HONEYWELL EDP

HARDWARE BULLETIN

SERIES 200

TYPE 286-4 AND TYPE 286-5 MESSAGE-MODE, MULTI-CHANNEL COMMUNICATION CONTROLS

SUBJECT: Equipment specifications for the Type 286-4 and Type 286-5 Message-Mode, Multi-Channel Communication Controls.

SPECIAL INSTRUCTIONS: References used in this publication include the Honeywell Series 200 Programmers' Reference Manual (Models 200/1200/2200), Order No. 139, and the ASA document Control Procedure for Data Communications, X3.3.4/60, dated March, 1965.

DATE: June 30, 1966

FILE NO.: 112.0005.1124.0-270*

8982
5666
Printed in U.S.A.

*When ordering this publication please specify Title and Underscored portion of File Number.

FOREWORD

The Type 286-4 or Type 286-5 Message-Mode, Multi-Channel Communication Control provides the interconnection of a Series 200 central processor with a maximum of 63 half-duplex communication lines via Type 285 Communication Adapters.

The communication control is compatible with all standard Series 200 central processors except the Models 201 and 201-1. The Model 121 central processor must be equipped with optional Feature 1015 or 1016 — see the Honeywell Series 200 Programmers' Reference Manual (Model 120), Order Number 141.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION I
INTRODUCTION

GENERAL DESCRIPTION

The Type 286-4 or 286-5 is organized as two processors, one servicing the bit terminus (interface with the communication adapter) and one servicing the character terminus (interface with the central processor). A one-microsecond — 2048 words by nine-bit — memory is time-multiplexed between each processor. Multiplexing is accomplished on an alternate cycle basis, providing an effective memory-cycle time of two microseconds for each processor. The data section of communication control memory has three memory areas for each of the 63 lines: buffer storage, adapter unit interface register, and block parity (BP) register. The control section of communication control memory is also subdivided into three parts: scan table, queues, and line control (see "Control Memory Organization," Section III). Interconnection of the communication control with the central processor and communication lines via Type 285 Communication Adapters is illustrated in Figure 1-1.

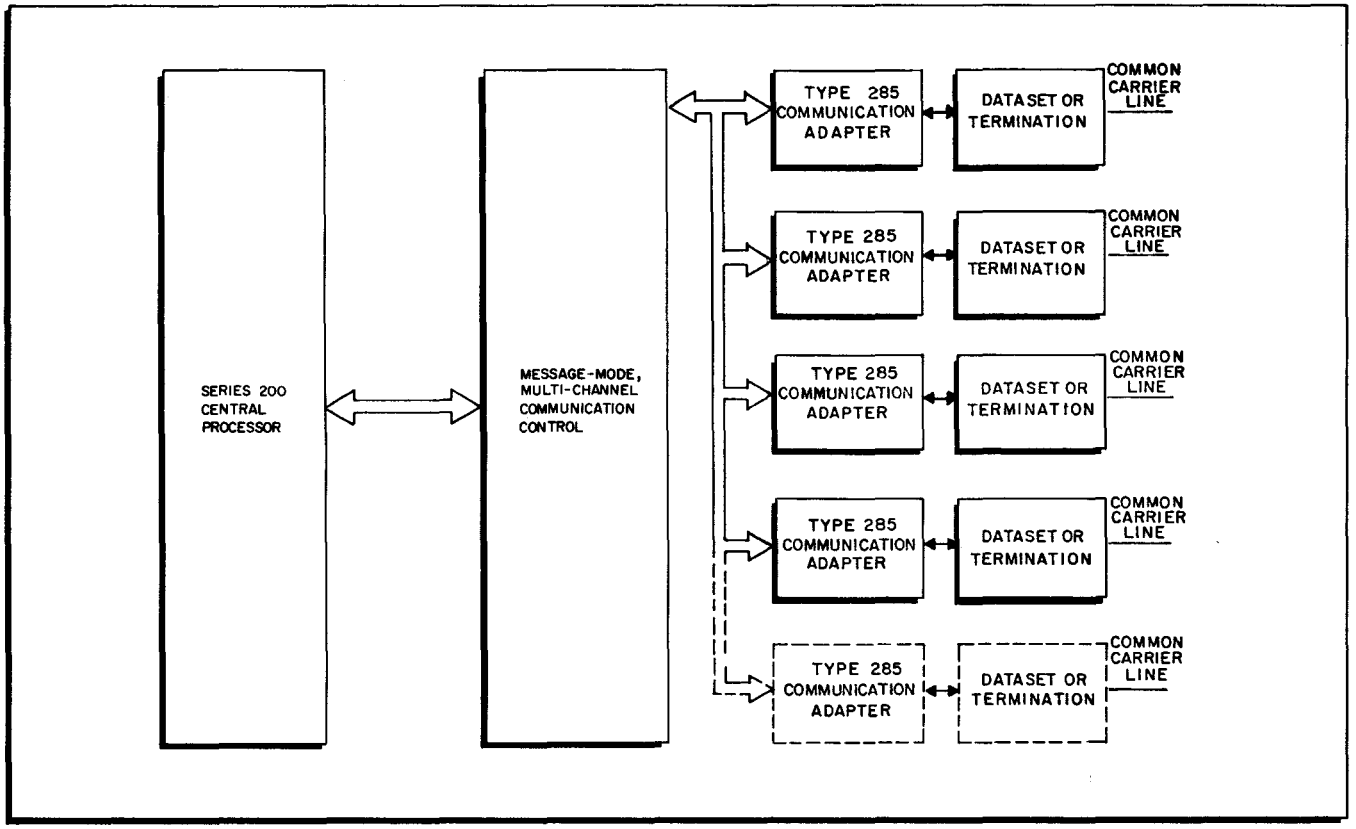


Figure 1-1. Types 286-4 and 286-5 Applications

INTERFACE WITH SERIES 200 CENTRAL PROCESSORS

Standard peripheral interface logic connects the communication control to the standard Series 200 peripheral bus.

INTERFACE WITH TYPE 285 COMMUNICATION ADAPTERS

The Type 286-4 and Type 286-5, equipped with a group of Series 200 interface packages, connects to a Type 285 Communication Adapter by means of miniature coaxial cables.

SECONDARY BUS

The standard secondary bus is required to interconnect the Type 286-4 and -5 with any Type 285 Adapter. Table 1-1 lists the interconnecting lines. The secondary bus is connected in series between the logic drawers in which it is utilized. The aggregate maximum cable length is 50 feet. Circuit-driving ability restricts the number of attached logic drawers to eight.

Table 1-1. Standard Secondary Bus Interconnections

LINES	DESCRIPTION
DATA	
1 — Type 286 to Type 285	Information out (serial by bit)
1 — Type 286 from Type 285	Information in (serial by bit)
ADDRESS	
6 — Type 286 to Type 285	Select a particular Type 285
REQUEST	
1 — Type 286 from Type 285	Type 285 requests a bit from Type 286
1 — Type 286 from Type 285	Type 285 requests Type 286 take a bit
TIMING	
1 — Type 286 to Type 285	Bit ready on data-out line
1 — Type 286 to Type 285	Bit received on data-in line
RESET	
1 — Type 286 to Type 285	Reset by INITIALIZE pushbutton on central processor control panel
SPECIAL STROBE	
1 — Type 286 to Type 285	Control information to the Type 285

SECTION II

TYPE 286-4 AND TYPE 286-5 SYSTEM CONSIDERATIONS

AMERICAN STANDARD ASSOCIATION (ASA) COMPATIBILITY

The message-mode, multi-channel communication control is capable of operation with the ASA-defined "basic mode" of control procedure (see Control Procedures for Data Communications) using the standard character set of the ASCII code.

READ/WRITE CHANNEL ASSIGNMENTS

The message-mode, multi-channel controls will not operate with the high-speed transfer feature. The maximum transfer rate of a read/write channel assigned to the communication control is one access every six microseconds. If a read/write channel access occurs less frequently, the throughput (i.e., the over-all productivity of the control) may decrease.

STORED ADDRESS CAPABILITIES

In data transfer to main memory, the multiplexer feature of the peripheral bus enables the control to enter an 18-bit address in the assigned read/write channel. This addressing ability allows the control to place a data character anywhere within a 262K memory block. Any additional address bits — if required — are entered into the read/write channel by the program when the PDT instruction assigning the read/write channel is executed. This means that the main memory buffer area must be located in the same 262K block as the status field. The status field is specified by the address of the Peripheral Data Transfer (PDT) Instruction (see Section IV).

DATA

Throughput

The maximum character throughput of the control, independent of code level, is 7000 characters per second. System throughput is determined by message size and programming.

Rate of Data Transfer

The message-mode, multi-channel communication control handles any rate of data transfer provided its maximum character throughput is not exceeded. The data rate of a line is determined by the terminating line adapter.

