRM and the controller. The TRM acts as bus master in all DMA transfers.

The TRM can operate in one of two DMA modes, Cycle or Burst. In Cycle Mode, the TRM will release the bus after every memory access. In Burst Mode, after the TRM is master of the bus, the TRM will remain master until:

- A parity error is detected from TRM RAM.
- A parity error is detected from controller RAM.
- An error occurs on the system bus.
- The system bus is released.
- No more data is to be transferred. The maximum amount of data that can be transferred in each burst is set in the Initialization Parameters, described in Section 2.

Addresses provided to the TRM by the controller are fullwords. The high order byte is ignored to provide 24-bit addressing.

TRM Buffers

The TRM has 1792 bytes of RAM available for a buffer pool of sixteen 112-byte buffers. The first 8 bytes of each buffer are reserved for a buffer header and a frame will occupy as many buffers as are required to hold it, up to the maximum space available. Examples are given below of buffer space required for two different frame sizes.

1. The frame size that will exactly occupy six 112-byte buffers is:

```
Buffer headers  -  48 bytes (6 x 8 bytes)
Frame header    -  32 bytes
Data field      - 588 bytes
Frame CRC       -   4 bytes
                ------------
                  672 bytes (6 x 112-byte buffers)
```

2. The frame size that will exactly occupy three 112-byte buffers is:

```
Buffer headers  -  24 bytes (3 x 8 bytes)
Frame header    -  32 bytes
Data field      - 276 bytes
Frame CRC       -   4 bytes
                ------------
                  336 bytes (3 x 112-byte buffers)
```

The format of the buffer header is:

Content	Size
Backward Pointer	2 Bytes
Forward Pointer	2 Bytes
Buffer Status	2 Bytes
Data Length	2 Bytes

Backward pointer: this is the address of the previous buffer in the chain, if there is one, or all zeros if this is the first buffer.

Forward pointer: this is the address of the next buffer in the chain, if there is one, or all zeros if this is the last.

Buffe. status: this gives the current status of the buffer. There is one status halfword for a transmit buffer and another for a receive buffer. The significance of the bits for a transmit buffer status is:

Bit	Meaning
0	In use
1 to 6	Ignored
7	End of frame
8 to 15	Model PCFE

Bit 0 - In use: when on, this buffer is in use.

Bits 1 to 6 - Ignored: these bits are ignored.

Bit 7 - End of frame: when on, this buffer is the last one in a frame. When off, this buffer is an intermediate buffer in a frame.

Bits 8 to 15 - Model PCFE: this is a model Physical Control Field Extension byte and is only valid if bit 7, End of frame, is set to '1'.

The significance of the bits for a receive buffer is:

Bit	Meaning
0	In use
1 to 6	Ignored
7	End of frame
8 to 15	Ignored

Bit 0 - In use: when on, this buffer is in use.

Bits 1 to 6 - Ignored: these bits are ignored.

Bit 7 - End of frame: when on, this buffer is the last one of a frame. This bit is initially reset to '0' in a receive buffer.

Bits 8 to 15 - Ignored: these bits are ignored.

Data length: this gives the length of the data that occupies this buffer.

The sixteen buffers are used for both reception of frames from the ring and transmission of frames to the ring. When the controller requests a frame transmission, buffers are allocated from the buffer pool one at a time until the frame has been transferred to the TRM. A maximum number of buffers can be specified in the TRM command OPEN. Transmission requires three or more buffers and at least two buffers must be available for frame reception.

The TRM can process a maximum of two transmit frames at one time. One could be queued for transmission while the other is being transferred from the controller, or both could be queued for transmission at the same time. However, the total number of buffers used cannot exceed the transmit buffer count specified in the OPEN command. If it does, then the transmission will be terminated with an error status set.

Additional RAM can be added to the TRM to increase the number of buffers available and, if required, the buffer size. The amount of additional RAM installed is specified with the 'External RAM' parameter of the OPEN command. The buffer size can also be changed with the OPEN command. Refer to the OPEN command description for details of these parameters and limitations of buffer characteristics.

Interrupt Mechanism

TRM to Controller Interrupts

The TRM will interrupt the controller when the status of the TRM, ring, or an unfinished command changes. The controller can read the TRM interrupt register with an MMIO instruction to discover the cause of an interrupt. The controller must reset the interrupt with another MMIO instruction. An MMIO instruction can be used to stop an interrupt from being generated.

Interrupts are provided to the controller by the TRM in an 8-bit interrupt vector. The controller can specify seven values for the interrupt vector to distinguish between different interrupts. If more than one TRM is connected to the controller each TRM can have a different interrupt vector set for it.

Controller to TRM Interrupts

The TRM can be interrupted by writing to the TRM Interrupt Register. When the TRM can respond to the interrupt, it will read the Interrupt Register, service the request, and reset the interrupt.

TRM Check Interrupt

This interrupt is generated when the TRM detects an unrecoverable hardware or software error. The SSB is not altered and the TRM will be closed and waitng for TRM reset interrupt from the controller (MMIO Write Interrupt, Bit 1 set to '1'). The Open command will have to be issued again, if it is required, after reset. The reason for the error can be found reading 8 bytes from TRM RAM starting X'05E0'. Use MMIO Write Address instruction to write the address into the TRM Address Register, then MMIO Read Data Autoincrement instructions to read the 8 bytes to the controller. These bytes contain:

Address	Meaning
X'05E0'	TRM Check
X'05E2'	Parameter 0
X'05E4'	Parameter 1
X'05E6'	Parameter 3

The contents of these addresses are described below.

TRM CHECK

The bits of TRM Check have the following significance:

Bit	Meaning
0	MMIO parity error
1	DMA abort - read
2	DMA abort - write
3	Illegal operation code
4	Parity error
5	Parity error - EXT
6	Parity error - SIF
7	Parity error - PH
8	Parity error - RECV
9	Parity error - XMIT
10	Ring underrun
11	Ring overrun
12	Invalid interrupt
13	Invalid error interrupt
14	Invalid XOP
15	Program check

Bit 0 - MMIO parity check: when on, the TRM has detected a parity error in data transferred from the controller during an MMIO instruction. Parameters 0 - 2 should be ignored.

Bit 1 - DMA abort - read: when on, the TRM has aborted a DMA transfer from the controller. This could be caused by:

- Parity errors in excess of the Parity Abort Threshold set during TRM Initialization (Byte 12, DMA Abort Thresholds of the Intialization Parameters), see Page 6-14. Parameter 0 will contain X'0001'.
- Bus errors in excess of the Bus Error Abort Threshold set during TRM Initialization (Byte 12, DMA Abort Thresholds of the Intialization Parameters), see Page 6-14. Parameter 0 will contain X'0002'.
- The TRM has waited more than 10 seconds (DMA timeout) for a DMA transfer to complete, with or without errors. Parameter 0 will contain X'0000'. DMA timeout can be disabled by setting Bit 9 of the Open command Parameter List (Disable DMA Timeout) to '1', see Page 6-32.

Parameters 1 and 2 will contain the failing controller address, plus or minus six bytes.

Bit 2 DMA abort - write: when on, the TRM has aborted a transfer to the controller. Parameters 0 - 2 will contain the same as for Bit 1, DMA Abort - Read.

Bit 3 - Illegal operation code: when on, the TRM has detected an illegal operation code. Parameters 0 - 2 will contain a copy of the contents of TRM registers R13, R14, R15.

Bit 4 - Parity error: when on, the TRM processor has detected a local bus parity error. Parameters 0 - 2 will contain a copy of the contents of TRM registers R13, R14, R15.

Bit 5 - Parity error - external master: when on, the TRM has detected a local bus parity error. Parameters 0 - 2 will contain a copy of the contents of TRM registers R13, R14, R15.

Bit 6 - Parity error - system interface (SIF) master: when on, the TRM processor has detected a local bus parity error. Parameters 0 - 2 will contain a copy of the contents of TRM registers R13, R14, R15.

Bit 7 - Parity error - PH master: when on, the TRM has detected a local bus parity error. Parameters 0 - 2 will contain a copy of the contents of TRM registers R13, R14, R15.

Bit 8 - Parity error - ring transmit: when on, the TRM processor has detected a local bus parity error while transmitting to the ring. Parameters 0 - 2 will contain a copy of the contents of TRM registers R13, R14, R15.

Bit 9 - Parity error - ring receive: when on, the TRM has detected a local bus parity error while receiving from the ring. Parameters 0 - 2 will contain a copy of the contents of TRM registers R13, R14, R15.

Bit 10 - Ring underrun: when on, the TRM has detected an underrun on the ring, that is, the data is arriving out of synchronization with the TRM's clocking, and pulses are arriving later than they should. Parameters 0 - 2 should be ignored.

Bit 11 - Ring overrun: when on, the TRM has detected an overrun on the ring, that is, the data is arriving out of synchronization with the TRM's clocking, and pulses are arriving earlier than they should. Parameters 0 - 2 should be ignored.

Bit 12 - Invalid interrupt: when on, an unrecognised interrupt has been generated. Parameters 0 - 2 will contain a copy of the contents of TRM registers R13, R14, R15.

Bit 13 - Invalid error interrupt: when on, an unrecognised error interrupt has been generated. Parameters 0 - 2 will contain a copy of the contents of TRM registers R13, R14, R15.

Bit 14 - Invalid XOP: when on, an unrecognized XOP request has been generated. Parameters 0 - 2 will contain a copy of the contents of TRM registers R13, R14, R15.

Bit 15 - Program check: when on, a software error has been detected by the TRM. Parameter 0 will contain the abend code, and Parameter 1 will contain the address at which the error occurred.

PARAMETER 0, 1, 2. These bytes contain will contain a copy of TRM registers R13, R14, R15, depending on the error that has occurred, see the error descriptions above.

MMIO Instruction Set

MMIO instructions are used to transfer data and addresses and set interrupts between the controller and the TRM. This can be done with the following set of instructions:

- Write interrupt
- Read interrupt
- Write data
- Read data
- Write data autoincrement
- Read data autoincrement
- Write address
- Read address

Each of these instructions is described in detail in Section 3.

TRM Commands

TRM commands are used to transfer frames between the controller and the ring, and for address, status and error handling. This can be done with the following set of commands:

- Open
- Transmit
- Transmit Halt
- Receive
- Set Group Address

- Set Functional Address
- Read Error Log
- Read TRM
- IMPL Enable

Each of these commands is described in detail in Section 4.

System Command Block

Commands are passed to the TRM for execution in the form of a System Command Block (SCB). It is of a formatted 6-byte block containing the command code and, optionally, the address of command parameters or buffers. The command is only executed after an MMIO Write Interrupt is executed with bit 3 set to '1' in its parameter halfword. The address of the SCB in the controller's storage is passed to the TRM in the Initialization Parameters during TRM initialization described in Section 2 of this chapter. There is only one SCB per TRM.
Some commands use less than the six bytes but the TRM always reads the full six bytes. Its structure is:

SCB Address	Content
+0	Command
+2	Address
+4	Address

Command halfword: this contains the command code, in hexadecimal, given to the TRM.

Address: this contains, depending on the command, a pointer to command parameters or buffer queues. The address must be a fullword, the high-order byte being ignored by the TRM. The address field for some commands contains parameters, and other commands require only the command halfword. However, this field must always be present regardless of what it contains.

System Status Block

The System Status Block (SSB) provides the controller with the status of the ring, reasons for command reject, and the status of commands issued to the TRM. Note that it does not provide the status of frames, this is provided in the CSTAT field of the TRM commands, Transmit and Receive. The address of the SSB in the controller's storage is passed to the TRM in the Initialization Parameters during TRM Initialization, see Section 2. There is only one SSB per TRM.

The format of the SSB is:

SSB Address	Content
+0	Command
+2	Status 1
+4	Status 2
+6	Status 3

Command: used by the TRM to identify the status type. The value in this field can be:

Value	Status Type
X'0001'	Ring status (described in detail below)
X'0002'	Command reject status
X'0003' to X'000C'	The status of a TRM command passed by the controller for execution. The value equals the command code whose status is being reported in this SSB.

Status 0,1,2: this is the status of the command. Its significance is given in the description of each command.

When the status has been processed by the program in the controller, issue an MMIO Write Interrupt with halfword value X'A000' (Interrupt TRM, bit 0 , and SSB Clear, bit 2, set to '1') to reset the TRM-to-controller interrupt, and to inform the TRM that the SSB is available for additional status posting.

Ring Status

The SSB will be loaded with the status of the ring when any status condition changes. The ring status contained in the SSB is always the last reported status. The status could change, however, faster than the controller could respond to a previous Ring Status interrupt.

The SSB will be loaded with Ring Status as follows:

SSB Address	Content
+0	X'0001'
+2	Ring Status

The bits of Ring Status have the following significance:

Bit	Meaning
0	Signal loss
1	Hard error
2	Soft error
3	Transmit beacon
4	Lobe wire fault
5	Auto-removal error 1
6	Reserved
7	Remove received
8	Counter overflow
9	Single station
10	Ring recovery
11 to 15	Reserved

Bit 0 - Signal loss: when on, the receive signal is no longer present on the ring.

Bit 1 - Hard error: when on, the TRM is transmitting or receiving beacon frames to/from the ring.

Bit 2 - Soft error: when on, the TRM has transmitted a Soft Error Report MAC frame.

Bit 3 - Transmit beacon: when on, the TRM is transmitting beacon frames to the ring.

Bit 4 - Lobe wire fault: when on, the TRM has detected an open or short circuit in the lobe data path. The TRM will be closed and put in the same state as that after initialization. The Open command will have to be issued again.

Bit 5 - Auto-removal error 1: when on, the TRM has detected an internal hardware error following the Beacon Auto-removal process and has removed the controller from the ring. The TRM will be closed and put in the same state as that after initialization. The Open command will have to be issued again.

Bit 6 - Reserved: this bit will be reset to '0'.

Bit 7 - Remove received: when on, the TRM has received a Remove MAC frame. The TRM will be closed and put in the same state as that after initialization. The Open command will have to be issued again.

Bit 8 - Counter overflow: when on, an attached product counter has been incremented from 254 to 255.

Bit 9 - Single station: when on, this controller is the only station on the ring. This bit will be reset to '0' when another station signals its presence on the ring.

Bit 10 - Ring Recovery: when on, there is an error on the ring and recovery is taking place. This bit will be reset to '0' when the ring is usable again.

Bits 11 to 15 - Reserved: these bits will be reset to '0'.

SECTION 2. TRM INITIALIZATION PROCEDURE

The TRM must be initialized before it can be used, and after TRM Reset has occurred. The procedure is given below. Initialization parameters must be given during this procedure and are specified following this procedure. Each of the MMIO instructions used in this procedure are described in detail in Section 3.

Initialization Procedure

The initialization procedure is:

1. Execute the MMIO Read Interrupt instruction, described on Page 6-23, repeatedly until the **Initialize, Test, and Error** bits of the Read Interrupt Initialization halfword are as follows:

   a. If Initialize='1', Test='0', and Error='0', proceed to step 2 below. Bits 12 to 15 will be reset to '0000'.

   b. If Test='1', and Error='1', the Bring-up diagnostics have detected an unrecoverable hardware error. Bits 12 to 15 define the error that occurred.

   c. If neither of the above occurs within 3 seconds of TRM Reset, there is a hardware error. Reset the TRM and retry the initialization. If this condition persists after three retries there is a hardware error that needs the attention of a hardware service representative.

2. Set the TRM Address Register to X'0200' using the MMIO Write Address instruction.

3. Load the Initialization Parameters using MMIO Write Data Autoincrement instructions. You can check that the parameters have been correctly loaded by setting the TRM Address Register back to X'0200' and reading the first 22 bytes.

4. Execute an MMIO Write Interrupt instruction (Execute) with the halfword set to X'9080' to interrupt the TRM. The SCB is not used.

5. Execute MMIO Read Interrupt instructions repeatedly until the Initialize, Test, and Error bits of the Initialization halfword are as follows:

   a. If Initialize='0', Test='0', and Error='0', then initialization has completed without error. Bits 12 to 15 will be reset to '0000'. The SCB should contain X'0000C1E2D48B' and the SSB should contain X'FFFFD1D7C5D9C3D4'.

   b. If Error='1', the initialization has failed. Bits 12 to 15 will define the reason for failure. The initialization procedure must be restarted from TRM Reset.

   c. If neither of the above occurs within 3 seconds of TRM Reset, there is a hardware error. Reset the TRM and retry the initialization. If this condition persists after three retries there is a hardware error that needs the attention of a hardware service representative.