Message Size

Characters must be handled in blocks if a high throughput is to be obtained. The number of characters required per block, or message, is a function of the number of active lines, character time, and program execution time. In order to utilize most effectively the capabilities of the communication control, the block size used must become progressively larger as the time between received or transmitted characters decreases. The formula below can be used to de-

termine the minimum block size for most efficient use of the Type 286-4 and Type 286-5. Block sizes for most efficient line usage will, in general, be far larger than the minimum figure derived from the formula.

$$\text{Number of characters/block/line} > \frac{\text{number of active lines x program time/line}}{\text{character time}}$$

SYNCHRONIZATION FOR SYNCHRONOUS LINE

The communication control compares the stored synchronization (SYNC) character with the incoming bit stream from a synchronous adapter. Upon recognition of two consecutive synchronization characters received over a line, the synchronization condition for that line is set. Synchronization is maintained from the time it is established until receipt of an EOT character. The only exception here occurs when the IBM STR unit is utilized (refer to the Honeywell Hardware Bulletins Type 281-2A Communication Control and Type 285-2A Communication Adapter, Order Number 416, and Type 281-2D Communication Control and Type 285-2D Communication Adapter, Order Number 079. In this case, the adapter recognizes the synchronization characters and it maintains synchronization throughout the operation.

The Type 286-4 or Type 286-5, having synchronized normally upon recognition of two consecutive synchronization characters, transfers all non-synchronization characters to the central processor. In addition to the synchronization characters, control and transmit leader characters — upon recognition — cause the communication control to synchronize. However, unlike the synchronization characters, the control and transmit leader characters are transferred to the central processor. Then, all non-synchronization characters are transferred to main memory under control of the present command.

The communication control remains in synchronization with the incoming bit stream until the control quits the receive mode. The "Resync" PCB instruction resets the synchronization condition for a specific line in the message-mode, multi-channel communication control.

START OF INPUT TRANSFER FOR ASYNCHRONOUS LINES

The Type 286-4 or Type 286-5 uses the stored synchronization character as a start-of-input-transfer (SIT) character. When the SIT character is other than zero, the control will not transfer any characters received from an asynchronous line to the central processor until it receives a line character equal to the stored SIT character. The SIT character and all characters subsequently received are transferred to the central processor. This process continues until the control leaves the receive mode.

The "Resync" PCB instruction, addressed to any asynchronous line in the receive mode with a stored SIT character, halts character transfer to the central processor until a character equal to the SIT character is received. The stored SIT character is not interpreted when the line is in the transmit mode.

SECTION III
FUNCTIONAL DESCRIPTION

TYPE 285 COMMUNICATION ADAPTER

A Type 285 Communication Adapter provides the buffering, timing, control and interface required by a particular communication line/remote terminal configuration. At least one Type 285 Communication Adapter is required for each communication line serviced by the Type 286-4 or Type 286-5 Communication Control.

TYPE 286-4, -5 MESSAGE-MODE, MULTI-CHANNEL COMMUNICATION CONTROL

The communication control regulates the data flow between the central processor and communication lines by means of commands stored in its memory. In transmission, the control of data flow consists of extracting a line character from main memory, transferring that character to buffer storage, and serializing it for transmission by a communication adapter. In reception, the communication control receives data from a Type 285 and assembles bits into characters then accesses main memory and transfers the characters from buffer storage to the central processor. The control, by means of the multiplexing peripheral bus, utilizes addresses stored in its own memory to access main memory.

The control is able to recognize characters for control purposes; such recognition may terminate the present command instruction and cause a stored second command to become the new present command instruction. The present command is also terminated when a record mark is sensed in main memory. An interrupt alerts the program to the change of commands. If the response bit is not set in the present command, the interrupt is prevented.

DATA TRANSFER

Data transfer occurs between the central processor and the communication control, also between the control and the adapters. Data transfer between the central processor and the communication control is accomplished in two phases: (1) the transfer of the main memory address, i.e., the location into which data is to be placed or from which it is to be removed; and (2) the transfer of the actual data. Data transfer between the control and the adapter is accomplished on a demand-per-bit basis to the adapter or from it, while the adapter is interrogated during scanning.

When the communication control senses a call for data transfer, the stored main memory address is transferred, in three buffer cycles, to the assigned read/write channel. The contents of the control buffer register is then transferred to main memory as one or two six-bit characters,

depending on the state of the character transfer bit. Thus, single-character transfers are located in consecutive main memory locations starting with location A, to location A+(n-1), where A is the initial address stored in the communication control by a "Control" PCB instruction and n is the number of characters transferred. Two character transfers are stored as consecutive pairs in main memory locations starting with location A, to location A+(2n-1), where A and n are as defined above. The main memory address to be stored in the control is incremented by one or two, depending on the state of the character transfer bit, before it is again stored in control memory.

MEMORY TEST FEATURE

The communication control is designed to allow the execution of a memory diagnostic program. The program is capable of writing into every memory location and interrogating it. An "Output" PDT instruction is required to enter the information into control memory and a subsequent "Input" PDT instruction transfers the contents of each word to main memory. The "Input" PDT instruction transfers the communication control parity bit together with the eight data bits. This enables a diagnostic program to have access to all bits in memory. No hardware parity check is associated with the input transfer. A record mark is required to terminate data transfer in either direction. Two central processor characters are required for each communication control word.

MEMORY ORGANIZATION

The control unit memory consists of two main sections, one for data and one for control. The data section of the control unit memory has three memory words for each of the possible 63 lines: buffer storage, adapter interface register, and block parity (BP) register. The control section of the control unit memory is subdivided into three parts: scan table, queues, and line control.

Scan Table

The scan table controls the servicing sequence of the Type 285 Adapters; it is "program-loaded" prior to on-line operation. To guarantee no loss of data, the table should be weighted according to line speed. No priority is associated with the servicing of the adapters. Two hundred and fifty-six words are allocated to this table as described in Section IV (see "Scan Table").

Queues

Four queue tables are required for operation of the communication control. During real-time operation, the program does not have access to the queue tables. Each queue can store 63 service requests. There is no protection against queue overflow, nor is there any indication of its occurrence. Three queues are required to service the interface with the central processor.

Interrupt calls — handled on a "first-come, first-served" basis — are stored in one queue table. Two queues are required to stored requests for character processing. One of these is used for high-speed lines; it has priority over the other queue, used for lower-speed lines. The program, during initial loading, selects which queue a line is to utilize. Requests are handled on a "first-come, first-served" basis. The fourth queue is used for replies sent to the bit processor from the character processor. Again, requests are handled on a "first-come, first-served" basis.

Line Control

Line operation is controlled by information stored in control words in the control unit memory. The control words consist of program/variable and fixed characteristics. The program/variable characteristics are: present command, present address, and present recognition characters; these characteristics are defined in the following paragraphs. The fixed characteristics are defined under "Memory Loading" in Section V of this publication.

PRESENT COMMAND

The present command specifies the mode and operating characteristics of the associated line. There are four possible modes:

1. Transmit — This mode transfers characters from the designated area in main memory and transfers one bit at a time to the adapter.
2. Transmit-Repeat — During operation in the transmit-repeat mode, the first data character specified by the stored address is sent continually until transmission is halted by another command. No normal punctuation is required in main memory.
3. Receive — In the present command receive mode of operation, the control receives bits from the line by means of the adapter. The control assembles these bits into characters and transfers them to the designated area in main memory.
4. Inhibit Line Operation — In this mode, line operation is inhibited. After line operation is inhibited, no data transfers take place, no internal interrupt calls are generated and no bit activity from the adapter is serviced. The timer is independent of this command and must be disallowed to prevent timeout. Any interrupt calls stored in the interrupt queue are not inhibited, thus it is possible for an interruption to occur from a line which has been inhibited by interrupt calls previously stored in the queue.

The present command can specify seven operating characteristics of the associated line in addition to the four possible modes. These operating characteristics are:

1. Character Parity — The generation and checking of character parity can be specified on a command basis. Parity is specified as either odd or even. A line may also be specified to operate without parity being generated or checked.

2. Block Parity — The generation and checking of block parity can also be specified. This modulo two sum (even parity) is calculated for all the characters in the message block (except synchronization characters) transferred to or from the central processor (see also "Block Parity," Section IV).

3. Parity Bit of the Block Parity Character — This operating characteristic allows the parity bit to be handled either the same as the data character parity bit or as the modulo two sum of the parity bits.

4. Response — This characteristic allows an interrupt to be initiated at the end of a termination sequence.

5. Allow Timer — The allow timer characteristic permits the action of the timer on a designated line.

6. Command Termination — This characteristic determines the number of characters transferred to/from main memory on account of termination caused by character recognition. The block parity and command termination characteristics are related (see "Block Parity," Section IV).

7. Character Transfer — The character transfer bit determines whether each line character is to be transferred as one central processor character or as two. This characteristic can be specified independent of line code.

PRESENT ADDRESS

The present address is stored as three six-bit characters in the communication control memory. When data is to be transferred to the central processor or from it, these characters are entered into the assigned read/write channel. The data itself is then transferred. The present address is incremented by one or two, depending upon the state of the character transfer bit.