Initialization Parameters

The Initialization Parameters consist of 22 bytes of information that must be passed to the TRM using MMIO Write Data Autoincrement instructions. **All 22 bytes must be passed.**

Initialization Parameters

Byte	Meaning
0	Initialization options
2	Command | Transmit
4	Receive | Ring
6	SCB Clear | TRM Check
8	Receive burst size
10	Transmit burst size
12	DMA abort thresholds
14	SCB address
16	SCB address
18	SSB address
20	SSB address

Initialization Options

The bits of the Initialization Options have the following significance:

Bit	Meaning
0	1
1	Parity enable
2	Parity enable
3	Burst SCB/SSB
4	Burst list
5	Burst list status
6	Burst receive data
7	Burst transmit data
8 to 15	Reserved

Bit 0 - Reserved: this bit must always be set to '1'.

Bits 1 and 2 - Parity enable: these bits should be set to '11' if the controller bus provides **odd** parity. Parity checking is then performed on both DMA and MMIO transfers from the controller to the TRM. If parity checking is not required, then these bits should be reset to '00'.

Bit 3 - Burst SCB/SSB: when set to '1', the TRM will transfer the SCB from the controller and the SSB to the controller in DMA Burst Mode. The burst sizes are:

- 6 bytes for SCB read.
- 2 bytes for clear of SCB command.
- 8 bytes for SSB write.

When reset to '0', transfers are in Cycle Mode.

The parameters for the Read TRM instruction will be transferred to the TRM in the same mode as that specified for the SCB.

Bit 4 - Burst list: when set to '1', the TRM will transfer Transmit and Receive lists from the controller in DMA Burst Mode. The burst size will be less than or equal to 26, as specified in List Size of the Open Parameters. When reset to '0', the lists will be transferred in Cycle Mode.

Bit 5 - Burst list status: when set to '1', the TRM will transfer List Status data to the controller in DMA Burst Mode. The burst sizes are:

- 2 bytes for Transmit CSTAT.
- 4 bytes for Receive CSTAT and Frame Size.

When reset to '0', the List Status will be transferred in Cycle Mode.

Bit 6 - Burst receive data: when set to '1', the TRM will transfer to the controller in DMA Burst Mode:

- Received data.
- Data returned by the TRM command Read TRM.
- Data returned by the TRM command Read Error Log.

The burst size is specified in Receive Burst Size in bytes 8 and 9 of the Initialization Parameters.

When reset to '0', data will be transferred in Cycle Mode.

Bit 7 - Burst transmit data: when set to '1', the TRM will transfer from the controller in burst mode:

- Transmit data.
- Open Parameters for the Open TRM command.

When reset to '0', data will be transmitted in Cycle Mode.

Bits 8 to 15 - Reserved: these bits must be reset to '0'.

Command Status Vector

This byte should contain the interrupt vector that the TRM will send to the controller when the SSB is updated with the command status and command reject status of all TRM commands except Transmit and Receive.

Transmit Command Status Vector

This byte should contain the interrupt vector value that the TRM will send to the controller when the SSB is updated with Transmit command status.

Receive Command Status Vector

This byte should contain the interrupt vector value that the TRM will send to the controller when the SSB is updated with Receive command status.

Ring Status Vector

This byte contains the interrupt vector value that the TRM will send to the controller when the SSB is updated with Ring status.

SCB Clear Vector

This byte should contain the interrupt vector value that the TRM will send to the controller when the SCB interrupt is generated.

TRM Check Vector

This byte should contain the interrupt vector value that the TRM will send to the controller when the TRM check interrupt is generated.

Receive Burst Size

This halfword should contain the count of the maximum number of bytes that the TRM will transfer to the controller in DMA Burst Mode for each transfer. If the count is specified as zero, the TRM will set the burst size to the amount of data to be transferred. This parameter is ignored if bit 6 of Initialization Options, Burst Receive Data above, is set to '0'. The count must be an even value.

Transmit Burst Size

This halfword should contain the count of the maximum number of bytes that the TRM will transfer from the controller in DMA Burst Mode for each transfer. If the count is specified as zero, the TRM will set the burst size to the amount of data to be transferred. This parameter is ignored if bit 7 of Initialization Options, Burst Transmit Buffer above is set to '0'. The count must be an even value.

DMA Abort Thresholds

This halfword should contain the count of the number of times the TRM is to retry a DMA operation if it is terminated with Bus Error or Parity error. The high-order byte (bits 0 to 7) contains a count for bus errors, and the low-order byte (bits 8 to 15) contains a count for parity errors. If the count is '1' then failed operations will not be retried. Both counts must be non-zero.

SCB Address

This fullword should contain the address in the controller's RAM of the System Command Block.

SSB Address

This fullword should contain the address in the controller's RAM of the System Status Block.

SECTION 3. MMIO INSTRUCTIONS

MMIO instructions are executed by using the IOH or IOHI instructions described on Page 2-41. A halfword containing the MMIO instruction, in bits 8 to 11 inclusive, must be written into a specified CCU register before execution of the IOH/IOHI instruction. Bit 15 of the halfword indicates the direction of the MMIO operation, whether it is output from the CCU (=0), or input to the CCU (=1).

Any data, or an address, that is to be transferred to the TRM by an MMIO instruction must be present in a CCU register specified in the IOH/IOHI instruction. Any data, or an address, that is to be transferred from the TRM will be written into a register specified in the IOH/IOHI instruction.

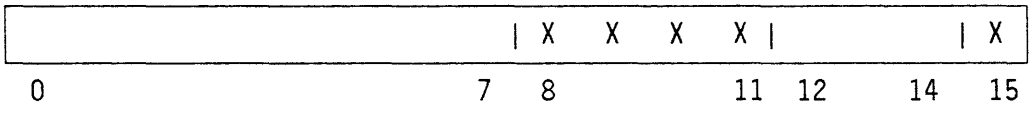
MMIO instructions available are:

MMIO Instruction	Bits 8 to 11	Bit 15
Write data	0 0 X X	0
Read data	0 0 X X	1
Write data autoincrement	0 1 X X	0
Read data autoincrement	0 1 X X	1
Write address	1 0 X X	0
Read address	1 0 X X	1
Write interrupt	1 1 X X	0
Read interrupt	1 1 X X	1

where X X is the number of the TRM that is to execute the MMIO operation:

X X	TRM
0 0	0
0 1	1
1 0	2
1 1	3

All MMIO transfers are in halfwords. Each instruction is described in detail in the following paragraphs with the instruction code for the IOH/IOHI halfword given in this form:

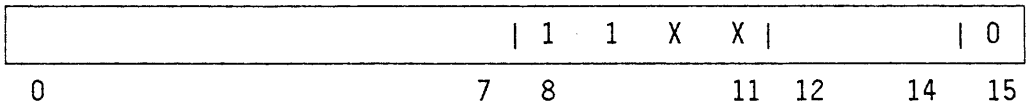


(For bits 0 to 7, and 12 to 14, refer to the description of the IOH/IOHI instruction on Page 2-41).

Write Interrupt

This instruction transfers a halfword to the TRM interrupt register and is used to interrupt the TRM and reset the TRM-to-controller interrupt.

The instruction code is:



(For bits 0 to 7, and 12 to 14, refer to the description of the IOH/IOHI instruction on Page 2-41).

The format of the halfword to be transferred is:

Bit	Meaning
0	Interrupt TRM
1	TRM reset
2	SSB clear
3	Execute TRM command
4	SCB request
5	Receive continue
6	Receive valid
7	Transmit valid
8	Reset controller interrupt
9 to 15	Reserved

Bit 0 - Interrupt TRM: when on, the TRM will be interrupted. When off, it has no effect. Bits 2 to 8 specify the interrupt requested but are examined only when the TRM has been interrupted. A '0' value for any of these bits has no effect, only a '1' is serviced. Any or all interrupt requests can be issued at any time.

Bit 1 - TRM reset: setting this bit to '1' with bits 2 to 7 also set to '1' will force the TRM to reset. After reset, the TRM should be re-initialized according to the instructions in "TRM Initialization" in Section 2. TRM Reset will also result in the execution of diagnostics.

Bit 2 - SSB clear: when set, it indicates to the TRM that the SSB is available for the TRM to post additional status information. SSB Clear should be used with Interrupt TRM when clearing a TRM-to-controller interrupt.

Bit 3 - Execute: setting this bit will cause the TRM to execute a TRM command specified in the SCB. All parameters, addresses, or lists, associated with the TRM command must have been prepared before this instruction is executed.

Bit 4 - SCB request: setting this bit will cause the TRM to interrupt the controller when the SCB is available for another command. The TRM will return an SCB Clear interrupt code which can be read by the MMIO Read Interrupt instruction.

For Transmit and Receive TRM commands this will occur when the first Transmit or Receive List is read into the TRM. For other commands, the interrupt will be generated after the command has completed and the SSB has been updated. The SSB is not altered when the SSB Clear interrupt is generated.

When the interrupt is recognized, the controller should examine the Command halfword of the SSB. If it is zero, the SSB is free for use. If it is not zero, an Execute interrupt was issued or the SCB was altered in preparation for an Execute subsequent to the SCB request.

Programming Note:

If SCB Request is desired, it is recommended that either the SCB Request be issued coincident with Execute or that the SCB alteration and Execute be performed only in response to SSB Clear.

Bit 5 - Receive continue: when set, it indicates to the TRM that buffers have been added to the Receive List chain.

Bit 6 - Receive valid: when set, it indicates to the TRM that the condition that caused a suspension of Receive List processing has been rectified. This interrupt is used when the TRM command Receive has had its List Valid bit changed from '0' to '1'.

Bit 7 - Transmit valid: when set, it indicates to the TRM that the condition that caused suspension of Transmit List processing has been rectified. This interrupt is used when the TRM command Transmit has had its List Valid bit changed from '0' to '1'.

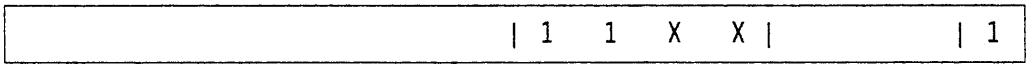
Bit 8 - Reset controller interrupt: when '0', it resets the TRM-to-controller interrupt. A value of '1' has no effect. Bit 0, Interrupt TRM, and bit 2, SSB Clear, must always be set to '1' when this bit is reset to '0'.

Bits 9 to 15 - Reserved: these bits are ignored by the TRM but nevertheless must always be sent.

Read Interrupt (Normal)

This instruction is used to read the TRM interrupt register and should be executed after every TRM-to-controller interrupt if interrupt vectors are not used. The content of the interrupt register is written into a CCU register specified in the IOH/IOHI instruction.

The instruction code is:



(For bits 0 to 7, and 12 to 14, refer to the description of the IOH/IOHI instruction on Page 2-41).

The TRM transfers a halfword as follows:

Bit	Meaning
0	TRM interrupt
1	TRM reset
2	SSB clear
3	Execute
4	SCB request
5	Receive continue
6	Receive valid
7	Transmit valid
8	Interrupt controller
9 to 11	Reserved
12	Interrupt code 0
13	Interrupt code 1
14	Interrupt code 2
15	Reserved

Bit 0 - TRM interrupt: when on, there is a current controller-to-TRM interrupt outstanding. When off, there is no interrupt outstanding.

Bits 1 to 7: these bits show the state of the interrupt requests that were issued by Write Interrupt instruction.

Bit 8 - Interrupt controller: when on, the TRM-to-controller interrupt request has been set. This bit is reset when the interrupt request has been reset by the controller.

Bits 9 to 11 - Reserved: these bits will be reset to '0'.

Bits 12 to 14 - Interrupt code: these bits define the reason for the TRM-to-controller interrupt. The lower the code value, the higher is the interrupt priority. The bits are presented serially by priority if multiple interrupts are pending. If bit 8 is '0' then no interrupt is pending and bits 12 to 14 should be ignored.

The interrupt codes have the following value:

Code	Meaning
000	TRM check
001	IMPL force
010	Ring status
011	SCB clear
100	Command status
101	Receive status
110	Transmit status

000 - TRM check: the TRM has encountered an unrecoverable hardware or software error.

001 - IMPL force: the TRM has received an IMPL Force MAC frame and an IMPL Enable command has been issued.

010 - Ring status: the SSB has been updated with a Ring status.

011 - SCB clear: this code will be set when, following an SCB Request interrupt, the SCB is clear.

100 - Command status: the SSB command status has been updated. This does not apply to commands Receive or Transmit.

101 - Receive status: the SSB has been updated with Receive Command status.

110 - Transmit Status: the SSB has been updated with Transmit Command status.

Bit 15 - Reserved: this bit will be reset to '0'.

Read Interrupt (Initialization)

This has the same operation as Read Interrupt (Normal) except that it returns a different halfword. This halfword is returned **only** during the initialization phase.

The bits of the initialization halfword have the following significance:

Bit	Meaning
0 to 8	Ignored
9	Initialization
10	Test
11	Error
12	Error code 0
13	Error code 1
14	Error code 2
15	Error code 3

Bits 0 to 8 - Ignored: these bits should be ignored, they have no significance for initialization.

Bit 9 - Initialization: when on, the TRM bring-up diagnostics have completed and the TRM is starting the initialization sequence. This will be reset when initialization has completed or there has been an error.

Bit 10 - Test: when on, the bring-up diagnostics have started following TRM reset. This bit is reset when bit 9, Initialization, is set to '1'.

Bit 11 - Error: when on, then either the bring-up diagnostics have detected an error, or an error has occurred during initialization. Bits 12 to 15 define the error.

Bits 12 to 15 - Error code: this is a 4-bit code that defines the error that occurred. If bit 10, Test, is set to '1', then this code applies to the Bring-up diagnostics. If bit 10 is '0', then this code applies to the initialization phase.

Bring-up Error Codes:

Code	Meaning
0000	Initial test error
0001	ROS CRC error
0010	RAM error
0011	Instruction test error
0100	XOP test error, interrupt test error
0101	PH hardware error
0110	SIF register error

Initialization Error Codes:

Code	Meaning
0001	Invalid parameter length
0010	Invalid options
0011	Invalid receive burst size
0100	Invalid transmit burst size
0101	Invalid DMA abort thresholds
0110	Invalid SCB address
0111	Invalid SSB address
1000	MMIO parity error
1001	DMA timeout
1010	DMA parity error
1011	DMA bus error
1100	DMA parity error
1101	TRM check

0001 - Invalid parameter length: 22 bytes were not passed for the Initialization Parameters.

0010 - Invalid options: in the initialization parameters, Tranfer Mode is not '1', or the Parity Enable bits are not equal, or the Reserved bits are not all '0'.

0011 - Invalid receive burst size: Receive burst size is odd.

0100 - Invalid transmit burst size: Transmit burst size is odd.

0101 - Invalid DMA abort thresholds: either the Bus error or Parity error count is zero.

0110 - Invalid SCB address: the SCB address is odd.

0111 - Invalid SSB address: the SSB address is odd.

1000 - MMIO parity error: .a parity error was detected during a controller MMIO Write operation.

1001 - DMA timeout: a test DMA transfer took more than 10 seconds to complete.

1010 - DMA parity error: a parity error was detected in a test DMA transfer from the controller and the transfer was tried, unsuccessfully, the number of times specified in the DMA Abort Thresholds of the Initialization Parameters.

1011 - DMA bus error: the controller has detected a bus error during a test DMA transfer and the transfer was tried, unsuccessfully, the number of times specified in the DMA Abort Thresholds of the Initialization Parameters.

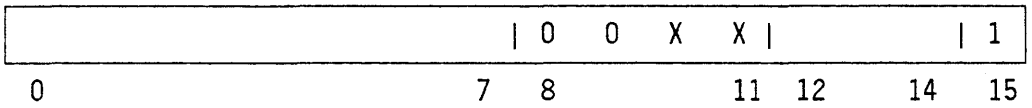
1100 - DMA data error: the initialize DMA test has failed because of a data compare error.

1101 - TRM check: the TRM has detected an unrecoverable hardware error. The error can be read from TRM RAM, see Page 6-7.

Read Data

This instruction is used to read a halfword from the TRM from a location previously loaded in the TRM Address Register by an MMIO Write Address instruction.

The instruction code is:



(For bits 0 to 7, and 12 to 14, refer to the description of the IOH/IOHI instruction on Page 2-41).