PRESENT RECOGNITION CHARACTERS

The communication control checks all characters received and all characters to be transmitted. If they compare with either of the two stored recognition characters, except zero, the present command is terminated according to the state of the command termination bit. A zero character indicates no recognition is to take place for that recognition character.

SYNCHRONIZATION CHARACTER

The stored synchronization character is interpreted according to the type of adapter associated with the line. When the synchronization characters are entered into the control unit memory, the complete line character, including a parity bit, is entered; any unused bits must be zero.

Synchronous Lines

The synchronization character should be entered (on a per-line basis) during initialization of the system; it should not be changed during real-time operation of the system. In reception,

the synchronization character is used to obtain synchronization with the incoming bit stream and is deleted from the received data. In transmission, the synchronization character is transmitted if no new data characters are available. When the block parity feature is used, the synchronization character is not included in block parity computation in either the receive or transmit modes.

Asynchronous Lines

The start-of-input-transfer (SIT) synchronization character can be entered during initialization of the system or by a "Line Control Information" PCB instruction. The latter instruction is described in "Peripheral Control and Branch (PCB) Instruction," Section IV. The SIT character is used to enable data transfers when the stored SIT character is other than zero. The SIT character is not deleted from the received data; it is included in block parity computation when the block parity feature is used. Characters which precede the SIT character and are not transferred to main memory are not included in block parity computation.

NEXT COMMAND

The next command, address, and recognition characters are stored in the Type 286-4 or Type 286-5 and become the present command instruction upon termination of the current present command. If no next command is stored, the line goes into an intermediate mode of operation (see "Intermediate Mode," Section V).

LINE STATUS

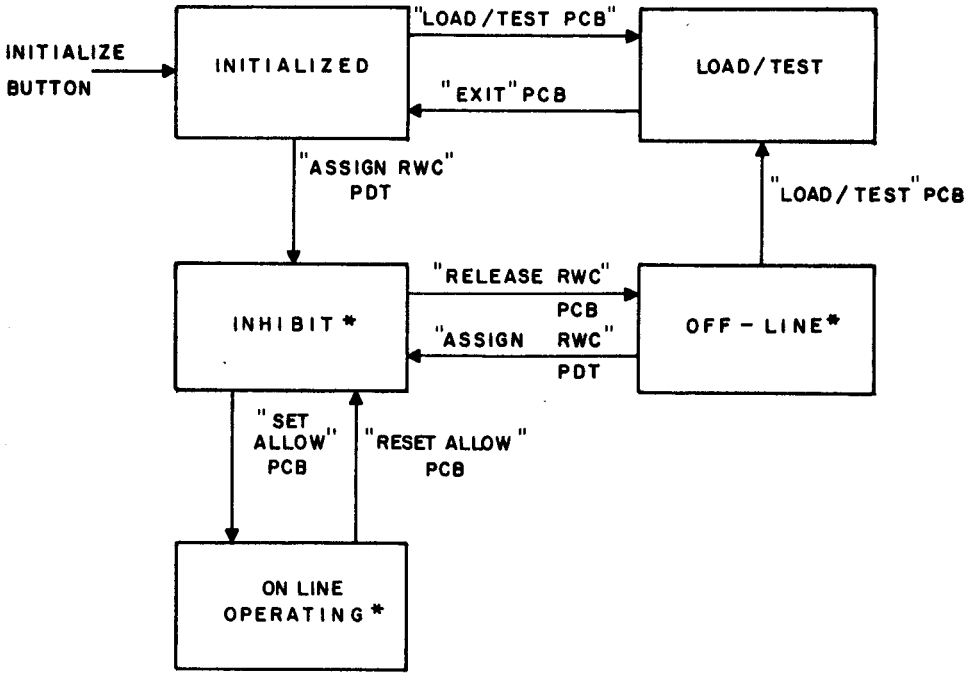
Prior to the setting of the interrupt condition, line status information is transferred to the field specified by the A address of the "Read/Write Channel Assignment" PDT instruction, together with the line number. This information tells the program the cause of the interrupt. The character location feature of the control unit causes the address to be transferred. Punctuation is not required in the status field (see "Status Field," Section IV).

SECTION IV

PROGRAMMING REFERENCE DATA

OPERATIONAL STATES OF THE TYPE 286-4, -5

There are five operational states of the communication control. Figure 4-1 illustrates these five states together with the instructions required for entering and leaving each. The action and state of the control is unspecified if an instruction is executed in any state other than that shown in the figure.



*A hardware parity error causes the control to enter the initialized state if the Halt/Continue function is in Halt.

Figure 4-1. State Chart

Initialized State

In its initialized state the control is completely reset, i.e., its lines are not serviced and data transfers and interrupts cannot occur. This state is entered by depressing the INITIALIZE button on the Series 200 control panel, or it may be entered from the Load/Test state of operation.

Load/Test State

The load/test state is entered by the control when it is in the initialize, or off-line, state and a "Load/Test" PCB instruction is executed. The load/test state enables the communication

control to accept "Data Transfer" PDT instructions. Information is entered into the control unit memory by "Output" PDT instructions. Odd parity generation is performed on each word and the result is stored together with the data as the ninth bit. "Receive" PDT instructions transfer all nine bits to the central processor, but no parity checking occurs.

The load/test state is reset by the execution of an "Exit" PCB instruction or by depressing the INITIALIZE button the the Series 200 control panel. Interrupts cannot be generated in the communication control while it is in the load/test state.

On-Line Operation

In this operational state the control performs the functions stored in the present command registers of each line in the scan table. This state is entered by the execution of the "Set Allow" PCB instruction permitting interrupts.

Inhibit Interrupt

During operation in the inhibit interrupt state, the communication control can not interrupt the program, but it is continuing to scan the communication lines and to receive and transmit bits. Lines continue to operate until their present command and next command activities are completed.

Operation in this state may cause the interrupt queue to overflow. If this occurs only the last 63 interrupt calls are remembered; all earlier interrupt calls are lost. If the control is to enter the inhibit interrupt state from the on-line operational state, the allow function must be disabled. The Allow function is reset by executing a "Reset Allow" PCB instruction. Interrupts are inhibited but data transfers can continue to occur. The control unit leaves this operational state when the Allow function is again set, thus returning it to the on-line state or when the read/write channel is released, thus entering it into the off-line state.

Off-Line State

Here the central processor communication lines, interrupt function and read/write channels are not available to the communication control. The Type 285 Communication Adapters continue to be serviced according to the stored characteristics. Line operation for transmit is as though no next command were available. In the receive mode, overwrite errors may occur depending on how long the control remains in the off-line state of operation. The interrupt queue may possibly overflow as described in the inhibit state.

The control enters the off-line state from the inhibit state when a PCB instruction releases the assigned read/write channel. A new read/write channel assignment can again place the communication control in the inhibit state, or a PCB instruction can enter it into the load/test

state. Bit activity is stopped and the communication adapters are no longer serviced when the communication control enters the load/test state.

INTERRUPT CAPABILITY

A condition requesting program attention in the Type 286-4 or Type 286-5 sets the interrupt to the central processor. Prior to the actual interrupt — except in the case of a communication control parity error — the line status and the number of the line causing the interrupt are transferred to main memory. Once the program has utilized or moved the status information, a "Status Taken" PCB instruction must be executed to tell the control that another status transfer and subsequent interrupt can be sent to the central processor.

The causes for interrupts from the communication control are: a hardware-detected parity error occurring in the control unit memory; internally, a record mark sensed in memory, a recognition of characters entered into memory by a PCB instruction, or a forced sequence of commands; externally, a line timeout; or an address request.

Timing of Transmit Interrupts

An interrupt request is stored in the interrupt queue after a record mark is sensed in main memory or a character is recognized. When a record mark is sensed, the present command is terminated and the next command — stored in the control — becomes the present command before the interrupt call is stored. When a character is recognized, the termination procedure depends on the state of the command termination and block parity bits (see "Block Parity," later in this section).

Timing of Receive Interrupts

An interrupt request is stored in the interrupt queue after a record mark is sensed in main memory or a character is recognized. If a record mark is sensed, the present command is terminated and the next command becomes the present command. When a character is recognized, the termination procedure again depends upon the state of the command termination and block parity bits.

Interrupt Caused by Communication Control Parity Error

If an interrupt is caused by a hardware parity error, a PCB instruction tests for that error. A second PCB instruction is executed to reset the error function.

Present Command Termination

The following events cause the present command to be terminated and the next command — if one is stored — to become the new present command; a record mark is sensed in main memory;

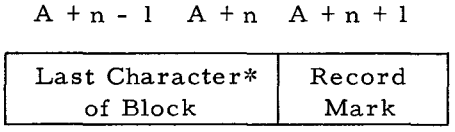
a character is recognized and termination is completed; or a PCB instruction causes a forced sequence of commands.

RECORD MARKS

Location of Record Marks

In the transmit mode the record mark is located in the next memory location following the last character of the block, as shown below.