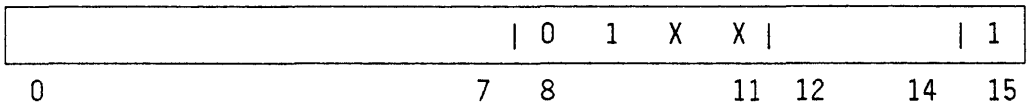
The halfword will be written into the register specified in the IOH/IOHI instruction.

After the TRM has been initialized, many TRM RAM locations can be accessed by MMIO Read instructions. The parameters and their addresses are described in Read TRM Command. Only the contents of RAM addresses X'0000' to X'07FF' can be read with MMIO Read Data instruction.

Read Data Autoincrement

This instruction is the same as Read Data except that the Address Register is automatically incremented so that the next location can be read.

The instruction code is:



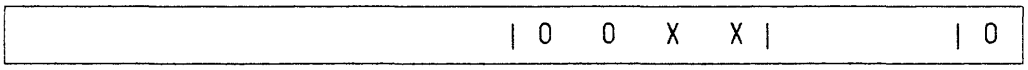
(For bits 0 to 7, and 12 to 14, refer to the description of the IOH/IOHI instruction on Page 2-41).

Write Data

This instruction writes a halfword from the controller to the TRM. The TRM address is specified by loading the TRM Address Register with an MMIO Write Address instruction.

The Write Data instruction is ignored after the TRM has been initialized as described in the TRM Initialization procedure in Section 2.

The instruction code is:

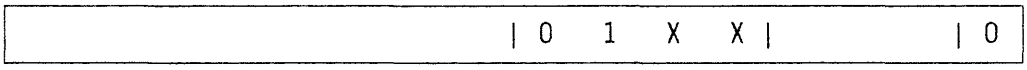


(For bits 0 to 7, and 12 to 14, refer to the description of the IOH/IOHI instruction on Page 2-41).

Write Data Autoincrement

This instruction is the same as Write Data except that the Address Register is automatically incremented so that the next location can be written.

The instruction code is:

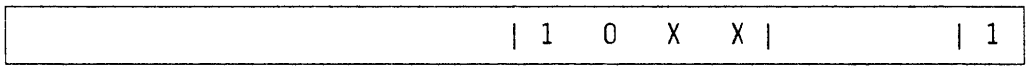


(For bits 0 to 7, and 12 to 14, refer to the description of the IOH/IOHI instruction on Page 2-41).

Read Address

This instruction reads the TRM Address Register and places the content into the register specified in the IOH/IOHI instruction.

The instruction code is:

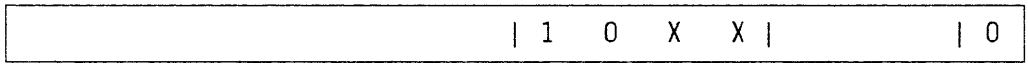


(For bits 0 to 7, and 12 to 14, refer to the description of the IOH/IOHI instruction on Page 2-41).

Write Address

This instruction is used to load an address into the TRM Address Register for the Write Data, Write Data Autoincrement, Read Data, and Read Data Autoincrement instructions.

The instruction code is:



(For bits 0 to 7, and 12 to 14, refer to the description of the IOH/IOHI instruction on Page 2-41).

The TRM Address Register is 16 bits, but only bits 5 to 14 can be set or reset by the controller. All 16 bits can be read.

Note:

- Bits 0 to 4, and bit 15 are set by the TRM.
- Bit 15 will always be '0'
- During initialization, bits 0 to 4 will be '00001'. After initialization, bits 0 to 4 will be '00000'.
- After TRM Check interrupt, bits 0 to 4 will be '00000'.
- When bits 0 to 4 are '00000', an MMIO read of TRM RAM will be restricted to addresses X'0000' to X'07FF'.

SECTION 4.TRM COMMANDS

The following commands can be issued from the controller:

Command	Code
Open	X'0003'
Transmit	X'0004'
Transmit Halt	X'0005'
Receive	X'0006'
Close	X'0007'
Set Group Address	X'0008'
Set Functional Address	X'0009'
Read Error Log	X'000A'
Read TRM	X'000B'
IMPL Enable	X'000C'

A TRM command is executed as follows:

1. Prepare any data areas, load addresses, or construct Transmit/Receive Lists as appropriate to the TRM command.

2. The command code and any addresses must be written into the SCB.

3. An MMIO Write Interrupt instruction must be executed with Bit 3 (Execute) set to '1'.

4. Status information relevant to the command will be written into the SSB following the execution, successful or otherwise, or rejection, of the command.

Command Rejection

If a command code other than those listed above is issued, or an individual command error is detected, the SSB will contain the reason for rejection.

SSB Address	Content
+0 +2 +4	X'0002' Reject Reason Invalid Command

Reject reason: this is a 2-byte field whose bits define the reject.

The bits of Reject Reason have the following significance:

Bit	Meaning
0	Illegal command
1	Address error
2	TRM open
3	TRM closed
4	Same command
5 to 15	Reserved

Bit 0 - Illegal command: when on, an illegal command code has been put into the SCB.

Bit 1 - Address error: when on, an address supplied in the SCB is odd.

Bit 2 - TRM open: when on, a command was issued that can only be executed when the TRM is closed.

Bit 3 - TRM closed: when on, a command was issued that can only be executed when the TRM is open.

Bit 4 - Same command: when on, a command has been issued that is currently being executed.

Bits 5 to 15 - Reserved: these bits will always be reset to '0'.

Invalid command: this is the code of the command that was rejected.

Open (X'0003')

The Open command must be used before any communication can begin. During the processing of this command the TRM will be enabled to receive frames. The TRM will also suspend all processing of interrupt requests, except Reset, from the time the Open is issued until successful completion. A Receive command must be issued immediately after the Open command has completed successfully. To change any options in a current Open command, a Close command must be issued followed by another Open with the new options. The Open command will be rejected with command reject status set in the SSB if the TRM is already open.

System Command Block

The SCB for an Open command is:

SCB Address	Content
+0	X'0003'
+2	Address
+4	Address

X'0003': is the Open command code.

Address: is a fullword pointer to the Open Parameter list in controller storage.

Open Parameter List

Byte	Meaning
0	Open options
2	Node address
4	Node address
6	Node address
8	Group address
10	Group address
12	Functional address
14	Functional address
16	Receive list size
18	Transmit list size
20	Buffer size
22	External RAM start address
24	Transmit buffer minimum count
26	Transmit buffer maximum count
28	Product identifier address
30	Product identifier address

OPEN OPTIONS: The bits of the Open options bytes have the following significance:

Bit	Meaning
0	Wrap interface
1	Disable hard error
2	Disable soft error
3	Pass TRM MAC frames
4	Pass Attention MAC frame
5	Pad routing field
6	Frame hold
7	Contender
8	Pass Beacon MAC frame
9	Disable DMA timeout
10 to 1	Not used

Bit 0 - Wrap Interface: when this bit is on, all user data transmitted to the ring will be re-input to the TRM as if it was received data. The data is transmitted on the lobe from the attached controller to the wiring concentrator. This can be used for system interface testing, system interface DMA testing, or lobe media testing. A Close command must be issued to terminate Wrap mode.

Bit 1 - Disable Hard Error: when this bit is on, the ring status Hard Error and Transmit Beacon interrupts will not be generated. The bits in ring status will always be set.

Bit 2 - Disable Soft Error: when this bit is on, the ring status Soft Error interrupts will not be generated. The bits in ring status will always be set.

Bit 3 - Pass TRM MAC frames: when on, MAC frames will be passed to the controller as if they were normal data frames. When off, the TRM will respond negatively to all unsupported TRM class MAC frames.

Bit 4 - Pass Attention MAC frames: when on, all Attention MAC frames that are not equal to the last Attention MAC frame will be passed to the controller as normal receive data. When off, no Attention MAC frames are passed to the controller.

Bit 5 - Pad routing field: When on, the TRM will pad the routing field of the received frame to 18 bytes. If no routing field is present in the received frame, then the entire field will be padded to 18 bytes. If the current buffer's data count is less than 32 bytes the frame will be passed as if this bit was not on.

Bit 6 - Frame hold: when on, the TRM will wait for receive frame completion before the frame is passed to the controller.

Bit 7 - Contender: when on, the TRM will participate in Monitor Contention if it has an option to do so.

Bit 8 - Pass Beacon MAC Frame: when on, the TRM will pass Beacon MAC frames to the controller. After passing the Beacon MAC frame, the next Beacon MAC frame will be passed only if the source address or the Beacon type changes.

Bit 9 - Disable DMA timeout: when this bit is on, the DMA timeout will be ignored.

Bits 10 to 15: these bits are not used.

NODE ADDRESS: This 6-byte parameter is set to the TRM's node address on the ring. If this addresss is all zeros, the TRM will use the Burned-in-address (BIA) read from the BIA PROM. When the node address is not zero the following checks are made:

1. Byte 0, bits 0 and 1 must equal X'01'.

2. Byte 2, bit 0 must equal X'0'.

If the BIA is used, the TRM will check that the first halfword is equal to X'7FFF'. If all these checks fail the TRM will set 'Node Address Error' in the SSB.

GROUP ADDRESS: This fullword parameter is used to set a Group Address and allows the TRM to receive frames that are sent to either the Node Address with Address Modifier, or the Group Address. Group Address can be any value and bit 0 is ignored by the TRM. If a Group Address is not required, this parameter must be set to X'00000000'.

FUNCTIONAL ADDRESS: This fullword parameter is used to set a Functional Address and allows the TRM to receive frames that are sent to the Node Address with Address Modifier, the Group Address, or the Functional Address. Bits 0 and 31 are ignored by the TRM. If a Functional Address is not required, this parameter must be set to X'00000000'.

RECEIVE LIST SIZE: This halfword decimal parameter is used to limit the number of bytes that the TRM will read of the Receive List from the controller. This parameter must be 0, 14, 20, or 26. If set to zero, the TRM will use the default value 26.

TRANSMIT LIST SIZE: This halfword decimal parameter is used to limit the number of bytes that the TRM will read of the Transmit List from the controller. This parameter must be 0, 14, 20, 26. If set to zero, the TRM will use the default value 26.

BUFFER SIZE: This halfword parameter sets the buffer size in the TRM. The size specified for the buffer must be exactly divisible by 8. The Buffer Size must be greater than or equal to X'60' (=96 decimal) and the three low-order bits must be X'000'. If set to zero, the TRM will use the default value X'70' (=112 decimal). If the buffer pool is 1792 bytes, that is, no external RAM is installed, it is recommended that:

- Buffer Size is set to 112.
- Transmit Buffer Count is set to 6.
- The maximum data length is restricted to 588 bytes.

EXTERNAL RAM START ADDRESS: This halfword parameter must be used to define the start address of external RAM for use as transmit and/or receive buffers, if any is installed, or set to zero if there is none. When external RAM is used, the external RAM addresses must be within the range X'2000' to X'C000'. The external RAM start address must be on an 8-byte boundary minus two bytes (bits 13 and 14 are X'11'). If internal TRM RAM is not to be used for transmit/receive buffers, then bit 15 must be set to one.

EXTERNAL RAM END ADDRESS: This halfword parameter defines the highest address in the external RAM. If the External RAM Start Address parameter is zero, then this end address will be ignored.

TRANSMIT BUFFER MINIMUM COUNT: This 1-byte parameter defines the number of TRM buffers to be reserved as transmit buffers. Note that the buffers will never be used as receive buffers. If zero, no buffers will be allocated for transmit operations. This parameter must be less than or equal to the Transmit Maximum Buffer Count parameter described in the next paragraph.

TRANSMIT BUFFER MAXIMUM COUNT: This 1-byte parameter defines the maximum number of TRM buffers to be reserved as transmit buffers. This parameter must be less than or equal to the total number of buffers minus two so that there are always at least two buffers available for receive buffers. When this parameter is set to zero, the TRM will use a default value of 6.

The Transmit Buffer Count and the Buffer Size parameter are used to calculate the maximum frame size that the TRM can transmit.

PRODUCT ID ADDRESS: This fullword parameter contains the address of the controller's Product ID. Eighteen bytes are read from the address specified.

System Status Block

When the Open command completes the SSB will contain the following:

SSB Address	Content
+0	X'0003'
+2	Open Completion

The bits of Open Completion have the following significance:

Bit	Meaning
0	TRM open
1	Node address error
2	List size error
3	Buffer size error
4	External RAM error
5	Transmit count error
6	Open error
7	Zero
8	Open phase C0
9	Open phase C1
10	Open phase C2
11	Open phase C3
12	Open error C0
13	Open error C1
14	Open error C2
15	Open error C3

Bit 0 - TRM open: when on, the Open command has completed successfully and all other bits will be zero.

Bit 1 - Node address error: when on, an error has been found in the Node Address parameter of the Open command options, or the BIA if the Node Address parameter was zero.

Bit 2 - List size error: when on, either the Receive list or Transmit list size was equal to 0, 14, 20, or 26 decimal.

Bit 3 - Buffer size error: when on, the buffer size was specified as one of the following:

- Negative
- Not greater than or equal to X'60' (96 decimal)
- The three low order bits are not '000'
- There are not at least two buffers.

Bit 4 - External RAM error: when on, one of the following errors has occurred:

- The start address specified is not within the range X'2000' to X'C000'.
- The address is not properly aligned.
- An error has been detected in the RAM that does not cause a parity error.

Bit 5 - Transmit buffer count error: when on, the number of buffers minus the Transmit Buffer Count is not greater than or equal to two.

Bit 6 - Open error: when on, an error has been detected during processing of the Open command. Bits 8 to 15, described below specify the reason.

Bit 7 - Reserved: This bit will be reset to zero.

Bits 8 to 11 - Open phase: these bits indicate the phase during which an error, defined by bits 12 to 15, was detected:

Code	Meaning
0001	Lobe media test
0010	Physical insertion
0011	Address verification
0100	Roll call poll
0101	Request parameters

Bits 12 to 15 - Open error code: these bits are set if a ring-related error occurred during processing of the Open command:

Code	Meaning
0001	Function failure
0010	Signal loss
0011	Wire fault
0100	Frequency error
0101	Timeout
0110	Ring failure
0111	Ring beaconing
1000	Duplicate node address
1001	Request parameters
1010	Remove received
1011	IMPL force received

Transmit (X'0004')

The Transmit command is used to transmit frames to other nodes on the ring. The command will be rejected with command reject status set in the SSB if:

- The TRM is not open.
- There is already a Transmit command being executed.
- The address passed in the SCB is not halfword aligned.

The address of user data in controller storage is passed to the TRM by the controller in the form of a Transmit List. This list contains the address and length of user data that are to comprise the frame to be transmitted. More than one data address can be stated in the list, thus creating a chain of user data for transmission in one frame. Note that one Transmit List chain can only be used to transmit one frame, but several Transmit Lists can be used to transmit a single frame.

System Command Block

The SCB for a Transmit command is:

SCB Address	Content
+0	X'0004'
+2	Address
+4	Address

X'0004': is the Transmit command code.

Address: is a fullword pointer to the Transmit List chain. The address must be halfword aligned.

Transmit List Chain

Byte	Meaning
0	Forward pointer
2	Forward pointer
4	Transmit CSTAT
6	Frame size
8	1 | Data count
10	Data address
12	Data address
14	1 | Data count
16	Data address
18	Data address
20	0 | Data count
22	Data address
24	Data address

FORWARD POINTER This is an **even** fullword address to the next Transmit List in the chain. When this address is **odd** then this Transmit List is the last in the chain. The TRM will continue to read Transmit Lists until it reaches one that contains an odd address forward pointer. It will then wait until the last frame is transmitted onto the ring. If the controller updates the forward pointer before the last frame is transmitted, then the TRM will continue. When the last frame has been transmitted, the Transmit command will complete and another must be issued for the next transmission. The controller must update the forward pointer from the most significant to the least significant byte to ensure that the address is valid before changing it from an odd to an even address. Whole frames should be added to the chain, not lists that define partial frames. Transmit Lists must be halfword aligned, the TRM will not alter this parameter.

Programming Notes:

The controller can create a chain comprising a fixed number of Transmit Lists, set the Forward Pointer of the last list to the address of the first list, then manipulate the List Valid bit of the Transmit CSTAT request to initiate transmission.

When the TRM reads a Start of Frame list with the List Valid bit reset to '0', it will suspend processing until a Transmit Valid interrupt request is issued by the controller, see MMIO Write Interrupt instruction. The controller is **not** informed of this suspension. The Transmit Valid interrupt **must** be issued when changing one or more List Valid bits from '0' to '1' when the list is on the Transmit chain.

The Transmit Valid interrupt can be issued at any time and the TRM will ignore the interrupt if it is not waiting for a List Valid bit transition.

If the fixed-Transmit-chain technique is used and more than one list is used to transmit a single frame, then lists that do not have Start of Frame set to '1' should have their List Valid bit reset to '0'. Since the TRM does not alter the CSTAT for lists that are not Start of Frame, revalidation of the Start of Frame list would also release the remaining frame lists if the List Valid bits were initially set to '1'.

FRAME SIZE: This is the total number of bytes to be transmitted including the header for this frame but excluding the frame check and delimiter bytes. This parameter is only valid for a Transmit List that has Start of Frame set to '1' in its Transmit CSTAT. However, Frame Size must be included in **all** Transmit Lists. This parameter is not altered by the TRM.

DATA COUNT: This is the number of bytes to be transmitted from the data address defined in the next parameter, Data Address. There can be a maximum of three Data Count/Data Address parameter pairs for any Transmit List. Bit 0 of Data Count must be set to:

- '1' to indicate there is a following Data Count/Data Address pair.
- '0' to indicate that this Data Count/Data Address is the last pair in **this** Transmit List.

The sum of all Data Count parameters in all Transmit Lists used for a frame must equal the Frame Size parameter. Data Count can be an odd or even quantity. The TRM will not alter this parameter.

DATA ADDRESS: This is the address of the data to be transmitted, and can be odd or even. The TRM will not alter this parameter.

Note:

If the TRM is to read data from an even controller address to an odd TRM RAM address (due to an odd Data Count for example), it will transfer a single byte and transfer the remaining data starting at the next odd controller address.

TRANSMIT CSTAT: This is a command status halfword that is set up by the controller when the Transmit List is created and overwritten by the TRM with a completion status when the Transmit Command has completed.

Transmit CSTAT Request: The controller must set the bits of this parameter as follows:

Bit	Meaning
0	List valid
1	Frame complete
2	Start of frame
3	End of Frame
4	Frame interrupt
5 to 15	Reserved

The bits of the Transmit List CSTAT request have the following significance:

Bit 0 - List Valid: the TRM will wait until this bit is set to '1' before processing this Transmit List. The controller must issue a Transmit Valid interrupt request when it changes the List Valid bit from '0' to '1'. This bit is ignored unless this List is an anticipated Start of Frame, that is, it follows an End of Frame or is the first List of this command.

Bit 1 - Frame complete: this bit should alway be reset to '0' for a CSTAT request.

Bit 2 - Start of Frame: this bit must be set to '1' if this Transmit List is the first of the frame to be transmitted.

Bit 3 - End of Frame: this bit must be set to '1' if the Transmit List is the last for the frame to be transmitted.

Bit 4 - Frame Interrupt: if this bit is set to '1' then the TRM will interrupt when this frame has been transmitted rather than wait until all frames in the chain have been transmitted. This bit is only valid if Start of Frame has been set to '1' in this Transmit List.

Bits 5 to 15 - Reserved: these bits are ignored.

Transmit CSTAT Completion: The TRM will return status information to the CSTAT in the Transmit List that contains Start of Frame = '1' when the frame has been transmitted. This status information is relevant to this frame only. (Status information for the **Transmit command** is returned into the SSB on completion of the command). The CSTAT in the following Transmit Lists for the frame will not be altered.

The CSTAT for completion is:

Bit	Meaning
0	List Valid
1	Frame complete
2	Start of frame
3	End of frame *
4	Frame interrupt *
5	Transmit error
6	Reserved *
7	Reserved *
8	PCFE 0 (ARI)
9	PCFE 1 (FCI)
10	PCFE 2
11	PCFE 3
12	PCFE 4 (ARI)
13	PCFE 5 (FCI)
14	Zero
15	Zero

* These bits (3, 4, 6, and 7) will be the same as they were in the CSTAT request.

The significance of the changed bits is as follows:

Bit 0 - List Valid: this bit will be reset to '0'.

Bit 1 - Frame Complete: this bit will be set to '1'.

Bit 2 - Start of Frame: this bit will be '1'.

Bit 5 - Transmit Error: this bit will be set if there has been a frame transmit or strip process error.

Bits 8 to 15 - Stripped PCFE: these bits contain a copy of the PCFE byte returned when the transmitted frame has been stripped off the ring. (When Bit

5, Transmit Error is set to '1' ignore the PCFE bits). Bits 13 and 14 will be reset to '0'.

System Status Block

When the Transmit command completes, the SSB will contain the following:

SSB Address	Content
+0	X'0004'
+2	Transmit Completion
+4	List address
+6	List address

TRANSMIT COMPLETION: The two bytes of transmit completion report the status of the completed frame:

Bit	Meaning
0	Command complete
1	Frame complete
2	List error
3 to 7	Reserved
8	Frame size
9	Transmit threshold
10	Odd address
11	Start of frame
12	Unauthorized priority
13	Unauthorized MAC
14	Zero
15	Zero

The bits of Transmit Completion have the following significance:

Bit 0 - Command complete: when on, this bit indicates:

- The command has completed and the List Address field of the same SSB contains the address of the last Transmit List processed.
- The command was terminated by a Transmit Halt command and no frames have been transmitted. In this case the List Address will contain X'00000000'.

Note that Command Complete and Frame Complete (described below) will not be set at the same time.

Bit 1 - Frame Complete: when on, a frame has been transmitted and Frame Interrupt (bit 4 of the CSTAT request) was set.

Since frames on the Transmit Chain can be transmitted faster than the controller can respond to interrupts and faster than the TRM can cause the interrupts, the Frame Complete interrupt can report the completion of more than one frame at a time. If lists with Frame Interrupt set are mixed with lists that do not have this set, then Frame Complete can include both types of frame.

List Address in the same SSB will contain the address of the last Transmit List of the last frame that was transmitted.

Bit 2 - List error: when on, there has been an error in one of the lists that comprise a frame. Bits 8 to 13 define the error. List Address in the same SSB contains the address of the list that starts the frame in which the error occurred. This bit will not be set until all other transmit status bits have been posted. The CSTATs of error lists will not be altered by the TRM. Neither Command Complete nor Frame Complete will be set if List Error has been set.

The Transmit command will be terminated and the controller must issue another Transmit command to continue transmission.

Bits 3 to 7 Reserved: these bits will be reset to '0'.

Bit 8 - Frame size: when on, this bit indicates either:

- The Frame Size parameter in the Transmit List does not equal the sum of the Data Counts in all the Transmit Lists of the frame, or
- The Frame Size is less than the required header plus one byte of data.

Bit 9 - Transmit threshold: when on, a Frame Size has exceeded the Buffer Count specified in the Open command.

Bit 10 - Odd address: when on, a forward pointer with an odd address was found in a Transmit List that was not an end of frame.

Bit 11 - Start of frame: when on, Start of Frame was set for a Transmit List that is not an anticipated Start of Frame, or it was not set on an anticipated Start of Frame.

Bit 12 - Unauthorized access priority: when on, the requested access priority has not been authorized.

Bit 13 - Unauthorized MAC frame: when on, it indicates that the controller tried to send a MAC frame and one of the following errors was set:

- The TRM is not authorized to send a MAC frame with the specified source class.
- The MAC frame has a source class of 0.
- The MAC frame PCF ATTN field is greater than 1.

Bits 14 and 15 - Reserved: these bits will be reset to '0'.

Transmit Halt (X'0005')

This command is used to interrupt the proccessing of a Transmit List chain. When this command is executed, the TRM will terminate the transmit chain as soon as possible. Any frames queued will be purged and the Transmit command terminated with the Command Complete bit set in the Transmit command's SSB. If a Transmit command is not being executed, then Transmit Halt will be ignored.

System Command Block

The SCB for a Transmit Halt Command is:

SCB Address	Content
+0	X'0005'
+2	Ignored
+4	Ignored

X'0005': is the Transmit Halt command.

Ignored: this fullword is read by the TRM but is ignored for command execution.

System Status Block

There is no SSB for Transmit Halt command.

Receive (X'0006')

This command is used to receive frames from other nodes on the ring. Receive command is normallly issued once only after an Open command because received data is added dynamically, that is when it arrives, to a Receive List chain. If you have had to Close then Open again to change Open options, you will have to issue another Receive command. The Receive command will be rejected with Command Reject status set in the SSB when:

- The TRM has not been opened with an Open command,
- There is already a Receive command executing, or
- The address passed in the SCB is not halfword aligned.

The data portion of the received frame is transferred to the controller as received from the ring. The frame check and delimiter bytes are not transferred.

The controller must create a chained Receive List and pass the first address in the SCB of the Receive command. A single Receive List cannot be used to receive more than on frame, but several Receive Lists can be used to receive a single frame.

Rerouting Received Data

If you want to reroute a received frame, or part of a frame, according to the content of a frame header, you can do this as follows:

1. Set the Frame Hold bit in the Open command optiors to '1'.
2. Create a Receive List that has an odd Forward Pointer and one Data Count/Data Address pair sufficient to hold the desired header.
3. The TRM will use the list and interrupt with Receive Suspended set in the Receive command SSB, leaving the CSTAT unchanged. (If the entire frame is less than or equal to the Data Count, a Frame Complete interrupt will be set if this has been requested in the Receive command CSTAT request).
4. After Receive Suspended has been set, the controller can examine the frame's header and determine the frame's new destination. Frame Size will not be updated by the TRM and is not valid.
5. Create additional lists to receive the data followed by another header list with an odd Forward Pointer.
6. Issue a Receive Continue interrupt request.
7. When the frame has been transferred, a Frame Complete interrupt will occur if this has been requested.
8. Create a Transmit List chain for the received frame and transmit it with the Transmit Command.

System Command Block

The SCB for a Receive command is:

SCB Address	Content
+0	X'0006'
+2	Address
+4	Address

X'0006': is the Receive command code.

Address: is a fullword pointer to the Receive List chain. The address must be halfword aligned.

Receive List Chain

Byte	Meaning
0	Forward pointer
2	Forward pointer
4	Receive CSTAT
6	Frame size
8	1 | Data count
10	Data address
12	Data address
14	1 | Data count
16	Data address
18	Data address
20	0 | Data count
22	Data address
24	Data address

FORWARD POINTER: This is a fullword pointer to the next Receive List in the chain. When this address is **odd** then this is the last Receive List in the chain. The TRM will write a received frame into the address, or addresses if more than one is given, specified in the Receive List then check the Forward Pointer. If it is odd, the TRM will interrupt the controller with a request to place additional lists on the chain. The Receive command will not be terminated. The TRM will wait for a Receive Continue interrupt request to resume the receive operation, see MMIO instruction 'Write Interrupt'. The controller must update the Forward Pointer from the most significant to least significant byte to ensure that the address is valid before changing to an

even address. Receive Lists must be halfword aligned, the TRM will not alter this parameter.

Programming Notes:

The controller can create a chain comprising a fixed number of Receive Lists, set the Forward Pointer of the last list to the address of the first list, then manipulate the List Valid bit of the Receive CSTAT request to initiate reception.

When the TRM reads a list with the List Valid bit reset to '0', it will suspend processing until a Receive Valid interrupt request is issued by the controller, see MMIO instruction 'Write Interrupt'. The controller is **not** informed of this suspension. The Receive Valid interrupt **must** be issued when changing one or more List Valid bits from '0' to '1' when the list is on the Receive chain.

The Receive Valid interrupt can be issued at any time and the TRM will ignore the interrupt if it is not waiting for a List Valid bit transition.

If the fixed-Receive-chain technique is used and more than one list is used to receive a single frame, caution must be used when validating the lists. Since the TRM does not alter the CSTAT for lists that are not Start of Frame or End of frame, revalidation of the Start of Frame list would also release the lists that are for the middle of the frame.

FRAME SIZE: This is the total number of bytes in the received frame. The TRM will store this count in the Receive List that starts a new frame. Frame Size is not altered by the TRM unless the list starts a new frame. Frame Size includes the header and data field but excludes the frame check and delimiter bytes.

DATA COUNT: This is the number of bytes that can be stored at the address given in the Data Address parameter specified next. There can be a maximum of three Data Count/Data Address pairs for any Receive List. Bit 0 of Data Count must be:

- '1' to indicate there is a following Data Count/Data Address pair.
- '0' to indicate that this Data Count/Data Address is the last in this Receive List.

A Data Count of 0 is allowed. Data Count can be odd or even. The TRM will not alter this parameter.

If the Pad Routing Field is specified in the Open options, then the first Data Count in a Receive List used to receive the start of a frame must be at least 32 so that the full header can be received. If the Data Count is less than 32 in this case then Pad Routing Field will be void.

DATA ADDRESS: This is the address for the received data. Data Address can be odd or even. The TRM will not alter this parameter.

Note:

If the TRM is to write data to an even controller address (due to an odd Data Count for example), it will transfer a single byte then transfer the remaining data at the next odd controller address. Thus it takes two transfer operations to transfer the data.

RECEIVE CSTAT: This is the command status halfword that is set up by the controller when the Receive List is created and overwritten by the TRM to report frame completion.

Receive CSTAT Request: The controller must set the bits of this parameter as follows:

Bit	Meaning
0	List Valid
1	Frame Complete
2	Start of frame
3	End of frame
4	Frame interrupt
5	Interframe wait
6 to 15	Reserved

Bit 0 - List Valid: the TRM will wait for this bit to be '1' before placing data in the current Receive List. The controller must issue a Receive Valid interrupt request when changing List Valid bits from '0' to '1'. This bit is examined in every Receive List.

Bit 1 - Frame complete: this bit should always be reset to '0' for a CSTAT request.

Bit 2 - Start of frame: this bit should always be reset to '0' for a CSTAT request.

Bit 3 - End of frame: this bit should always be reset to '0' for a CSTAT request.

Bit 4 - Frame interrupt: this bit must be set to '1' if you want the TRM to interrupt when a frame has been received. This bit is ignored for a list that does not start a frame.

Bit 5 - Interframe wait: this bit must be set to '1' if you want the TRM to interrupt when a frame has been received **and** you want the TRM to go into a Receive Suspend state. Receiving will not continue until the controller issues a Receive Continue interrupt request. The next list to be processed is the one addressed by the forward pointer of the **last** list with the End of frame bit set to '1'. The Receive Completion bit of the SSB will report Frame Complete. When Interframe Wait bit is set, then Frame Interrupt bit will be ignored. Interframe Wait is ignored for a list that does not start a frame.

Bits 6 to 15 - Reserved: these bits are ignored.

Receive CSTAT Completion: When a frame has been transferred to the controller, the CSTATs for the lists that start and end a frame are updated by the TRM with status information about the frame. The CSTAT for completion is:

Bit	Meaning
0	List Valid
1	Frame complete
2	Start of frame
3	End of frame
4 to 7	Reserved
8	PCFE 0 (ARI)
9	PCFE 1 (FCI)
10	PCFE 2
11	PCFE 3
12	PCFE 4 (ARI)
13	PCFE 5 (FCI)
14	Address Match 0
15	Address Match 1

The significance of these bits is as follows:

Bit 0 - List Valid: this bit will be reset to '0'.

Bit 1 - Frame complete: this bit will be set to '1'.

Bit 2 - Start of frame: when on, this Receive List is the start of frame.

Bit 3 - End of frame: when on, this Receive List is the end of frame.

Bits 4 to 7 - Reserved: these bits will be reset to '0'.

Bits 8 to 13 - Received PCFE: when Start of frame is reset to '0', these bits will also be reset. When Start of frame is set to '1', these bits will contain the high order six bits of the received PCFE.

Bits 14 and 15 - Address match: when Start of frame is reset to '0', these bits will also be reset. When Start of frame is set to '1', these bits will indicate the reason that the To-Address field in the frame header was matched by the TRM.

The Address Match codes are:

Code	Match Reason
00	Node address match
01	Group address match
10	Functional address match
11	Other reason (for example broadcast frame)

System Status Block

SSB Address	Content
+0	X'0006'
+2	Receive completion
+4	List address
+6	List address

RECEIVE COMPLETION The two bytes of Receive Completion report the status of the completed frame, as shown in the following table:

Bit	Meaning
0	Frame complete
1	Receive suspended
2 to 15	Reserved

The bits of Receive completion have the following significance:

Bit 0 - Frame complete: when on, a frame has been received and the Frame Interrupt bit of CSTAT has been set to '1'. Since frames can be received and transferred to the controller faster than the controller can respond to the interrupts and/or faster than the TRM can cause the interrupts, the Frame Complete interrupt can report the completion of more than one frame.