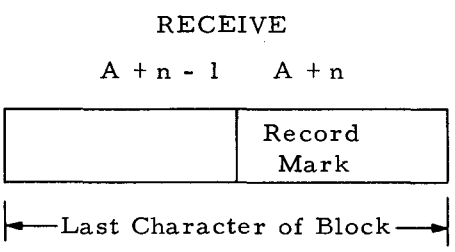
TRANSMIT



*an 8-bit character is assumed.

In the receive mode, the record mark is located in the rightmost position of the last character in the receive area as shown below.

RECEIVE



Record Mark Effects

If character recognition starts a termination sequence and a record mark is sensed in main memory, the record mark is ignored and the termination sequence is completed. In receive areas, allowances should be made for an extra character location (maximum of two locations) after the record mark. Here the status indicators would indicate termination due to character recognition and record mark sensing.

PARITY

Character Parity

Lines designated not to have character parity do not modify data received from the central processor or adapter in any manner. Lines designated to have character parity carry the parity bit in the most significant bit position of the line code. Parity for any line is designated on a block basis by the parity bits stored in the present command. When a character is recognized, its parity is checked; however, the block parity bit determines whether or not the parity of the next character is to be checked.

In the transmit mode of operation, the communication control generates the parity as designated by the parity bits, then inserts it in the parity bit position of the character before the character is included in the generated block parity register. The state of the parity bit position as it is received from main memory does not in any way affect parity generation.

In the receive mode of operation, the communication control checks the parity of each character. If the checked parity differs from that specified in the present command, the control stores this fact and sends it to main memory at the time of the next internal interrupt from the line. In the case of lines designated to check parity, the line parity bit is deleted before the character is transferred to main memory.

Block Parity

The block parity feature provides a half-add sum (even modulo two sum) on a bit position basis for each line character transmitted or received. If the unit has a stored synchronization character for the line, synchronization characters are not included in block parity computation. The start-of-input-transfer (SIT) characters are not considered to be synchronization characters; they are included in block parity computation. Characters not transferred to main memory preceding the SIT characters are not included in block parity computation.

The block parity feature is specified on a per-line basis by the block parity (BP) bit. To provide flexibility, the BP bit is used in conjunction with the character termination bit when the block parity feature is used. Both the character termination and block parity features are used in conjunction with character recognition.

TRANSMIT MODE

Four combinations of the character termination bit and the block parity bit are used in the transmit mode.

1. When a one-character termination sequence with no block parity bit is specified, the recognized character is sent to the Type 285 interface register to be transmitted. The present command is terminated.
2. When a two-character termination sequence with no block parity bit is specified, the recognized character is sent to the Type 285 interface register, then a second character is transferred from main memory. After this second character is sent to the adapter interface register to be transmitted, the present command is terminated.
3. When a block parity bit and a one-character termination sequence is specified, the termination sequence is started by recognition of a character. After that character is transferred to the adapter interface register, the contents of the block parity register are moved to buffer storage. From buffer storage they are moved to the adapter interface register when that register becomes available. The present command is terminated.

4. When a block parity bit and a two-character termination sequence is specified, the termination sequence is started by the recognition of a character. After that character is transferred to the Type 285 interface register, a second character (block parity correction) is transferred from main memory. This second character is included in the block parity register, then the contents of the block parity register are moved to buffer storage. From buffer storage they are moved to the adapter interface register when that register becomes available. The present command is terminated.

Character parity generation is not performed on the block parity correction character. Recognition is inhibited on the block parity correction character as well as on the block parity character.

RECEIVE MODE

Four combinations of the character termination bit and the block parity bit are also used in the receive mode.

1. When a one-character termination sequence with no block parity bit is specified, the recognized character is transferred to the central processor. The present command is terminated.
2. When a two-character termination sequence with no block parity bit is specified, the recognized character is transferred to the central processor and the termination sequence begins. The next character received is transferred to the central processor. The present command is terminated.
3. When a block parity bit and a one-character termination sequence are specified, the recognized character is transferred to the central processor and the termination sequence begins. The next character received is included in the block parity register. Neither this last line character nor the contents of the block parity register are sent to the central processor, but the block parity error indicator is set if the block parity register does not equal zero at this time. When the block parity register has been checked, the present command is terminated.
4. When a block parity bit and a two-character termination sequence are specified, the recognized character is transferred to the central processor and the termination sequence begins. The next character received is included in the block parity register, then the contents of the block parity register are moved to buffer storage. From buffer storage they are transferred to the central processor. The block parity error indicator is set if the contents of the block parity register did not equal zero at the time of the transfer. Finally, the present command is terminated.

CHARACTER PARITY OF THE BLOCK PARITY CHARACTER

The parity bit of the block parity character can be handled in one of two ways:

1. The parity bit is the same as the character parity for the rest of the message. In the receive mode, the parity bit of the second character of a termination is deleted if a two-character termination sequence is called for.

2. The parity bit is the half-add sum of all the preceding parity bits of the message. In the receive mode, the parity bit of the second character of a termination is transferred to the central processor along with the character if a two-character termination sequence is called for.

The block parity character parity bit selects the method that is to be used. If the bit is set, step 1 is in effect; if the bit is reset, step 2 is in effect.

RESET BLOCK PARITY REGISTER

The block parity register is reset to zero by any command sequencing that results in a change in the mode of the unit. If a line goes into a No-Next-Command-Stored condition, and the line is returned by a new command back to the same mode it was in prior to the No-Next-Command-Stored condition, this is not considered a change in command as far as resetting the block parity register is concerned. The block parity register is also reset after a termination sequence caused by a character recognition.

Control Unit Parity

Each time a word is read from the control unit memory in the inhibit, off-line, or on-line operating state, its parity is checked. If there is an error, the parity error indicator is set and an interrupt is sent to the central processor. The "Parity Error" PCB instruction tests the error indicator. A "Reset Parity Error" PCB instruction is then executed to reset the error indicator and the interrupt caused by the error. The Halt/Continue function determines the action of the control unit when a control unit parity error occurs.

HALT/CONTINUE CONTROL

A Halt/Continue control is provided to enable the program to specify the action of the control unit when a control unit parity error occurs. Depressing the INITIALIZE pushbutton on the Series 200 control panel sets the communication control to the Halt condition. Two PCB instructions are provided to change the state of the communication control by program.

Halt Condition

When the Halt/Continue control is set to the Halt condition and a parity error occurs, the interrupt function is set, the Read/Write Channel is released, the Allow condition of the control unit is reset and the unit goes into the initialize state. When the "Parity Error" PCB is executed to test for a parity error, the program will branch. The "Reset Parity Error" PCB must be executed to reset both the error function and the interrupt caused by the error.

Continue Condition

When the Halt/Continue control is set in the Continue condition and a parity error occurs, the interrupt function is set and the unit continues to operate. The interrupt function from the control unit could be set by both a memory parity error and an indication that the status field

has been loaded. The parity error must be tested first. The "Parity Error" PCB to test for a parity error will branch. The "Reset Parity Error" PCB resets the error function and interrupt condition caused by the error. If the interrupt is tested again after the "Reset Parity Error" PCB instruction, it will branch indicating that the status field has new data. The status information is used and a "Status Taken" PCB is executed to alert the unit to the fact that the status field is available.

PERIPHERAL INSTRUCTIONS

Peripheral Data Transfer (PDT) Instruction

A PDT instruction is interpreted according to the state of the control unit. When the unit is in the load/test state, the PDT instruction is a data transfer type. When the unit is in the initialized or off-line state, a PDT instruction assigns the RWC to the unit and loads the status field address into the starting location counter. The format for the PDT instruction used in conjunction with the communication control is shown in Figure 4-2.

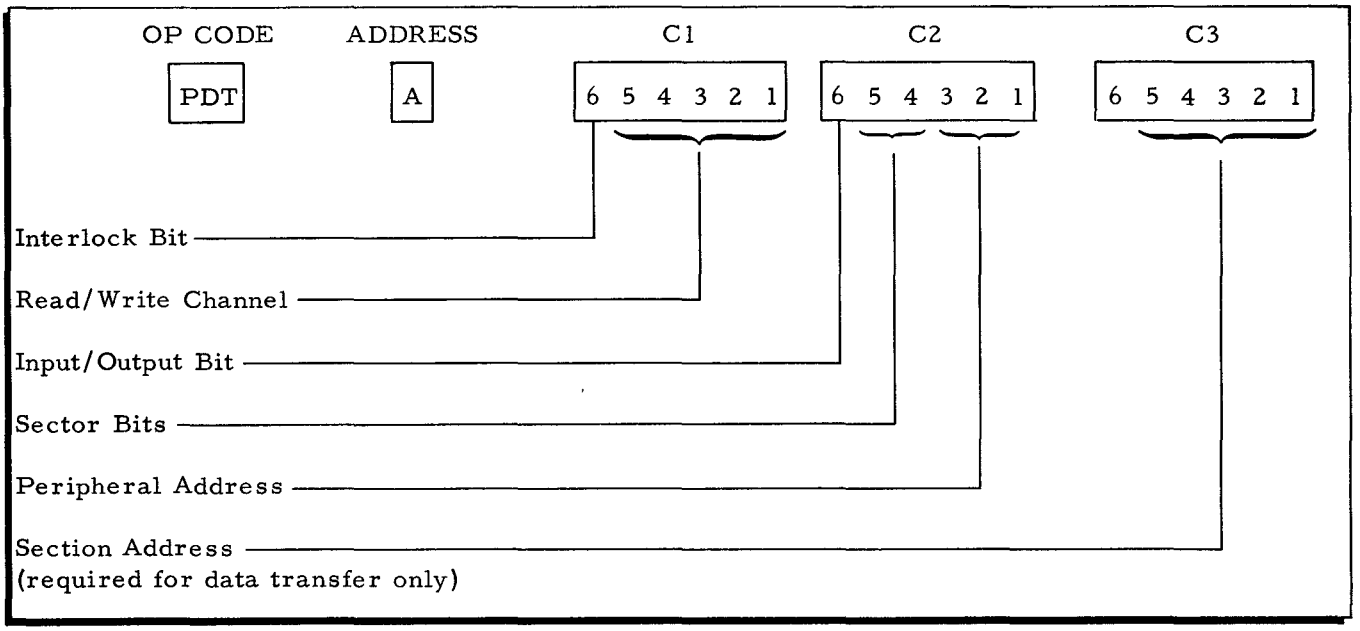


Figure 4-2. Format of PDT Instruction

A address — The A address interpretation is determined by the control unit:

Load/Test state: Specifies the leftmost character of the data field used by the PDT instruction.

Inhibit and On-Line operating state: Specifies the leftmost character of the status field used by the control unit to store the interrupt information.

Interlock Bit	— ONE = memory access not granted to alternate RWC. ZERO = memory access granted to alternate RWC.
Read/Write Channel (RWC)	— Selects the RWC between main memory and the control unit (see the appropriate Programmers' Reference Manual).
Input/Output Bit	— Indicates direction of data transfer in load/test state: ZERO = output, ONE = input. This bit is not interpreted for the PDT assigning an RWC to the unit.
Sector Bits (C2)	— Bits 5 and 4 designate the sector: sector 1 = 00, sector 2 = 10, sector 3 = 11.
Peripheral Address	— Logical address of the control unit on the peripheral bus.
Section Address (C3)	— This character is required when the control unit is in the load/test state, otherwise it is not used.

Control character C3 specifies the section where the transfer is to start and is loaded into the memory test address counter. The six least significant bits of the memory test address counter are reset at the beginning of each PDT instruction. The counter supplies the address to the memory address register. A record mark in the central processor main memory is required to terminate the transfer. The entire 2048 words of memory can be loaded or interrogated by a single PDT instruction. Address "wrap-around" can occur in the memory.

Peripheral Control and Branch (PCB) Instruction

The PCB instruction is the only means the program has to communicate with the control unit except for the two types of PDT instructions outlined above. There are three types of PCB instructions for performing the following functions:

1. Questions and control of the communication control unit.
2. Questions and control of a particular line.
3. Transfer control information to a particular line.

The general format for PCB instructions is shown in Figure 4-3.

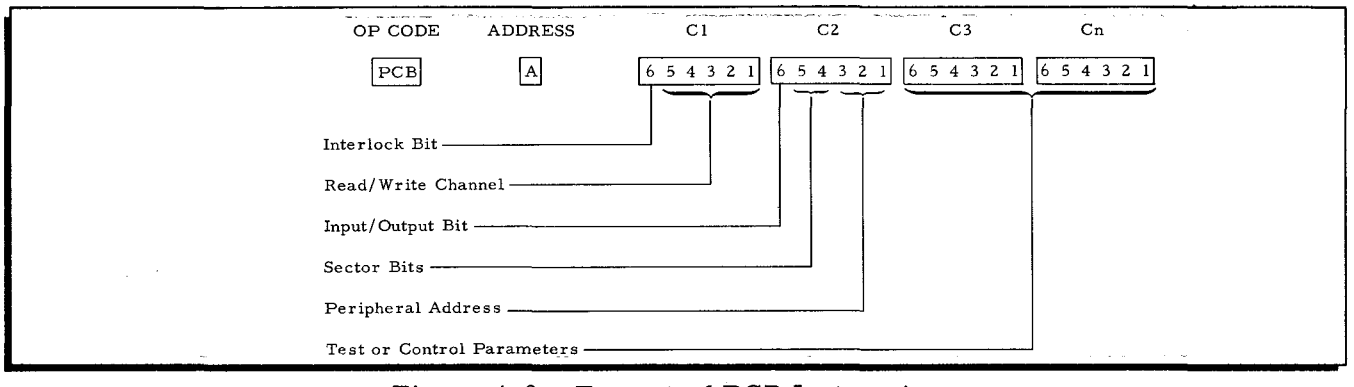


Figure 4-3. Format of PCB Instruction

A address — Dependent on the type of PCB, as follows:

- Branching PCB — The A address is the memory location to which the program will branch if the condition tested for by the PCB is true or the control unit cannot accept a control PCB.
- Nonbranching PCB — The address is not interpreted but one must be specified.

Interlock Bit — Not interpreted.

Read/Write Channel (RWC) — Selects the RWC being tested for availability and must be zero if the RWC is not to be tested (see the appropriate Programmers' Reference Manual).

Input/Output Bit — Not interpreted.

Sector Bits (C2) — Bits 5 and 4 designate the sector: sector 1 = 00, sector 2 = 10, sector 3 = 11.

Peripheral Address — Logical address of the communication control unit on the peripheral bus.

Control Characters:

- C1 — Should be zero except when a test for RWC activity is desired.
- C2 — The logical address address assigned to the control unit.
- C3 — Distinguishes between the different types of PCB's; control of the control unit or asking questions of the unit.

QUESTIONS AND CONTROL OF THE UNIT

The general format, using control character C3, is implemented with this type of PCB instruction. The format is as follows; chaining of C3 characters is permitted.

Communication control unit PCB C3 character coding for both questions and commands is shown in Table 4-1.

Table 4-1. Communication Control PCB C3 Character Coding

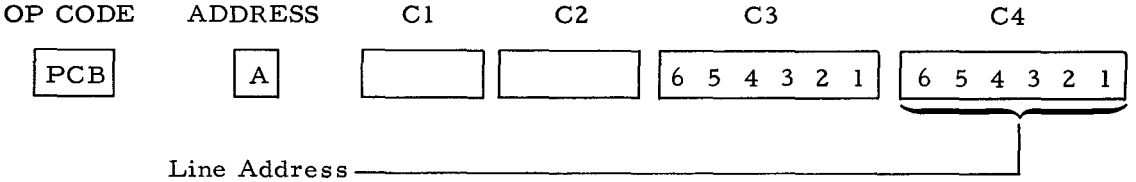
TYPE OF INSTRUCTION	BRANCHING	DESCRIPTION	(OCTAL)
		QUESTIONS	
Busy	Yes	Is the communication control busy (RWC assigned)? If yes, branch to A.	10
Parity error	Yes	Is the control unit parity error indicator set? If yes, branch to A.	40

Table 4-1 (cont). Communication Control PCB C3 Character Coding

TYPE OF INSTRUCTION	BRANCHING	DESCRIPTION	(OCTAL)
Interrupt	Yes	Did the communication control specified by the C2 variant cause the interrupt? If yes, branch to A.	75
		COMMANDS	
Set Allow	No	Sets the Allow function in the control unit to permit program interrupts from the unit.	71
Reset Allow	No	Resets the Allow function and prevents future interrupts.	70
Load/Test State	No	Places the control unit in the load/test state and allows a PDT instruction to transfer data to or from the unit.	25
Exit	No	Resets the load/test state.	24
Status Taken	No	Notifies the control unit that the status field is available for the next transfer and subsequent interrupt. This PCB resets the present interrupt if it was caused by an internal or external source or an address request.	74
Release RWC	No	The assigned RWC is released by a "Release RWC" PCB instruction addressed to the control unit. The release does not take place during a data transfer; i.e., once the RWC is loaded with an address the associated data is transferred then, if required, the RWC is released. If the status is being transferred when a "Release RWC" PCB instruction is executed, the transfer is completed and the interrupt function is set, then the RWC is released. Once the RWC is released, status information cannot be sent to the central processor, therefore interrupts cannot be sent from the control unit.	27
Halt	No	Places the Halt/Continue function in the Halt condition.	20
Continue	No	Places the Halt/Continue function in the Continue condition.	21
Reset Parity Error	No	Resets the parity error function and the interrupt caused by the error.	26

QUESTIONS AND CONTROL OF A LINE

The following is the only allowable format for this type of PCB; control characters cannot be chained.



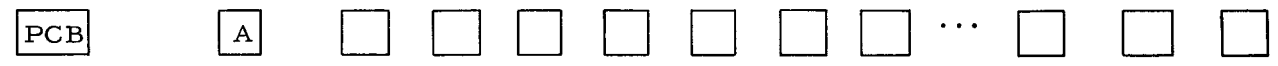
Coding of control character C3 in Questions and Control PCB's is shown in Table 4-2. All line questions and line commands shown are branching PCB's. These PCB instructions may have to be reextracted; this is controlled by the communication control unit. It is suggested that these be self-branching PCB instructions (the A address specifies the address of the instruction).