List Address of the SSB will contain the address of the last Receive List of the last frame transferred to the controller.

If lists with Frame Interrupt set are mixed with lists that do not, then Frame Complete can include both types of frame.

This bit will not be set with Receive Suspended.

Bit 1 - Receive suspended: when on, the TRM has detected an odd address in the Forward Pointer of a Receive List. List Address will contain the address of the list that has an odd Forward Pointer. The controller must update the Forward Pointer and issue a Receive Continue interrupt request, using the MMIO Write Interrupt instruction, to continue.

Receive Suspended will not be set with Frame Complete.

Programming Note:

The Receive Continue interrupt request, see MMIO instruction 'Write Interrupt', can be issued at anytime but the TRM will ignore the interrupt if it is not waiting for a Forward Pointer transition from odd to even.

Bits 2 to 15 - Reserved: these bits will be reset to '0'.

Close (X'0007)

This command is used to end communication with the ring, or to stop the Open Wrap command. All frames in the TRM at the time this command is issued will be purged.

System Command Block

The SCB for a Close command is:

SCB Address	Content
+0	X'0007'
+2	Ignored
+4	Ignored

X'0007': is the Close command code.

Ignored: this fullword is reset by the TRM but is ignored for command execution.

System Status Block

When the command completes, the SSB will contain the following:

SSB Address	Content
+0	X'0007'
+2	Close completion

The bits of Close Completion have the following significance:

Bit	Meaning
0	TRM closed
1 to 15	Reserved

Bit 0 - TRM closed: when on, the Close command has completed and the TRM is closed for further operation. An Open command will have to be issued if operation is to continue.

Bits 1 to 15 - Reserved: these bits will be reset to '0'.

Set Group Address (X'0008')

This command is used to change the group address of the TRM after an Open command has been executed. The command will be rejected with Command Reject status if the TRM is not open.

System Command Block

The SCB for command is:

SCB Address	Content
+0	X'0008'
+2	Group address
+4	Group address

X'0008': is the Set Group Address command code.

Group address: this is a fullword parameter giving the new group address for the TRM. Bit 0 of this address is ignored.

System Status Block

When the Set Group Address command completes, the SSB will contain the following:

SSB Address	Content
+0	X'0008'
+2	Set Group completion

The bits of Set Group completion have the following significance:

Bit	Meaning
0	Command complete
1 to 15	Reserved

Bit 0 - Command complete: when on, the Set Group Address command has completed.

Bits 1 to 15 - Reserved: these bits will be reset to '0'.

Set Functional Address (X'0009')

This command is used to change the functional address of the TRM after an Open command has been executed. The command will be rejected with Command Reject status if the TRM is not open.

System Command Block

The SCB for the Set Functional Address command is:

SCB Address	Content
+0	X'0009'
+2	Functional address
+4	Functional address

X'0009': is the Set Functional Address command code.

Functional address: this is a fullword parameter giving the new functional address for the TRM. Bit 0 of this address is ignored.

System Status Block

When the Set Functional Address command completes, the SSB will contain the following:

SSB Address	Content
+0	X'0009'
+2	Set Functional Address completion

The bits of Set Functional Address completion have the following significance:

Bit	Meaning
0	Command complete
1 to 15	Reserved

Bit 0 - Command complete: when on, the Set Functional Address command has completed.

Bits 1 to 15 - Reserved: these bits will be reset to '0'.

Read Error Log (X'000A')

The Read Error Log command is used to read **and reset** the TRM Error Log. After command completion, the Error Log will be all zeros. Each byte of the Error Log contains a count of the number of times that each error has occurred.

System Command Block

The SCB for command is:

SCB Address	Content
+0 +2 +4	X'000A' Address Address

X'000A': is the Read Error Log command code.

Address: is the address the 14-byte error log will be written to.

TRM Error Log

The Error Log is as follows:

Byte	Error
0	Line error
1	Internal error
2	Burst error
3	ARI/FCI error
4	Abort delimiter
5	Reserved
6	Lost frame
7	Receive congestion
8	Frame copied error
9	Frequency error
10	Token error
11	Reserved
12	DMA bus error
13	DMA parity error

Bytes 0 to 11 are isolating/non-isolating error counters, bytes 12 and 13 are DMA controller errors.

System Status Block

When the Read Error Log command completes, the SSB will contain the following:

SSB Address	Content
+0 +2	X'000A' Error Log completion

The bits of Error Log completion have the following significance:

Bit	Meaning
0 1 to 15	Command complete Reserved

Bit 0 - Command complete: when on, the Read Error Log command has completed.

Bits 1 to 15 - Reserved: these bits are reset to '0'.

Read TRM (X'000B')

The Read TRM command is used to transfer the contents of TRM storage to the controller.

System Command Block

The SCB for command is:

SCB Address	Content
+0	X'000B'
+2	Address
+4	Address

X'000B': is the Read TRM command code.

Address: is a fullword pointer to the controller storage area that is to receive the contents of TRM storage. Before the command is executed this controller area must contain the parameters specified in Read TRM Buffer below. The TRM will take these parameters and write the desired contents into this area, overwriting the command parameters.

Read TRM Buffer

The buffer transferred to the controller has the following structure:

Byte	Meaning
0	Data count
2	Data address
4 to n	Data area

DATA COUNT: This halfword parameter specifies the number of bytes to be read from the TRM.

DATA ADDRESS: This halfword contains the address of the data in the TRM to be read. Bit 15 is reset by the TRM to '0'. The address specified is **not** checked for valid extents; if the address is outside the limits of installed storage, a TRM Check error may occur.

TRM Storage

Storage locations are defined by five halfword base pointers. These pointers must be read after initialization has completed with no errors. The base pointers start at location X'0A00'. The pointers are read using the MMIO Read TRM or DMA Read TRM commands. After the pointers have been read, parameters associated with the pointers must be read using the DMA Read TRM command. The TRM will prevent the MMIO Read Data command from reading storage below the address X'0800'.

The pointers and their associated parameters are shown below:

X'200' - pointer to BIA

X'202' - pointer to microcode

X'204' - pointer to TRM addresses, as follows:

Offset	Length	Content
0	6	TRM node address
6	4	TRM group address
10	4	TRM functional address

X'206' - pointer to TRM parameters, as follows:

Offset	Length	Content
0	4	TRM physical address
4	6	Upstream node address
10	4	Upstream physical address
14	6	Last poll address
20	2	Authorized environment
22	2	Transmit access priority
24	2	Source class authorization
26	2	Last attention code
28	6	Last source address
34	2	Last beacon type
36	2	Last major vector
38	2	Ring status
40	2	Soft error timer value
42	2	Front end error counter
44	2	Reserved
46	2	Monitor error code
48	2	Beacon transmit type
50	2	Beacon receive type
52	2	Frame correlator save
54	6	Beaconing station NUAN
60	4	Reserved
64	4	Beaconing station physical address

X'208' - pointer to MAC buffer.

System Status Block

When the command completes, the SSB will contain the following:

SSB Address	Content
+0	X'000B'
+2	Read Completion

The bits of Read completion have the following significance:

Bit	Meaning
0	Command complete
1 to 15	Reserved

Bit 0 - Command complete: when on, the command has completed and the required TRM data has been transferred to the controller.

Bits 1 to 15 - Reserved: these bits will be reset to '0'.

IMPL Enable (X'000C')

This command is used to enable the TRM to process an IMPL Force MAC frame. If this is received after the command has been issued, then:

1. The TRM will be closed.

2. A data byte will be written to a specified location.

3. The controller will be interrupted with a specified interrupt vector.

The TRM will then be in the same state as after initialization so the Open command will have to be reissued. IMPL Enable command will have to be reissued if it is required.

If this command has not been issued then this frame will be rejected.

System Command Block

The SCB for IMPL Enable command is:

SCB Address	Content
+0	X'000C'
+2	Data | Vector
+4	System address

X'000C': is the IMPL Enable command code.

Data: this byte will be written to the controller at the address specified by System Address.

Vector: this byte specifies the interrupt vector to be used when the IMPL Force interrupt code is generated. When both Data and Vector are X'00', the IMPL Force procedure will be disabled.

System address: the Data byte will be written to the controller location specified by this parameter. The high-order byte of the address must be X'00'. The maximum value allowed for this parameter is 64k. If System Address is X'0000', the data will not be written.

System Status Block

When the command completes, the SSB will contain the following:

SSB Address	Content
+0	X'000C'
+2	IMPL Completion

The bits of IMPL Completion have the following significance:

Bit	Meaning
0	Command complete
1 to 15	Reserved

Bit 0 - Command complete: when on, the IMPL command has completed.

Bit 1 to 15 - Reserved: these bits will be reset to '0'.

APPENDIX A. CCU EXTERNAL REGISTERS

Input/Output X'00' through X'27' (General Registers)

The bit assignments of these registers are, in general, not fixed, but vary with the use of the register. There is however one exception: the first register of each group always contains the address of the next sequential instruction in that interrupt level. Note that these registers are implemented in hardware in the 3725, but mapped into local storage in the 3720/21. Their use, however, is the same for both controllers.

Input/Output X'28' through X'2F' (Reserved)

The 8 registers addressed by these instructions are reserved.

Input/Output X'30' through X'35' (Cycle Steal Address Registers)

Register	Channel adapter
X'30'	1
X'31'	2
X'32'	3
X'33'	4
X'34'	5
X'35'	6

Input/Output X'36' through X'3E' (Pointer Registers)

Input/Output X'3F' (Communication Scanner CS Address)

Input/Output X'40' through X'43' (Interrupt Start Address)

Input/Output X'44' (Byte Operations Base Register)

Input/Output X'45' (Halfword Operations Base Register)

Input/Output X'46' (Fullword Operations Base Register)

Input/Output X'48' (IOH Address Substitution Register)

Input/Output X'49' through X'4F' (Reserved)

Input/Output X'50' through X'5F' (Programmable Registers)

Input/Output X'60' through X'67' (Reserved)

Input X'68' (Zero Register)

Input/Output X'69' through X'6F' (Reserved)

Input X'70' (Storage Size Installed)

Byte	Bit	Meaning
0	0	0 (For 3725 only. For 3720/3721 see below)
	1	1 (For 3725 only. For 3720/3721 see below)
	2	0
	3	2048 K
	4	1024 K
	5	512 K
	6	256 K
	7	Storage not a multiple of 256K
1	0-7	0

3720/3721:

Byte 0 Bits 0 1	Meaning
0 0	All cards 512k
0 1	Reserved
1 0	Reserved
1 1	Reserved

Output X'70' (Hardstop)

Input X'71' (Operator Address/Data Entry Register)

Byte	Bit	Meaning
X	2-7	Operator address/data register byte X, bits 2-7
0	0-7	Operator address/data register byte 0, bits 0-7
1	0-7	Operator address/data register byte 1, bits 0-7

Output X'71' (Display Register 1)

Byte	Bit	Meaning
X	2-7	Display register 1 byte X, bits 2-7
0	0-7	Display register 1 byte 0, bits 0-7
1	0-7	Display register 1 byte 1, bits 0-7

Input X'72' (Operator Display/Function Select Control)

Byte	Bit	Meaning
0	0	Function select 8
	1	Function select 9
	2	Function select 10
	3	Function select 11 (storage address)
	4	Function select 12 (register address)
	5	Function select 13
	6	Function select 14
	7	Function select 15
1	0	Function select 16
	1	Function select 1
	2	Function select 2
	3	Function select 3
	4	Function select 4
	5	Function select 5
	6	Function select 6
	7	Function select 7

Output X'72' (Display Register 2)

Byte	Bit	Meaning
X	2-7	Display register 2 byte X, bits 2-7
0	0-7	Display register 2 byte 0, bits 0-7
1	0-7	Display register 2 byte 1, bits 0-7

Input X'73' (Insert Storage Protect/Address Exception Key)

Byte	Bit	Meaning
0	0-7	(not used)
1	0	(not used)
	1	(not used)
	2	(not used)
	3	(not used)
	4	(not used)
	5	Key Bit 0
	6	Key Bit 1
	7	Key Bit 2

Output X'73' (Set Storage Protect/Address Exception Key)

Byte	Bit	Meaning
X	2	Storage key address bit 0
	3	Storage key address bit 1
	4	Storage key address bit 2
	5	Storage key address bit 3
	6	Storage key address bit 4
	7	Storage key address bit 5
0	0	Storage key address bit 6
	1	Storage key address bit 7
	2	Storage key address bit 8
	3	Storage key address bit 9/user key address bit 0
	4	Storage key address bit 10/user key address bit 1
	5	User key address bit 2
	6	User key address bit 3
	7	User key address bit 4
1	0	(not used)
	1	Enable storage protect/address exception
	2	Key type bit 0
	3	Key type bit 1
	4	Modify key value
	5	Key bit 0
	6	Key bit 1
	7	Key bit 2

Input X'74' (Lagging Address Register)

Input X'75' (CCW for AIO Operations)

Byte	Bit	Meaning
0	0	CCW Bit 5 (0 = CA AIO, 1 = scanner AIO)
	1	CCW Bit 11 (pointer no./scanner address bit 0)
	2	CCW Bit 12 (pointer no./scanner address bit 1)
	3	CCW Bit 13 (pointer no./scanner address bit 2)
	4	CCW Bit 14 (pointer no./scanner address bit 3)
	5	(not used)
	6	(not used)
	7	(not used)
1	0-7	(not used)

Input X'76' (Adapter Level 1 Interrupt Requests)

Byte	Bit	Meaning
0	0	Addressing exception during I/O operations
	1	Storage protection check during I/O operations
	2	Invalid CCW during I/O operations
	3	(not used)
	4	Time out condition
	5	Bus in parity check
	6	Adapter initiated operation
	7	MOSS initiated operation
1	0-7	(not used)

Output X'76' (Miscellaneous Control 1)

Byte	Bit	Meaning
0	0	Reset errors detected during I/O operations
	1	(not used)
	2	(not used)
	3	Control program to MOSS request
	4	Control program to MOSS response
	5	(not used)
	6	(not used)
	7	(not used)
1	0-7	(not used)

Input X'77' (Adapter Levels 2 and 3 Interrupt Requests)

Byte	Bit	Meaning
0	0	(not used)
	1	Scanner level 2 interrupt
	2	(not used)
	3	(not used)
	4	(not used)
	5	(not used)
	6	(not used)
	7	(not used)
1	0	Level 3 channel adapter interrupt
	1	(not used)
	2	(not used)
	3	(not used)
	4	(not used)
	5	(not used)
	6	(not used)
	7	(not used)

Output X'77' (Miscellaneous Control 2)

Byte	Bit	Meaning
0	0	Reset IPL level 1 interrupt
	1	Reset CCU hardware checks
	2	Reset MOSS panel interrupt request level 3
	3	Reset MOSS diagnostic interrupt request level 3
	4	Reset MOSS service interrupt request level 4
	5	Reset MOSS service interrupt response level 4
	6	(not used)
	7	Reset program controlled interrupt level 2
1	0	Reset MOSS inoperative level 1 interrupt
	1	Reset interval timer level 3 interrupt
	2	Reset program controlled interrupt level 3
	3	Reset MOSS diagnostic interrupt request level 2
	4	Reset address compare level 1 interrupt
	5	Reset software checks
	6	Reset program controlled interrupt level 4
	7	Reset supervisor call level 4 interrupt

Output X'78' (Force ALU Checks)

Input X'79' (Utility)

Byte	Bit	Meaning
0	0	(not used)
	1	Probe condition satisfied
	2	Probe address compare received
	3	(not used)
	4	(not used)
	5	(not used)
	6	Program level 5 C latch
	7	Program level 5 Z latch
1	0	Program level 2 interrupted by level 1
	1	Program level 3 interrupted by level 1
	2	Program level 4 interrupted by level 1
	3	Program level 5 interrupted by level 1
	4	(not used)
	5	(not used)
	6	(not used)
	7	(not used)

Output X'79' (Utility)

Byte	Bit	Meaning
0	0	(not used)
	1	(not used)
	2	Set programmed IPL request
	3	(not used)
	4	Remote power off
	5	Inhibit program level 5C, 5Z replace
	6	Set program level 5 C latch
	7	Set program level 5 Z latch
1	0	(not used)
	1	(not used)
	2	Set AIO stop mode
	3	Reset AIO stop mode
	4	Set bypass CCU check stop mode
	5	Reset bypass CCU check stop mode
	6	Scope sync pulse 1
	7	Scope sync pulse 2

Input X'7A' (High Resolution Timer/Utilization Counter)