Table 4-2. Questions and Control PCB C3 Character Coding

TYPE OF INSTRUCTION	DESCRIPTION	C3 (OCTAL)
	LINE QUESTIONS	
Address Request	Request the address of the next transfer that is to take place from the designated line. (NOTE: This feature is available only when the Character Location option is present.)	36
	LINE COMMANDS	
Forced Sequence with Response	Aborts the present command and causes the next command to become the new present command. An interrupt is generated due to command sequencing.	33
Forced Sequence without Response	Same action as the Forced Sequence with Response instruction but the interrupt is not generated due to command sequencing.	32
RESYNC	Resets the SYNC conditions of the designated line (see Section II, "Syncronization for Synchronous Line," and "Start of Input Transfer for Asynchronous Lines").	37
Special Strobe	Provides the program with direct control of an adapter unit by activating the special strobe line to the adapter unit. Refer to the appropriate Type 281/285 Hardware Bulletins for the use of this special strobe line.	34

LINE CONTROL INFORMATION PCB

This type of PCB must have control characters C1 through C7. Additional control characters through C13 can be used but are only required if contents of the next registers are to be changed. Control characters C14 and C15 should only be used for asynchronous lines.



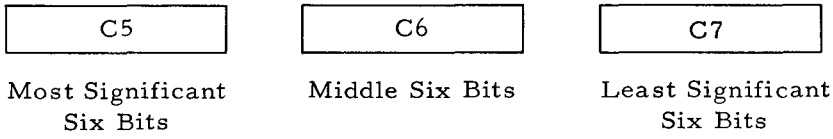
Coding of control character C3 in Line Control Information PCB's is shown in Table 4-3. All line commands are branching PCB's. These PCB instructions may have to be reextracted; this is controlled by the communication control. It is suggested that these be self-branching PCB instructions (the A address specifies the address of the instruction).

Table 4-3. Line Control PCB C3 Character Coding

TYPE OF INSTRUCTION	DESCRIPTION	C3 (OCTAL)
	LINE COMMANDS	
Next Command	Alerts the communication control that the PCB contains the next command information.	30
Next Command and Forced Sequence with Response	Alerts the communication control that this PCB contains the next command information and once it is loaded the present command is aborted and the information just loaded becomes the new present command. As a result of the command changes, an interrupt call is generated.	33
Next Command and Forced Sequence without Response	Same action as the next command and Forced Sequence with Response instruction but the interrupt is not generated.	32

The other control characters, C4 through C15, in the Line Control Information PCB's are used for the following purposes:

1. C4 is the line address control character.
2. C5, C6, and C7 are the characters which contain the central processor memory address, six bits in each character, that is used by the control to load the RWC when a data transfer is to be made, as shown below.



3. C8 and C9 control characters specify what action, if any, the line is to take. They are loaded into the next order sections of the memory. Bit arrangement is shown below; function and meaning of the individual bits in the C8 and C9 control characters are listed in Table 4-4.

4. C10 and C11 control characters contain the first recognition character in eight bits (two bits of C10 and six bits of C11).
5. C12 and C13 control characters contain the second recognition character in eight bits (two bits of C12 and six bits of C13). When the recognition characters are loaded by a "Line Control Information" PCB, the complete line character, including a parity bit, is loaded. Any ususued bits must be zero. The following is the ASCII control character ETX (End-of-Text) with odd parity in the PCB format to be loaded as the first recognition character.

CP bit position	C10 B	A	8	4	2	1	C11 B	A	8	4	2	1
ETX	X	X	X	X	1	0	0	0	0	0	1	1

X-these bits are transferred to the control unit.

6. C14 and C15 control characters contain the SIT character for asynchronous lines in eight bits (two bits of C14 and the six bits of C15). It should be noted that due to the delay in the control unit (interrupt queue), care must be taken when changing this character because the SIT character is loaded directly into the present command register and not into the next command register, as is the rest of the information contained in a "Line Control Information" PCB.

Table 4-4. Function of C8 and C9 Control Character Bits

BIT	MEANING OF C8 BIT
6, 5	Bits 6 and 5 specify the mode of operation of the line as follows: Bits 6 5 0 0 - Inhibit 0 1 - Receive 1 0 - Transmit 1 1 - Transmit repeat
4	Bit 4 is the allow-timer bit: 0 - The timer is not allowed. 1 - The timer is allowed.
3, 2	Bits 3 and 2 specify the character parity feature as follows: Bits 3 2 0 0 No character parity generation or checking is performed. 0 1 1 0 - Even character parity is specified. 1 1 - Odd character parity specified.
1	Bit 1 is the character transfer bit: 0 - Specifies one six-bit character transfer per line character. 1 - Specifies two six-bit character transfers per line character.

BIT	MEANING OF C9 BIT
6	Bit 6 is the response bit: 0 - No interrupt is generated at the termination of the command. 1 - An interrupt is allowed.
5, 4	Not used; must be zero.
3	Bits 3 is the block parity bit: 0 - The block parity feature is not used. 1 - The block parity feature is used.
2	Bit 2 is the command termination bit: 0 - The character that is recognized is the last transfer to or from the central processor before the command is terminated. 1 - One more data transfer is made to or from the central processor after the character that was recognized before the command is terminated.
1	Bit 1 is the parity bit: 0 - The parity bit will be the half-add sum of the preceding parity bit. 1 - The block parity character will have the same parity generated or checked as data characters.

STATUS FIELD

The status field associated with interrupts is shown in Figure 4-4. The status field is not used when the interrupt is caused by a control unit parity error. Punctuation is not required in this field.

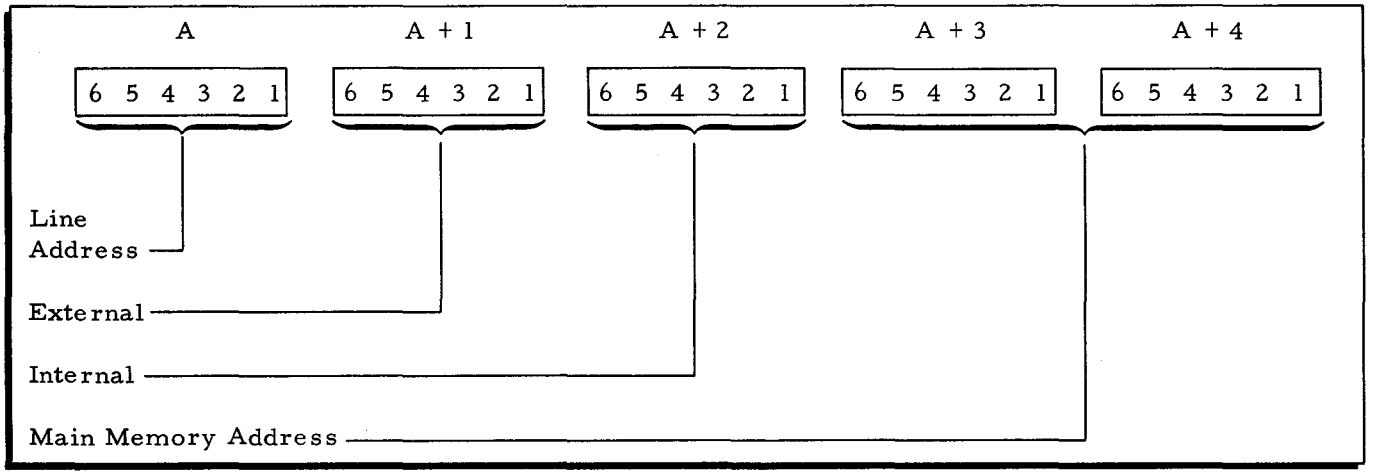


Figure 4-4. Status Field

Line Address — The character specifies the line that caused the interrupt.

External — If bit 6 is set (one) the interrupt is caused by external conditions. If bit 6 is not set, the cause of the interrupt is either internal or is a response to an address request. Bit 5, when set, indicates a timeout has occured on the line (no data has been moved by the bit end for at least eight seconds). Bits 1 through 4 are not used; they will be loaded as zeros.

Internal — The setting of each bit indicates a status condition:

Bit 1 — When set, indicates termination has occurred due to character recognition.

Bit 2 — When set, indicates termination has occurred due to finding a record mark in the central processor main memory.

Bit 3 — When set, indicates the block parity summation, in the receive mode, was not equal to zero. (This bit can only be set when the block parity feature is specified for the line in the command just terminated.)

Bit 4 — When set, indicates a received character was detected with bad parity.

Bit 5 — When set, indicates that no next command was available at the time commands were sequenced.

Bit 6 — When set, indicates an overwrite error has occurred. The character processor was unable to transfer received data to the central processor.

Main Memory Address — The address sent to the central processor is modulo 4096. The A+3 character includes the middle six bits of the stored address in the communication control and the A+4 character includes the least significant six bits.