Byte	Bit	Meaning
X	2	Timer Bit 0
	3	Timer Bit 1
	4	Timer Bit 2
	5	Timer Bit 3
	6	Timer Bit 4
	7	Timer Bit 5
0	0	Timer Bit 6
	1	Timer Bit 7
	2	Timer Bit 8
	3	Timer Bit 9
	4	Timer Bit 10
	5	Timer Bit 11
	6	Timer Bit 12
	7	Timer Bit 13
1	0	Timer Bit 14
	1	Timer Bit 15
	2	Timer Bit 16
	3	Timer Bit 17
	4	Timer Bit 18
	5	Timer Bit 19
	6	Timer Bit 20
	7	Timer Bit 21

Output X'7A' (High Resolution Timer/Utilization Counter Control)

Byte	Bit	Meaning
0	0	Timer/counter (1 = reset timer/enable count)
	1	High/low resolution (1 = low resolution)
	2	Timer/utilization counter (0 = timer)
	3	(not used)
	4	(not used)
	5	(not used)
	6	(not used)
	7	(not used)
1	0-7	(not used)

Input X'7B' (Branch Trace Address Pointer)

Byte	Bit	Meaning
X	2-6	Branch trace address pointer byte X, bits 2-7
0	0-7	Branch trace address pointer byte 0, bits 0-7
1	0-7	Branch trace address pointer byte 1, bits 0-7

Output X'7B' (Set PCI Level 2)

Input X'7C' (Branch Trace Buffer Count)

Byte	Bit	Meaning
0	0	Branch trace buffer count bit 0
	1	Branch trace buffer count bit 1
	2	Branch trace buffer count bit 2
	3	Branch trace buffer count bit 3
	4	Branch trace buffer count bit 4
	5	Branch trace buffer count bit 5
	6	Branch trace buffer count bit 6
	7	Branch trace buffer count bit 7
1	0	Branch trace buffer count bit 8
	1	Branch trace buffer count bit 9
	2	Branch trace buffer count bit 10
	3	Branch trace buffer count bit 11
	4	Branch trace buffer count bit 12
	5	(not used)
	6	(not used)
	7	(not used)

Output X'7C' (Set PCI Level 3)

Input X'7D' (CCU Hardware Check Register)

Output X'7D' (Set PCI Level 4)

Input X'7E' (CCU Level 1 Interrupt Requests)

Byte	Bit	Meaning
0	0	MOSS inoperative
	1	CCU hardware error summary
	2	(not used)
	3	Level 5 I/O error
	4	Invalid operation
	5	Adapter level 1 interrupt request
	6	(not used)
	7	CCU level 1 interrupts during I/O summary
1	0	Address compare level 1 interrupt
	1	Addressing exception on instruction fetch
	2	Storage protect exception on instruction fetch
	3	Addressing exception on program execution
	4	Storage protect exception on program execution
	5	(not used)
	6	IPL level 1 interrupt
	7	(not used)

Output X'7E' (Set Program Interrupt Mask Bits)

Byte	Bit	Meaning
0	0-7	(not used)
1	0	(not used)
	1	Mask adapter program level 1 requests
	2	Mask program level 2 requests
	3	Mask program level 3 requests
	4	Mask program level 4 requests
	5	Mask program level 5 execution
	6	(not used)
	7	(not used)

Input X'7F' (CCU L2, 3, or 4 Interrupt Requests)

Byte	Bit	Meaning
0	0	Program controlled interrupt (PCI) level 2
	1	MOSS diagnostic interrupt request level 2
	2	MOSS diagnostic interrupt request level 3
	3	MOSS service interrupt request level 4
	4	MOSS service interrupt response level 4
	5	(not used)
	6	CE/operator interrupt request level 3
	7	Program controlled interrupt (PCI) level 4
1	0	(not used)
	1	(not used)
	2	(not used)
	3	(not used)
	4	(not used)
	5	Interval timer interrupt request level 3
	6	Program controlled interrupt (PCI) level 3
	7	Supervisor call level 4

Output X'7F' (Reset Program Interrupt Mask Bits)

Byte	Bit	Meaning
0	0-7	(not used)
1	0	(not used)
	1	Unmask adapter program level 1 requests
	2	Unmask program level 2 requests
	3	Unmask program level 3 requests
	4	Unmask program level 4 requests
	5	Unmask program level 5 execution
	6	(not used)
	7	(not used)

APPENDIX B. CA INPUT/OUTPUT INSTRUCTION SUMMARY CHARTS

In the 3725, the registers are implemented in hardware. In the 372021, these registers are mapped into local storage. Their use, however is the same for both controllers.

Hardware Status Byte Register

Bit	Meaning
0	Attention
1	Status Modifier
2	Control Unit End
3	Busy
4	Channel End
5	Device End
6	Unit Check
7	Unit Exception

Input X'0' (Initial Selection Control Register)

Byte	Bit	Meaning
0	0	Initial selection interrupt
	1	Interface disconnect
	2	Selective reset
	3	Channel bus out check
	4	Emulation subchannel operation
	5	Stacked initial status
	6	Status byte cleared
	7	System reset
1	0-7	(not used)

Output X'0' (Reset Initial Selection)

Input X'1' (Initial Selection Address and Command Register)

Byte	Bit	Meaning
0	0-7	Address byte bit 0-7 (initial selection address)
1	0-7	I/O cmnd byte bit 0-7 (initial selection cmnd)

Output X'1' (Initial Selection Address and Command Register)

Byte	Bit	Meaning
0	0-7	Address byte bit 0-7 (initial selection address)
1	0-7	I/O cmnd byte bit 0-7 (initial selection cmnd)

Input X'2' (Data/Status Control Register)

Byte	Bit	Meaning
0	0	Outbound data transfer sequence
	1	Inbound data transfer sequence
	2	Status transfer sequence
	3	NCP subchannel if 0; EP subchannel if 1
	4	Channel end presented
	5	Channel stop/interface disconnect
	6	Suppress out monitor interrupt
	7	Program requested interrupt
1	0	Channel bus out check
	1	Selective reset
	2	Suppress out
	3	Stacked ending status
	4	Priority outbound service
	5	Residual byte count bit 5
	6	Residual byte count bit 6
	7	Residual byte count bit 7

Output X'2' (Data/Status Control Register)

Byte	Bit	Meaning
0	0	Set/reset outbound data transfer sequence (Note)
	1	Set/reset inbound data transfer sequence (Note)
	2	Set/reset status transfer sequence (Note)
	3	Set/reset ESC operation (Note)
	4	Set/reset PIO mode (Note)
	5	Reset initial selection interrupt
	6	Reset data/status interrupt
	7	(not used)
1	0	Set monitor for circle B
	1	(not used)
	2	Set monitor for 2848 ETX
	3	Set suppressible status
	4	Priority outbound service
	5	Request byte count bit 5
	6	Request byte count bit 6
	7	Request byte count bit 7

Note: Set = 1; reset = 0.

Input/Output X'3' (ESC Address and Status Byte Register)

Byte	Bit	Meaning
0	0-7	Address byte bits 0-7 (data/status transfer)
1	0	ESC status byte bit 0 (attention)
	1	ESC status byte bit 1 (status modifier)
	2	ESC status byte bit 2 (control unit end)
	3	ESC status byte bit 3 (Busy)
	4	ESC status byte bit 4 (channel end)
	5	ESC status byte bit 5 (device end)
	6	ESC status byte bit 6 (unit check)
	7	ESC status byte bit 7 (unit exception)

Input/Output X'4' and X'5' (Data Buffer Registers)

Register X'4' (Data Buffer Bytes 1 and 2 or 5 and 6)

Byte	Bit	Meaning
0	0-7	Data buffer byte 1 or 5 bits 0-7
1	0-7	Data buffer byte 2 or 6 bits 0-7

Register X'5' (Data Buffer Bytes 3 and 4 or 7 and 8)

Byte	Bit	Meaning
0	0-7	Data buffer byte 3 or 7 bits 0-7
1	0-7	Data buffer byte 4 or 8 bits 0-7

Input X'6' (NSC Status/Control Register)

Byte	Bit	Meaning
0	0	Channel adapter switched to interface B
	1	Channel adapter switched to interface A
	2	(not used)
	3	(not used)
	4	(not used)
	5	(not used)
	6	(not used)
	7	(not used)
1	0	NSC status byte bit 0 (attention)
	1	NSC status byte bit 1 (status modifier)
	2	NSC status byte bit 2 (control unit end)
	3	NSC status byte bit 3 (Busy)
	4	NSC status byte bit 4 (channel end)
	5	NSC status byte bit 5 (device end)
	6	NSC status byte bit 6 (unit check)
	7	NSC status byte bit 7 (unit exception)

Output X'6' (NSC Status/Control Register)

Byte	Bit	Meaning
0	0	Set force A busy
	1	Set force B busy
	2	Force error
	3	Diagnostic storage mode (set = 1; reset = 0)
	4	(not used)
	5	Check the checkers
	6	A/B data buffer diagnostic mode
	7	Reset to neutral state
1	0	Set NSC status byte bit 0 (attention)
	1	Set NSC status byte bit 1 (status modifier)
	2	Set NSC status byte bit 2 (control unit end)
	3	Set NSC status byte bit 3 (busy)
	4	Set NSC status byte bit 4 (channel end)
	5	Set NSC status byte bit 5 (device end)
	6	Set NSC status byte bit 6 (unit check)
	7	Set NSC status byte bit 7 (unit exception)

Input X'7' (Channel Adapter Condition Register)

Byte	Bit	Meaning
0	0	CA5 enabled
	1	(not used)
	2	CA6 enabled
	3	(not used)
	4	(not used)
	5	NSC address active
	6	PIO mode
	7	(not used)
1	0	CA1 interface A enabled
	1	CA1 interface B enabled
	2	CA2 interface A enabled
	3	CA2 interface B enabled
	4	CA3 interface A enabled
	5	CA3 interface B enabled
	6	CA4 interface A enabled
	7	CA4 interface B enabled

Output X'7' (Channel Adapter Control Register)

Byte	Bit	Meaning
0	0	Enable auto-select) cannot be on at
	1	Disable auto-select) the same time
	2	Select CA addressed by bits 4-6
	3	Execute output on CA addressed by bits 4-6
	4	Channel address bit 0)
	5	Channel address bit 1) CA address 1-6
	6	Channel address bit 2)
	7	Channel adapter reset
1	0	Set suppress out monitor
	1	Set program requested interrupt
	2	Reset channel adapter interrupt level 1 checks
	3	Reset system reset/NSC address active
	4	Set allow channel interface enable (A and B)
	5	Set ESC operational
	6	Set ESC command free
	7	Set allow channel interface disable (A and B)

Note: Bits 4 and 7 cannot be active at the same time.

Input/Output X'B' (ESC Test I/O Address and Status Register)

Byte	Bit	Meaning
0	0-7	ESC TIO address byte bits 0-7
1	0	ESC TIO status byte bit 0 (attention)
	1	ESC TIO status byte bit 1 (status modifier)
	2	ESC TIO status byte bit 2 (control unit end)
	3	ESC TIO status byte bit 3 (Busy)
	4	ESC TIO status byte bit 4 (channel end)
	5	ESC TIO status byte bit 5 (device end)
	6	ESC TIO status byte bit 6 (unit check)
	7	ESC TIO status byte bit 7 (unit exception)

Input X'C' (Cycle Steal Mode Control Register)

Byte	Bit	Meaning
0	0	SYN monitor latch
	1	DLE temporary latch
	2	USASCII monitor control latch
	3	EBCDIC monitor control latch
	4	(not used)
	5	(not used)
	6	(not used)
	7	(not used)
1	0-7	Residual byte count bits 0-7

Output X'C' (Cycle Steal Mode Control Register)

Byte	Bit	Meaning
0	0	SYN monitor control latch (Note)
	1	DLE remember control latch (Note)
	2	USASCII monitor control latch (Note)
	3	EBCDIC monitor control latch (Note)
	4	(not used)
	5	(not used)
	6	(not used)
	7	(not used)
1	0-7	Residual byte count bits 0-7

Note: 1 = set; 0 = reset

Input X'D' (Channel Adapter Level 1 Interrupt Check Register)

Byte	Bit	Meaning
0	0	PIO bus parity error
	1	Internal bus parity error
	2	CCU interconnect card check
	3	(not used)
	4	Channel interface card check
	5	Address compare error
	6	Initiate service latch ungated
	7	(not used)
1	0	Output exception check
	1	PIO halt remember latch
	2	Cycle steal halt remember latch
	3	Bus in check interface A
	4	Ground fault error
	5	Bus in check interface B
	6	Driver/receiver card check I/F A
	7	Driver/receiver card check I/F B

Input X'E' (Channel Adapter Level 1 Interrupt Requests)

Byte	Bit	Meaning
0	0	Channel adapter 5 level 1 interrupt request
	1	(not used)
	2	Channel adapter 6 level 1 interrupt request
	3	Channel adapter (any) level 1 interrupt request
	4	Channel address bit 0)
	5	Channel address bit 1) CA address 1-6
	6	Channel address bit 2)
	7	(not used)
1	0	Channel adapter 1 level 1 interrupt request
	1	(not used)
	2	Channel adapter 2 level 1 interrupt request
	3	(not used)
	4	Channel adapter 3 level 1 interrupt request
	5	(not used)
	6	Channel adapter 4 level 1 interrupt request
	7	(not used)

Input X'F' (Channel Adapter Level 3 Interrupt Requests)

Byte	Bit	Meaning
0	0 1 2 3 4 5 6 7	(not used) Two processor switch installed Selected CA initial selection L3 request Selected CA data/status L3 request pending Channel address bit 0) Channel address bit 1) CA address 1-6 Channel address bit 2) (not used)
1	0-7	(not used)

APPENDIX C. COMMUNICATION SCANNER COMMANDS

Grouped by Function

Common Commands:

Command	Hex
Set Mode	X'01'
Enable	X'02'
Disable	X'03'
Monitor Incoming Call	X'04'
Dial	X'05'
Change	X'06'
Wrap	X'07'
Raise DTR	X'08'
Flush Data	X'09'
Reset-D	X'0B'
Reset-N	X'0C'
Halt	X'F0'
Halt Immediate	X'F1'

NCP Commands:

Command	Hex
SDLC Transmit Control	X'10'
SDLC Transmit Data	X'11'
SDLC Transmit Continue	X'1D'
SDLC Receive Monitor	X'12'
SDLC Receive	X'13'
SDLC Receive Continue	X'14'
X.21 Call Request	X'15'
X.21 Monitor Incoming Call	X'16'
X.21 Clear Request	X'17'
NCP BSC Control	X'18'
NCP BSC Transmit	X'19'
NCP BSC Transmit Continue	X'1A'
NCP BSC Receive	X'1B'
NCP BSC Receive Continue	X'1C'

EP Commands:

Command	Hex
EP BSC Transmit Initial	X'20'
EP BSC Transmit SYN	X'21'
EP BSC Transmit Data	X'22'
EP BSC Poll	X'23'
EP BSC Receive	X'24'
EP BSC Receive Continue	X'25'
EP BSC Prepare	X'26'
EP BSC Monitor for Phase	X'27'
EP BSC Address Prepare	X'28'
EP BSC Search	X'29'

Character Mode Commands:

Command	Hex
Write ICW (1-byte transfer)	X'40'
Start-Stop Transfer (4-byte burst)	X'41'
Read ICW	X'F2'

Miscellaneous Commands:

Command	Hex
IBM 386X/58xx Test	X'2B'
Trace	X'2C'
Stop Trace	X'2D'
Line wrap	X'2E'

COMMANDS IN NUMERICAL ORDER

Hex	Command
X'01'	Set Mode
X'02'	Enable
X'03'	Disable
X'04'	Monitor Incoming Call
X'05'	Dial
X'06'	Change
X'07'	Wrap
X'08'	Raise DTR
X'09'	Flush Data
X'0B'	Reset-D
X'0C'	Reset-N
X'10'	SDLC Transmit Control
X'11'	SDLC Transmit Data
X'12'	SDLC Receive Monitor
X'13'	SDLC Receive
X'14'	SDLC Receive Continue
X'15'	X.21 Call Request
X'16'	X.21 Monitor Incoming Call
X'17'	X.21 Clear Request
X'18'	NCP BSC Control
X'19'	NCP BSC Transmit
X'1A'	NCP BSC Transmit Continue
X'1B'	NCP BSC Receive
X'1C'	NCP BSC Receive Continue
X'1D'	SDLC Transmit Continue
X'20'	EP BSC Transmit Initial
X'21'	EP BSC Transmit SYN
X'22'	EP BSC Transmit Data
X'23'	EP BSC Poll
X'24'	EP BSC Receive
X'25'	EP BSC Receive Continue
X'26'	EP BSC Prepare
X'27'	EP BSC Monitor for Phase
X'28'	EP BSC Address Prepare
X'29'	EP BSC Search
X'2B'	IBM 386X/58XX Test
X'2C'	Trace
X'2D'	Stop Trace
X'2E'	Line Wrap
X'40'	Write ICW
X'41'	Start/Stop Transfer
X'F0'	Halt
X'F1'	Halt Immediate
X'F3'	Read ICW

APPENDIX D. MOSS COMMANDS

MAILBOX OUT COMMANDS

Mailbox In commands always have the high order command bit **off**. The hexadecimal values of these commands are therefore always in the range X'0x' through X'7x'.

Command	Hex
Transfer Path Information Unit Out Command (SNA Only)	X'06'
Box Error Records Command	X'07'
Buffers Now Available Command	X'08'
Wrap Test Results Command	X'09'
Time/Date Valid Command	X'0C'
Control Program Parameters Command	X'23'
Request Hardware Configuration Data File Command	X'24'
Control Program Initialization Complete Command	X'25'
Control Program Loaded Command	X'41'
Roll In Saved Storage For Dump Command	X'42'

Mailbox In Commands (MOSS to CCU)

Mailbox In commands always have the high order command bit **on**. The hexadecimal values of these commands are therefore always in the range X'8x' through X'Fx'.

Command	Hex
Transfer Path Information Unit In Command (SNA Only)	X'86'
Wrap Test Request Command	X'89'
Connect Scanner Command	X'8D'
Request Buffer Command	X'8E'
Free Buffer Command	X'8F'
MOSS Offline Command	X'90'
MOSS Online Command	X'91'
Control Program Parameters Saved Command	X'A3'
Configuration Data File Information Available Command	X'A4'
Scanner IML Complete to Load/Dump Command	X'C1'
Roll In Complete to Load/Dump Command	X'C2'

APPENDIX E. REDRIVE LOGIC

The redrive logic is logically situated between the internal logic and the channel adapters or communication scanners. The individual redrives are addressed by the Primary Redrive Address (PRA) and Secondary Redrive Address (SRA) fields contained in register R2 of an IOH instruction, or in the second halfword of an IOHI instruction. The chart below and Figure E-1 on page E-2 for the adapter associated with each redrive and the PRA/SRA bit structure required to address the redrives.

All 3725 configurations except 3725 Model 2

PRA (Byte 0 bits)			SRA (Byte 1 bits)			
5	6	7	4	5	6	Controlled Adapters
0	0	0	0	0	0	Channel Adapter 1 and Line Attachment Base 1
0	0	0	0	0	1	Channel Adapter 2 and Line Attachment Base 2
0	0	0	0	1	0	Line Attachment Base 3
0	0	0	0	1	1	Channel Adapters 3, 4, 5, and 6
0	0	1	0	0	0	Frame Redrive for LABs 4 through 8
0	0	1	0	1	1	Line Attachment Base 4
0	0	1	1	0	0	Line Attachment Base 5
0	0	1	1	0	1	Line Attachment Base 6
0	0	1	1	1	0	Line Attachment Base 7
0	0	1	1	1	1	Line Attachment Base 8

where:

PRA = primary redrive address

SRA = secondary redrive address

3725 Model 2 only

PRA (Byte 0 bits)			SRA (Byte 1 bits)			
5	6	7	4	5	6	Controlled Adapters
0	0	0	0	0	0	Channel Adapters 1/2 and Line Attachment Base
0	0	0	0	0	1	Channel Adapters 3/4 and Line Attachment Base

where:

PRA = primary redrive address

SRA = secondary redrive address

3720/3721

PRA (Byte 0 bits)			SRA (Byte 1 bits)			
5	6	7	4	5	6	Controlled Adapters
0	0	0	0	0	0	3720 Models 1 and 2 (Basic frame)
0	0	0	0	0	1	3721 Models 1 and 2 (Expansion frame)
0	0	0	0	1	1	3720 Model 1 and Channel Adapters 1/2

where:

PRA = primary redrive address

SRA = secondary redrive address

The following diagrams should make this clear:

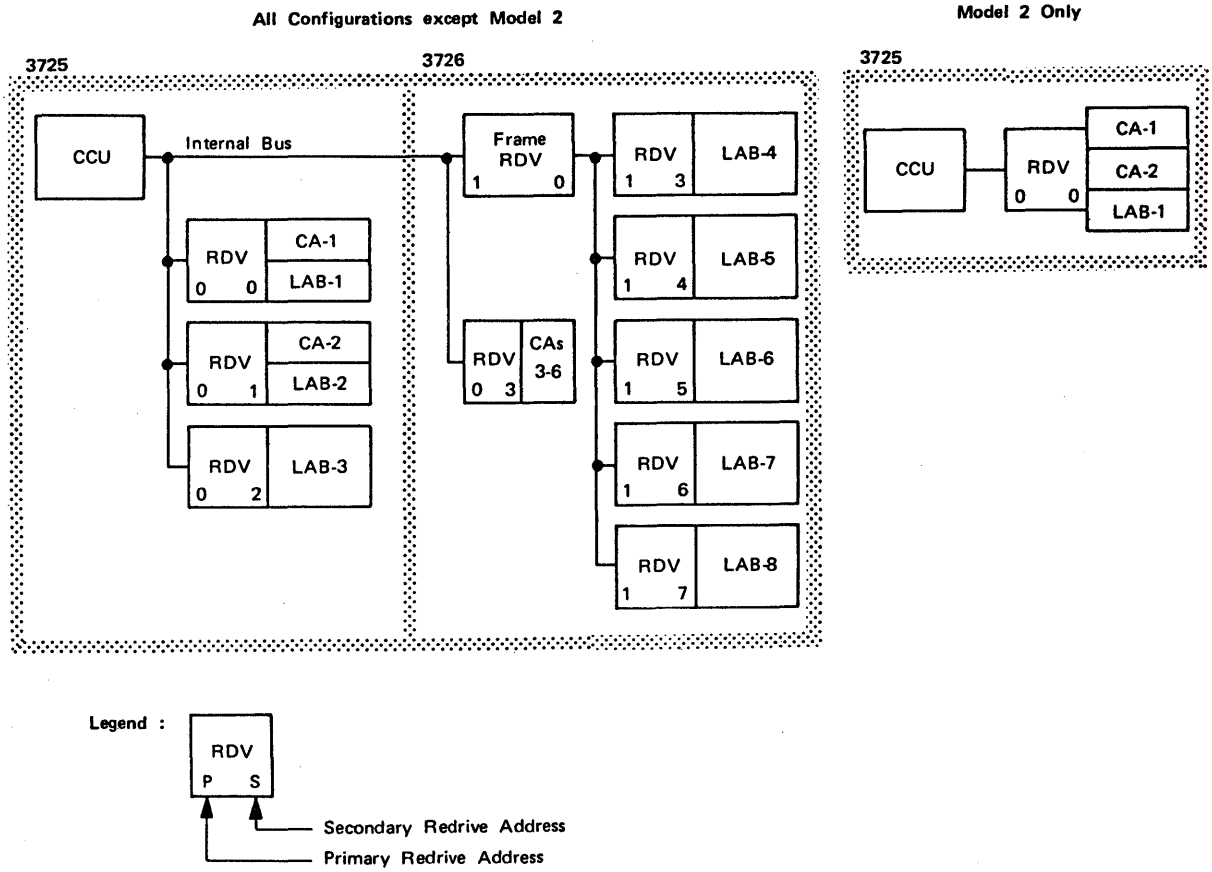
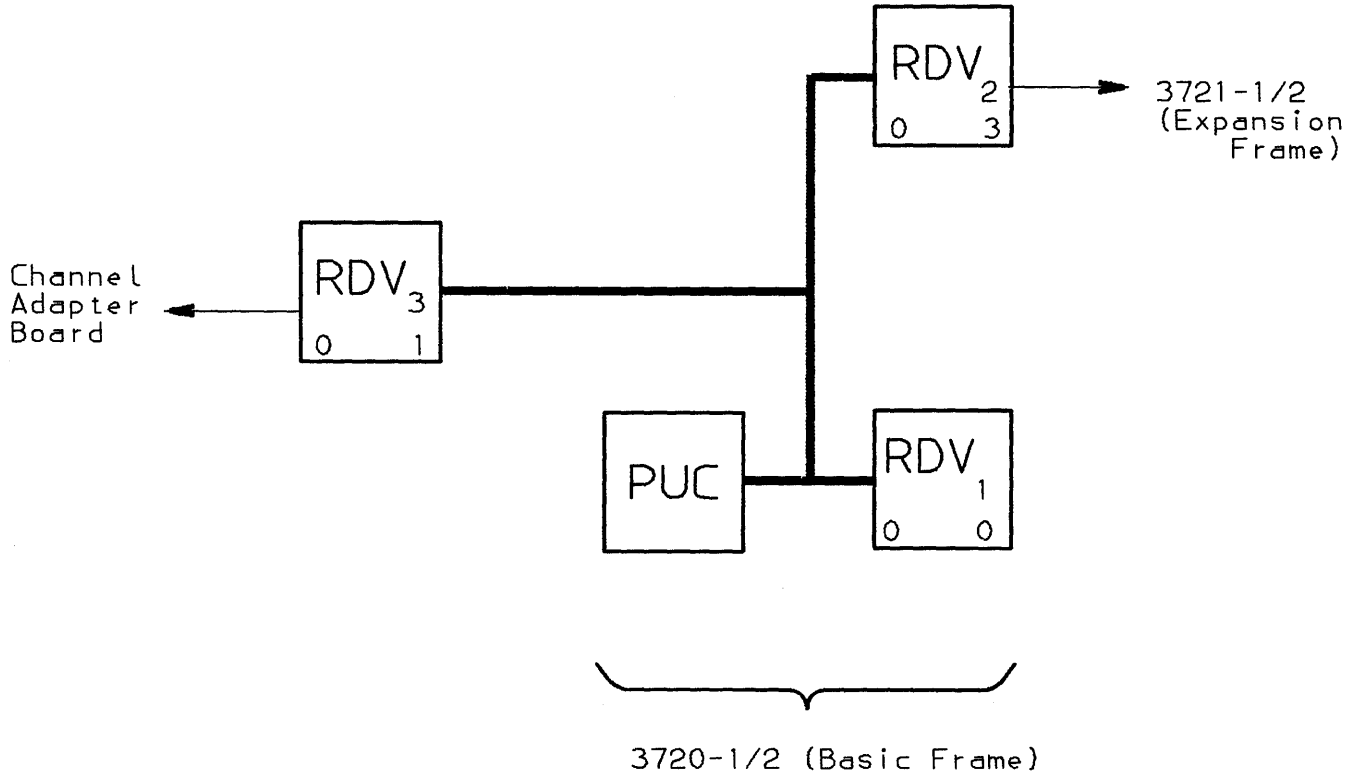


Figure E-1. 3725 Redrive Logic



| Figure E-2. 3720/3721 Redrive Logic

REDRIVE IOH/IOHI INSTRUCTIONS

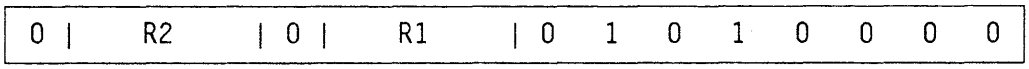
The redrive IOH/IOHI instructions are used to transfer the contents of one of the general registers to a selected redrive register or vice versa. There are two types of redrive input/output instruction:

- Adapter Input/Output (IOH)
- Adapter Input/Output Immediate (IOHI)

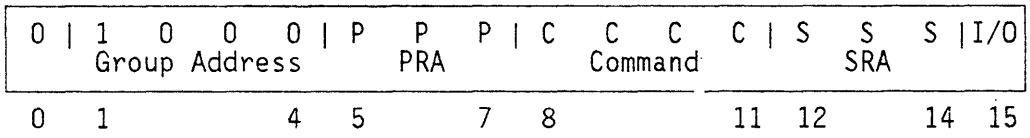
They are used as follows:

Adapter Input/Output (IOH)

This instruction transfers the contents of the register specified by R1 to the redrive logic, or places information coming from the redrive logic into the register specified by R1. The redrive address, the command, and the direction of data movement are all specified by the contents of R2.



R2 must be loaded as follows:



Bits 1 through 4 contain the group address (always 1 0 0 0)

Bits 5 through 7 contain the primary redrive address (PRA)

Bits 8 through 11 indicate the redrive command

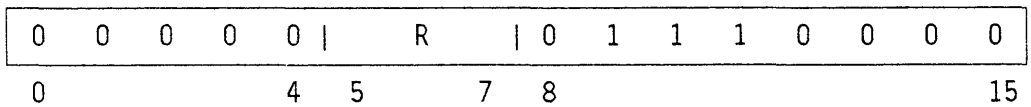
Bits 12 through 14 contain the secondary redrive address (SRA)

I/O = input/output bit: 0 = output, 1 = input

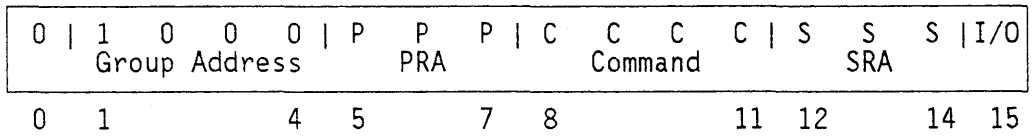
Adapter Input/Output Immediate

This instruction transfers the contents of the register specified by R to the redrive logic, or places information coming from the redrive logic into the register specified by R. The redrive address, the command, and the direction of data movement are all specified by the contents of the second halfword.

First halfword



Second halfword



Bits 1 through 4 contain the group address (always 1 0 0 0)

Bits 4 through 7 contain the primary redrive address (PRA)

Bits 8 through 11 indicate the redrive command

Bits 12 through 14 contain the secondary redrive address (SRA)

I/O = input/output bit: 0 = output, 1 = input

REDRIVE COMMANDS - DETAILED BIT STRUCTURE

The high order bit of the redrive command indicates whether the command is broadcast (high order bit = 0) or addressed to a particular redrive address.

Command Input X'0' (Poll)

The high order bit of the command is always off, indicating a broadcast command. The register addressed by the instruction is loaded from the redrive logic and contains the redrive address and the event that set the redrive level 1 interrupt. If more than one redrive has a level 1 interrupt pending, multiple Input X'0's must be executed, since the redrive logically closer to the CCU on the internal bus presents its poll data first. The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	1
	1	Enable/disable (enable = 0, disable = 1)
	2	PRA Bit 4
	3	PRA Bit 2
	4	PRA Bit 1
	5	SRA Bit 4
	6	SRA Bit 2
	7	SRA Bit 1
1	0	In Bus Parity Error
	1	Out Bus Parity Error
	2	Out Bus Tag Check
	3	In Bus Tag Check
	4	Halt Remember
	5	Secondary Select Out
	6	Secondary Cycle Steal Grant
	7	Command Reject

The bits of byte 0 have the following meaning:

Byte 0, Bit 1 - Enable/disable: this bit, if off, indicates that the redrive address in byte 0, bits 2 through 7 is enabled; if on, it indicates that the redrive address is disabled.

Byte 0, Bits 2 through 4 - Primary Redrive Address (PRA).

Byte 0, Bits 5 through 7 - Secondary Redrive Address (SRA).

Redrive Error Register: Byte 1 is collectively called the Redrive Error Register. Its bits have the following meaning:

Byte 1, Bit 0 - In Bus Parity Error: this bit, if on, indicates that a parity error has occurred on the In Bus between the adapter and the redrive logic.

Byte 1, Bit 1 - Out Bus Parity Error: this bit, if on, indicates that a parity error has occurred on the Out Bus between the adapter and the redrive logic.

Byte 1, Bit 2 - Out Bus Tag Check: this bit, if on, indicates that a tag check (incorrect sequence of tags) has occurred on the Out Bus between the adapter and the redrive logic.

Byte 1, Bit 3 - In Bus Tag Check: this bit, if on, indicates that a tag check (incorrect sequence of tags) has occurred on the In Bus between the adapter and the redrive logic.

Byte 1, Bit 4 - Halt Remember: this bit, if on, indicates that the redrive logic has been selected, but a Halt signal was received before the operation ended.