Overwrite

When an overwrite error occurs, the character in buffer storage can not be transferred to the central processor before the next character is received from the line. The character in buffer storage is not altered in any way. All subsequent received characters, until buffer storage is available, are lost. Character recognition, parity checking, and block parity computation are not performed on the lost characters.

SCAN TABLE

The scan table controls the servicing sequence of the attached Adapter unit. The table must be organized to guarantee that data is not lost between the communication control unit and the adapters. Each entry in the scan table contains a six-bit line address and a two-bit line type indicator. Each time a line number appears in the table, its line type indicator must also be entered. The line type indicators are:

00 - Not assigned at this time.

01 - Start-stop line adapters.

10 - Synchronous lines — non-IBM STR series adapters.

11 - Synchronous lines — IBM STR series adapters.

The frequency of occurrence of a particular line in the scan table should be in proportion to its bit rate in relation to the total bit rate of the unit. The scan period, listed in Table 4-5, is defined as the minimum occurrence that a particular line of a certain rate must appear in a scan table (i.e., the 900-cps line should appear at least every seventh position in the table — based on a bit time of 138 μs and a scan time of 18 μs/bit; 18 x 7 = 126 μs).

Table 4-5. Line Scan Period

SPEED OF LINE		SCAN PERIOD
CHAR/Sec.	Bits/Sec.	Number of Lines
900	7200	7
600	4800	11
450	3600	15
300	2400	23
225	1800	31
150	1200	46
75	600	88
TELETYPE	< 150	requires only one entry in any scan table

The zero address must always be assigned to the slowest or one of the slowest speed lines and is the last entry in the table. The scan table is recycled after the zero line address is serviced. If the line zero is not active, it still must be entered to recycle the scan table. Assume the lines listed in Table 4-6 are connected to a control unit.

Table 4-6. Example: Lines in a Communication Network

SPEED		NUMBER OF LINES	TOTAL BITS	ADDRESS (OCTAL)
C/S	B/S			
10	110	4	440	0, 1, 2, 3,
300	2400	3	7200	4, 5, 6,
600	4800	3	14400	7, 10, 11
			22040	

$$\frac{\text{22040 total bit rate}}{\text{4800 bit rate of a 600 C/S line}} = 4.6 \approx 5$$

Each 600 C/S line number should be entered into the scan table as every fifth table entry.

$$\frac{\text{22040 total bit rate}}{\text{2400 bit rate of a 300 C/S line}} = 9.1 \approx 10$$

Each 300 C/S line number should be entered into the scan table as every tenth table entry.

$$\frac{\text{22040 total bit rate}}{\text{110 bit rate of 10 C/S line}} = 200$$

Each 10 C/S line number should be entered into the scan table at least once.

Table 4-7 is an example of what the scan table should look like after the 600 and 300 character per second line numbers are entered. The teletype line numbers are entered into the scan positions that are vacant.

Table 4-7. Partially Completed Scan Table

1	2	3	4	5
11	10	7	6	5
11	10	7	4	
11	10	7	6	5
11	10	7	4	
11	10	7	6	5
11	10	7	4	
11	10	7	6	5
11	10	7	4	

The completed scan table is shown in Table 4-8.

Table 4-8. Completed Scan Table

1	2	3	4	5
11	10	7	6	5
11	10	7	4	3
11	10	7	6	5
11	10	7	4	2
11	10	7	6	5
11	10	7	4	1
11	10	7	6	5
11	10	7	4	0 (recycle)

Note that if the scan table, using this method, exceeds the allocated 256 locations, the frequency of entrance of line numbers should be decreased until the scan table no longer exceeds 256 entries. The frequency of occurrence of a line number in the scan table for a given line speed should never be less than the scan period shown above.

TIMER

The timer associated with each line is considered an external source as far as interrupts are concerned. That is, if a line has an inhibit command as the present command, the timer will timeout if not overridden. If a line is allowed to get into the no command stored condition and the last present command overrides the timer, no timeout will take place. If the timer was allowed by the last command, timeout depends on the type and state of the line. For an asynchronous line, timeout could occur whether the line is in transmit or receive state. For synchronous lines, a timeout could occur only if the line was in the receive mode. If a line was in the transmit mode and there is no next command, the control unit will maintain synchronism by transmitting the stored SYNC character until a command is forced or the control unit is reset. The unit is reset by releasing the RWC, resetting the Allow condition, and putting the unit in the load/test state. A timeout can occur on an inhibited line as long as the line number is in the scan table.

The timeout bit of the status field indicates, when set, that no data has been moved (either transmitted or received) through the bit processor for at least eight seconds. A line number must be present in the scan table for the timer to function for a line. The timer can be inhibited during a present command by resetting the allow timer bit to zero. This feature is provided as a fail-safe timer; it is not intended to be an interval timer. The present command is not terminated when the timeout occurs. An interrupt caused by a timeout indicates to the program that the line has not moved data for a minimum of eight to a maximum of twenty-four seconds. The

stored commands are not modified by a timeout and no new commands are required unless the state of the line is to be changed. If no action is taken, another interrupt from this line will occur in 16 seconds, assuming no data is moved.

CHARACTER LOCATION FEATURE

The Character Location feature provides the address to which the next data character would be transferred, on a per line basis. The address is put in the status field locations A+3 and A+4, along with the rest of the status information prior to the setting of the interrupt caused by internal or external sources. The address transferred to the central processor is that of the location (least significant 12 bits) where the next data transfer would have accessed. For transfers of one character, the address is that of A+n, where A is the stored starting address and n is the number of characters transferred. For transfer of two characters, the address is that of A+2n, where A and n are the same as above.

The address may also be obtained by a PCB instruction. The request is stored in the interrupt queue and serviced, address transferred and Interrupt condition set, when the request becomes the active request in the queue. The status information is not transferred when the requested address character locations A+1 and A+2 are loaded with zeros. It should be noted that the interrupt queue is only 64 words in length. That is, only 64 interrupt requests can be stored at any given time. Care must be taken not to overflow the queue because no error indication is given.

SECTION V
PROGRAMMING INFORMATION

CONTROL UNIT MEMORY SECTIONS

The control unit memory is divided into 32 sections of 64 words each. Table 5-1 lists the name and octal address of each section. The address is that of the C3 variant character of the "Data Moving" PDT instruction in the load/test state. Twenty of the 32 sections are organized on a line-address basis, i.e., a line assigned an address, for example seven, has the information pertaining to that line as the seventh entry of each of the twenty line-oriented sections. The scan table and the four queue tables occupy the remaining sections. Prior to on-line operation, it is necessary to load the entire memory in order to load fixed characteristics and guarantee the state and parity of each word.

MEMORY LOADING

The load/test PCB instruction must be executed to place the unit in the load state so that a subsequent PDT(s) to load the memory is interpreted as a "Data Transfer" PDT. It is recommended that the entire memory be first loaded with zeros, then the scan table and fixed characteristics are loaded.

Scan Table Entries

There are four sections allocated for the scan table, but only the number of entries required for an installation need be loaded. This assumes the memory was first loaded with zeros. The first entry in the scan table must be loaded into the first word of section zero. If more than one section of the scan table is required, section one is used first, then section two and section three, in that order.

Table 5-1. Communication Control Memory Sections

ADDRESS (OCTAL)	NAME OF SECTIONS	LINE-ORIENTED SECTIONS
00	First section of scan table	
01	Second section of scan table	
02	Third section of scan table	
03	Fourth section of scan table	
04	Bit-to-character interrupt call, low priority	
05	Bit-to-character status, low priority	
06	Bit-to-character interrupt call, high priority	

Table 5-1 (cont). Communication Control Memory Sections

ADDRESS (OCTAL)	NAME OF SECTIONS	LINE-ORIENTED SECTIONS
07	Bit-to-character status, high priority	
10	Logic word	X
11	Adapter unit interface buffer	X
12	Bit order	X
13	Parallel buffer	X
14	SYNC character	X
15	Block parity	X
16	Character-to-bit interrupt call	
17	Character-to-bit status	
20	Present recognition character #1	X
21	Present recognition character #2	X
22	Character order (message)	X
23	Character order (character)	X
24	Next recognition character #1	X
25	Next recognition character #2	X
26	Next order (message)	X
27	Next order (character)	X
30	Present address (low-order bits)	X
31	Present address (middle bits)	X
32	Character-to-central processor status	
33	Present address (high-order bits)	X
34	Next address (low-order bits)	X
35	Next address (middle bits)	X
36	Character-to-central processor interrupt call	
37	Next address (high-order bits)	X

The format for scan table entries is as follows:

LINE TYPE	LINE ADDRESS
8 7	6 5 4 3 2 1

Line Type - The line type is designated by bits 8 and 7:

BITS

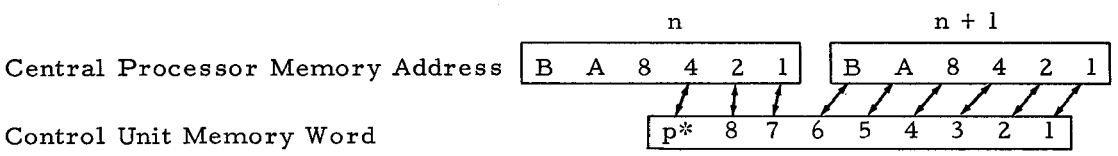
8 7

0 1 - Start-stop (asynchronous)
1 0 - SYNC (bit stream) — non-IBM STR series
1 1 - SYNC (bit stream) — IBM STR series

Line Address - The binary number of the line. (Assume line number 4):

Bits	6	5	4	3	2	1
Line Number	0	0	0	1	0	0

The corresponding bit positions of the central processor memory and the control memory areas follows:

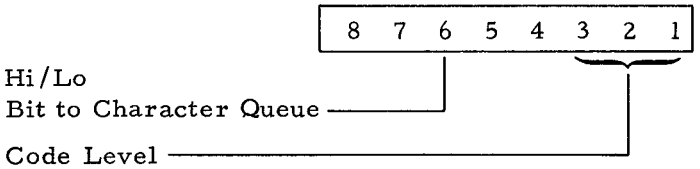


*The parity bit is only transferred to the central processor when the unit is in the load/test state, during a receive PDT instruction.