Byte 1, Bit 5 - Secondary Select Out: this bit, if off, indicates that the Select Out tag was propagated to the next redrive address; if on, it indicates that the tag was not propagated.

Note: The bit is set when the select out tag is sent to the adapter controlled by the redrive logic, and is reset automatically if the tag is propagated to the next redrive logic.

Byte 1, Bit 6 - Secondary Cycle Steal Grant: this bit is set when the secondary cycle steal grant signal is sent to the adapter controlled by the redrive logic, and is reset automatically by:

- The next cycle steal operation if the redrive logic propagates the cycle steal grant signal to the next redrive address.
- The next PIO operation on the bus.

Byte 1, Bit 7 - Command Reject: this bit, if on, indicates that an invalid command was sent to the the addressed redrive logic.

Command Output X'0' or X'8' (Write Error Register)

This command is used to set the redrive logic error register for diagnostic purposes. The high order bit of the command may be off, indicating a broadcasted command, or on if the command is addressed to a specified redrive logic. The error register(s) of the redrive logic(s) addressed by the instruction is loaded from the specified register. Byte 1 is not used. If the command is broadcast, all redrive logic error registers are set. The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	In Bus Parity Error
	1	Out Bus Parity Error
	2	Out Bus Tag Check
	3	In Bus Tag Check
	4	Halt Remember
	5	Secondary Select Out
	6	Secondary Cycle Steal Grant
	7	Command Reject
1	0/7	(not used)

Redrive Error Register: Byte 0 is collectively called the Redrive Error Register. For a detailed description of its bits, see under the 'Po command above.

Command Input X'1' or X'9' (Read Error Register)

The high order bit of the command may be off, indicating a broadcast command, or on if the command is addressed to a specified redrive logic. The register addressed by the instruction is loaded from the redrive logic and contains the redrive address and the event that set the redrive level 1 interrupt. If the command is broadcast, and more than one redrive has a level 1 interrupt pending, the redrive logically closer to the CCU on the internal bus presents its error register data first. Additional Input X'1' or X'9' instructions must be executed if more than one redrive has a level 1 interrupt pending. The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	1
	1	Enable/disable (enable = 0, disable = 1)
	2	PRA Bit 4
	3	PRA Bit 2
	4	PRA Bit 1
	5	SRA Bit 4
	6	SRA Bit 2
	7	SRA Bit 1
1	0	In Bus Parity Error
	1	Out Bus Parity Error
	2	Out Bus Tag Check
	3	In Bus Tag Check
	4	Halt Remember
	5	Secondary Select Out
	6	Secondary Cycle Steal Grant
	7	Command Reject

Redrive Error Register: For a detailed description of the bits of this command, see under 'Command Input X'0' (Poll) on page E-6.

Command Output X'1' or X'9' (Disable Drivers)

This command is used to inhibit the **inputs** of the dependant adapter(s) and dependant redrive logics (if any) from the bus. It is, however, still possible to **send** information to the dependant logics, and to **send and receive** information from the redrive logic itself. The high order bit of the command may be off, indicating a broadcast command, or on if the command is addressed to a specified redrive logic. As this is a command that performs a function, the contents of the register addressed by the command are ignored.

Command Output X'2' or X'A' (Enable Drivers)

This command is used to enable the addressed redrive logic and the adapter(s) and redrive logics dependant on it. The high order bit of the command may be off, indicating a broadcast command, or on if the command is addressed to a specified redrive logic. As this is a command that performs a function, the contents of the register addressed by the command are ignored.

Command Output X'5' or X'C' (Reset)

This command is used to reset all latches of the addressed redrive logic(s), except the enable/disable latch. The high order bit of the command may be off, indicating a broadcast command, or on if the command is addressed to a specified redrive logic. As this is a command that performs a function, the contents of the register addressed by the command are ignored.

APPENDIX F. INITIAL PROGRAM LOAD (IPL)

The initial program load (IPL) mechanism controls the loading of the control program into the controller. Loading takes place via one of the ports defined in the IPL port table, using either a channel adapter (channel-attached controller) or a transmission line and one of the communication scanners (link-attached controller).

Initial program load is performed under the following conditions:

- When the system is first powered up.
- When power is lost and an auto-start occurs.
- When processing cannot continue because of a controller error condition.
- When the channel adapter decodes a 'Write IPL' command from the host.

SEQUENCE OF IPL

IPL takes place in several phases:

- Phase 0: load the MOSS.
- Phase 1: initialize and test the CCU.
- Phase 2: load the Controller Loader Dump Program (CLDP).
- Phase 3: load the communication scanners.
- Phase 4: load and/or initialize the control program.

Note: For a channel-attached controller, phases 3 and 4 take place simaltaneously; for a link-attached controller, phase 4 cannot take place until phase 3 (load the communication scanners) has been completed.

Phase 0: Load the MOSS

This phase loads the MOSS from the MOSS diskette, executes a series of internal tests, and initializes the MOSS. The hexadecimal display indicates 3 digits from X'F00' through X'FE0 while the MOSS is being loaded and initialized; at the end of this phase, the indicators show X'FEF'.

Apart from the changing digits on the hexadecimal display, this phase is invisible to the user, except in the case of a power on sequence. In this case, the MOSS General Menu is displayed and the MOSS status indicates 'MOSS ALONE'.

Note: Phase 0 only takes place after a 'cold start', that is, under the following conditions:

- At power on time.
- After and auto restart.
- If the START switch is pressed at the control panel.

Phase 1: Initialize and Test the CCU

During this phase, the MOSS initializes the CCU, as follows:

- The CCU latches are initialized.
- The local storage registers are set to all zeros.
- The main storage is set to all zeros with good parity (power on IPL only).
- The storage protect/address exception mechanism is disabled (power on IPL only).
- The channel adapter registers are initialized with good parity (power on IPL only).
- The CCU and internal bus are tested.
- The communication scanners are tested.

The hexadecimal display indicates the 3 digits X'FF1' during phase 1; at the end of this phase, the indicators show X'FF2'.

Apart from the changing digits on the hexadecimal display, and the changing CCU status on the MOSS Machine Status Area, this phase is also invisible to the user.

Phase 2: Load the Controller Loader Dump Program (CLDP).

During this phase, the following events take place:

- The CLDP loader program and the IPL Ports table are loaded into the CCU from the MOSS diskette.
- Control is passed to the CLDP.

While the MOSS is entering IPL Phase 2, the CLDP does the following:

- Enables the channel adapters (channel attached controllers only).

  Note: Until the channel adapters are enabled, they do not reply to initial selection, but simply propagate Select Out to the next device.

- Signals to the hosts that IPL is required:
  - Channel-attached controller:
    an asynchronous Device End/Unit Check (DE/UC) status is sent to all hosts to signal that control program loading may begin.
  - Link-attached controller: the CLDP program must wait for the communication scanners to be loaded and initialized before it can communicate with the host(s).
  - Monitors the IPL ports.

At the end of this phase, the indicators show X'FF3', and the message

ENABLED PORTS CA xxxxxx L NNNNNNNN

is displayed on the operator console.

where:

xxxxxx is a pattern of mixed 'Y' and 'N' characters to indicate the enabled (Y) or disabled (N) state of channel adapters 1 through 6.

Note: The 'L NNNNNNNN' pattern indicates that all the Link IPL ports are disabled at this point in time.

Phase 3: Load the Communication Scanners.

During this phase, the following events take place:

- All scanners are loaded and initialized in parallel.
- The MOSS monitors the scanner loading process, and sends to the CLDP a list of the scanners that have been successfully initialized.

  Note: Control program loading starts here for link-attached controllers.

At the end of this phase, the indicators show X'FF4', and the message

ENABLED PORTS CA xxxxxx L yyyyyyyy

is displayed on the operator console.

where:

xxxxxx is a pattern of mixed 'Y' and 'N' characters to indicate the enabled (Y) or disabled (N) state of channel adapters 1 through 6.

yyyyyyyy is a pattern of mixed 'Y' and 'N' characters to indicate the enabled (Y) or disabled (N) state of Link IPL ports 1 through 8.

Phase 4: Load and/or Initialize the Control Program.

For a channel-attached controller, control program loading may take place simultaneously with phase 3; in this case, phase 4 is limited to the initialization of the control program already loaded. For a link-attached controller, control program loading cannot take place until phase 3 (load the communication scanners) has been completed.

Channel-Attached Controller Loading

During this step, the following events take place:

1. The host sends a Write IPL (X'05') command to the channel adapter to inform the CCU that the host is ready to send the control program modules. No data is actually sent via this command. The channel adapter replies with an initial status of Channel End alone; this allows the channel to disconnect immediately. This is necessary, because the controller may not be ready at this time (for example, after a restart, when it must be reinitialized).
2. At the same time that the channel adapter sends channel end to the host in reply to Write IPL, it raises a level 3 interrupt to the CCU, and informs the MOSS by hardware. If necessary, the MOSS initializes the CCU, and loads it with the CLDP.

3. When the CLDP is ready, it sends a Device End status to the host (this Device End may be immediate in the case of normal power on, or delayed in the case of a restart, for example). At this point, the hexadecimal indicators display X'FF5', and the message:

   CA IPL DETECTED ON CA x

   is displayed on the operator console, followed a little later by a second message:

   LOAD IN PROGRESS ON CA x

   where:

   x is the number of the channel adapter 1 through 6.

   Note: Channel End and Device End never occur in the same status in reply to a Write IPL command.

4. At this point, the host transfers the control program load module from the host to the CCU, using the normal Write (X'01') command for each block of text. The CLDP reads each block of text, and answers each with a Channel End/Device End status.

5. On the last block of text, the host sends a Write Break (X'09') command to the CLDP, followed by a final Write (X'01') command containing the control program entry point; this causes the CLDP to signal to the MOSS that the control program is loaded, and to transfer control to the control program. At this point, the hexadecimal indicators display X'FF7', and the message

   CONTROL PROGRAM LOADED

   is displayed on the operator console.

6. Under the control of the control program, CCU software initialization takes place. In particular, the CCU receives the CDF parameters from the MOSS, and the MOSS receives the Control Program Initialization Table (CPIT) from the CCU.

7. When CCU initialization is complete, the hexadecimal indicators display X'000' and the message

   IPL COMPLETE

   is displayed on the operator console; the controller is ready.

   Note: If an error is detected during the IPL, the hexadecimal indicators display X'FFE', and the message

   IPL COMPLETE + ERRORS

   is displayed on the operator console.

Link-Attached Controller Loading

At the end of phase 3, the scanners are ready to transmit and receive data. The FF4 indication appears on the hexadecimal display, and the message:

ENABLED PORTS CA xxxxxx L yyyyyyyy

is displayed, where yyyyyyyy indicates those lines that are designated as IPL ports.

The transfer of the control program now takes place via one of the designated IPL ports. A series of indications on the hexadecimal display and on the operator console allows the operator to follow the operation.

1. When one of the hosts is ready to send the load modules to the controller, the hexadecimal indicators display X'FF6', and the message:

   LINK IPL DETECTED ON L xxx

   is displayed on the operator console, followed a little later by a second message:

   LOAD IN PROGRESS ON L xxx

   where:

   xxx is the address of the link from 0 through 255.

2. The host now transfers the control program load module from the host to the controller via a channel-attached 3725, 37XX, or 3705 using the SDLC protocol under the control of the CLDP.

3. When the last block of data has been loaded into the controller, the hexadecimal indicators display X'FF7', and the message

   CONTROL PROGRAM LOADED

   is displayed on the operator console.

4. Under the control of the control program, CCU software initialization takes place. In particular, the CCU receives the CDF parameters from the MOSS, and the MOSS receives the Control Program Initialization Table (CPIT) from the CCU.

5. When CCU initialization is complete, the hexadecimal indicators display X'000' and the message

   IPL COMPLETE

   is displayed on the operator console; the controller is ready.

   Note: If an error is detected during the IPL, the hexadecimal indicators top display X'FFE', and the message

   IPL COMPLETE + ERRORS

   is displayed on the operator console.

APPENDIX G. BRANCH TRACE

Branch Trace Introduction

Branch trace is intended as a general debugging tool for the control program. It records, in the **branch trace table**, the addresses where branches are taken during CCU instruction execution. Interrupts and returns from interrupts via the EXIT instruction are considered as branches for branch trace operations.

Notes:

1. The range of addresses to be traced may be selected by the user via the MOSS.

2. The interrupt levels to be traced may also be selected by the user.

Branch Trace Table

The branch trace table consists of a series of eight-byte entries, one for each branch. The following information is recorded in the table:

- The 'come from' program level.
- The 'come from' instruction address.
- The 'go to' program level.
- The 'go to' instruction address.

The information actually recorded varies slightly depending on the type of branch. The following table should make this clear:

	Trace table entry byte							
	0	1	2	3	4	5	6	7
Type of branch	Come from level	Come from address			Go to level	Go to address		
True branch	CPL	Instruction address			CPL	Branch address		
IAR modification	CPL	Instruction address			CPL	New IAR (Reg 0)		
Program interrupt	Old level	Address of last inst executed in old lev.			New level	Address of 1st inst. executed in new lev.		
EXIT instruction	Level EXIT'd	Address of EXIT instruction			New level	Address of 1st inst. executed in new lev.		

Where:

CPL = current program level

The 'come from' and 'go to' levels are encoded as follows:

Level	Hex value
1	X'01'
2	X'02'
3	X'03'
4	X'04'
5	X'05'

Setting up the Branch Trace

The branch trace is set up from the operator console via the MOSS. The user must pass to the MOSS the address of a suitable buffer area, and an initial buffer count (in multiples of eight bytes), for the branch trace table (alternatively, these two parameters may be set by the control program, if one is resident). At this time, other parameters may also be specified, such as:

- Range of addresses to be traced.
- Interrupt levels to be traced.
- Whether or not branch trace wrapping is required. If wrapping is allowed, when the branch trace table is full, the following entries overwrite the earlier entries in the buffer, which are therefore lost.
- Whether or not stop on address is simultaneously required.

Note: The combination of branch trace wrap and stop on address may be used to record the last 'n' branch traces before the stop on address occurred, where n is equal to or less than the number of entries reserved for the branch trace table.

After this initial setting up, the operation is totally transparent to the user.

Programming Notes:

1. When branch tracing is in operation, some degradation of instruction execution time occurs.
2. The address of the branch trace table, as received from the MOSS is available to the control program via the Input X'7B' instruction.
3. The branch trace buffer count, as received from the MOSS is available to the control program via the Input X'7C' instruction.
4. To avoid filling the branch trace table with unwanted timer interrupt traces, the code traced should not include any level 3 code associated with the servicing of timer interrupts.
5. Local storage register X'18' (the IAR of level 3) should be set to a storage address outside of the range of the storage block being traced. Register X'18' must only be set to this value while the CCU is in the 'Wait' state.
6. If no control program is resident in CCU storage, the address and the maximum number of entries must be fixed by the user when he calls the function from the MOSS. If a control program is resident, the address and the count may be set up by the program, and passed to the MOSS. However, the user may still modify these two parameters via the MOSS.

INDEX

C

D

E

F

G

L

M

N

O

P

R

S

T

U

W

X

Z

Numerics

3720/3725
Communication Controllers
Principles of Operation
Order No. GA33-0013-6

READER'S COMMENT FORM

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

Your comments will be sent to the author's department for whatever review and action, if any, are deemed appropriate.

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

Possible topics for comments are:

Clarity Accuracy Completeness Organization Coding Retrieval Legibility

If you would like a reply, please give details overleaf:

Number of latest Newsletter associated with this publication: . . . . . .

Thank you for your cooperation. No postage stamp necessary if mailed in the USA. (Elsewhere, an IBM office or representative will be happy to forward your comments or you may mail directly to the address in the Edition Notice on the back of the title page.)

Note: *Staples can cause problems with automated mail sorting equipment. Please use pressure sensitive or other gummed tape to seal this form.*

Reader's Comment Form

Fold and tape | **Please Do Not Staple** | Fold and tape

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 40 ARMONK. N.Y.

POSTAGE WILL BE PAID BY ADDRESSEE:

International Business Machines Corporation
Department 6R1LG
180 Kost Road
Mechanicsburg
PA 17055

Fold and tape | **Please Do Not Staple** | Fold and tape

If you would like a reply, please print:

Your Name _____

Company Name _____ *Department* _____

Street Address _____

City _____

State _____ *Zip Code* _____

IBM Branch Office serving you _____

IBM

Publication Number
GA33-0013-6

File Number
S370/30xx/4300-09

Printed in
USA

IBM®

GA33-0013-06