Fixed Characteristics Entries

There are three fixed characteristics which must be loaded.

1. Bit Order - Section 12:



Code level Bits 3 2 1

0 0 0 - 8 level
0 0 1 - 7 level
0 1 0 - 6 level
0 1 1 - 5 level

Hi/Lo Bit to Character Queue Bit 6

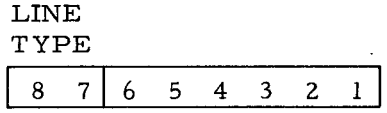
0 - specifies the low speed queue table

1 - specifies the high speed queue table

All other bits are loaded as zero.

2. SYNC/SIT Characters - Section 14. The SYNC/SIT characters are stored in section 14 in the same bit positions as a SYNC character for that line would appear in a data block in the central processor memory.

3. Character Order (character) - Section 23:



Line Type Bits 8 and 7

0 1 - start/stop (asynchronous) line

1 0 - SYNC (bit stream) - Non-IBM STR series

1 1 - SYNC (bit stream) - IBM STR series

All other bits are loaded as zero.

"Exit" PCB Instruction

Once the memory is loaded, the communication control is returned to the initialize state by the execution of an "Exit" PCB instruction. This PCB must not be issued until the control is no longer busy, indicating that data transfer is complete.

PLACING THE CONTROL UNIT ON-LINE

Once the memory is loaded and returned to the initialized state, the RWC is assigned to the control. It is recommended that the RWC remain assigned for the duration of communication control operation. When the RWC is assigned, the control is capable of responding to bit activity. Due to initialization, all the present commands are set to the inhibit mode, therefore, bit activities are not honored.

In order to place a line in the transmit or receive mode (also transmit-repeat), it is required to load the next command register and cause a forced sequence via a "Line Control Information" PCB instruction. A forced sequence overrides the response bit in the present command. Lines are not set to a data-moving command during initial loading because of the complexities involved in setting up the memory. The Allow condition need not be set prior to loading the next and present command registers.

INTERMEDIATE MODE

Each line of the unit will be in one of four modes when the unit is operating. These modes are: (1) transmit; (2) receive; (3) transmit-repeat; and (4) inhibit. An intermediate mode also exists; no command stored. This intermediate mode is entered by the character processor when there is no next command stored when a sequence change occurs. The bit processor remains in the mode it was in prior to the sequence change. If the mode of the line is not changed, although the line goes into the no command stored mode, the block parity register is not reset.

Note that it is possible to receive an interrupt from a line with the no-next-command indicator set even though a next command was sent to the line. This can happen due to the storage of the queue and the program not keeping up with the line. If this occurs, a forced sequence is required prior to sending new control information to the next command register. The action of a line in the no-command-stored mode for transmit and receive is as follows:

1. If an asynchronous line was in transmit, the last character is transmitted and the adapter unit will not take any further action. The timer, if not

inhibited, would timeout. In the case of a synchronous line, with no new command, the bit processor would remain in the transmit state and, being unable to obtain a new character, would transmit the stored SYNC character until a new command is forced into the present command register. The timer in this case would not time out.

2. If the line was in receive and no new command is present after a sequence change, no data is transferred to the central processor because a new address is not available. The overwrite indicator would be set by the second character received under this condition. In both the transmit and receive cases, a "Forced Sequencing" PCB instruction is required to load the present command register.

LINE CONTROL PROCEDURE

In describing the procedure that must be followed to control a line, it is assumed that the unit is in the on-line operation state, the RWC is assigned, the Allow condition is set, and the designated line is in the inhibit mode.

Start Transmission

Once the decision is made to transmit a message, a "Line Control Information" PCB instruction is sent to the communication control. The C3 variant character indicates that the PCB is "Control and Forced Sequencing without Response." This is done to load the next command register, then force the next command to become the present command. The no response is used to prevent a needless interrupt. If the entire message is contained within the area called out by the address, no other action is required. If a long message is to be transmitted and it is in several segments in memory, a second PCB is sent after the first. This second PCB would load the next command register. Only the address characters need be sent, if the recognition characters (when required) were loaded with the first PCB.

Continual Transmission

If a message is composed of many segments (more than two), in response to an interrupt from the line indicating change of commands when a record mark is sensed, a new "Line Control Information" PCB is sent with the address of the next segment of the message. This process, providing the next segment of the message, continues until the program senses the last segment of the message. The last segment address is sent to the next command register. In response to the next interrupt, indicating that the last segment is now being transmitted, the program makes the decision whether to transmit a new message, wait for a reply, inhibit the line, or do nothing.

Transmission Termination

If hardware block parity is not used, transmission can be terminated by using a record mark after the last character of the message or by recognizing a character of the message and specifying the proper termination sequence. If there is no block parity character, a one-character

termination sequence is specified. If the program provides the block parity character, the two-character termination sequence is specified.

When hardware block parity is used, transmission is terminated by character recognition only. The one-character termination sequence is used if the hardware-generated block parity character is to be transmitted as generated. The two-character termination sequence provides for handling a block parity correction character. This character is in the next character position (following the recognized character) in the data block. The block parity correction character is included in the block parity computation as received, no parity generation is performed, and the bit in the parity bit position is not altered. This character is not transmitted; the block parity character is transmitted.

Reception

In order to receive any information from an adapter unit, it is necessary to have a stored receive command. For synchronous lines, the unit must first get into synchronization with the line; thereafter, all non-SYNC characters will be transferred to the processor. For asynchronous lines, all information received for the adapter unit (provided there is no SIT character stored), is transferred to the processor. When a SIT character is stored, only the SIT character and all subsequent characters are transferred to the central processor. If hardware block parity is not specified, no special action is required at the beginning of the message. If hardware block parity is used, the block parity register must be cleared before the message is received. A forced sequence with a change in the mode of the line resets the block parity register.

Continual Reception

In response to each interrupt, a new next command (to receive data) would be sent to the unit. The command (C8 and C9 variant characters) and recognition characters need not be included in each PCB after they have been stored once. If character recognition is not required, zeros must be stored in the two character recognition words. The process of storing receive commands would continue until all the data is received or a character is recognized.

End of Reception of a Message

Reception can be terminated by a record mark in memory if the line has a fixed or known message length. When variable-length messages are being received, they will usually be terminated by character recognition. The termination sequence is controlled by the block parity bit and the command termination bit. Any subsequent action depends on the line format.

INTERPRETATION OF THE STATUS FIELD

The bit configuration of the status field is explained in Section IV (see "Status Field").

The following steps describe what the status field will be when an interrupt is sent from the communication control.

1. Control Unit Parity Error - The status field may not be valid (see "Halt/ Continue Control," Section IV). The line number is loaded for each of the following causes of interrupt.
2. Timeout - Bits 6 and 5 of the external character are set. The remaining bits of the external character and the entire internal character are zero.
3. When termination is due to a record mark or character recognition:
   a. Either bit 1 or 2 or both will be set. The no-next-command-stored bit can be set.
   b. Error indicators may also be set, but only if the line was in the receive mode.
4. Forced Sequence - Error indicators may be set, but only if the line was in the receive mode. The no-next-command-stored bit may be set. All remaining bits of the internal character will be zero as is the external character. Two characters of the address would be received along with the interrupts in steps 2, 3, and 4.
5. Address Request - Both the external and internal characters will be zero and the two-character address would be present.

CHARACTER LOCATION FEATURE

An address received in response to an Address Request, along with a timout interrupt or when the no-next-command bit is set, is obtained from the present command register and is the address that is transferred at the time of the interrupt and not at the time of the request. The address transferred at the time of any other interrupt (termination of a command and forced sequence is obtained from the next command register. The control uses the next command register as temporary storage until the transfer can be made. Note that if a forced sequence is sent to a line followed by a "Line Control Information" PCB instruction loading the next command register without an intervening interrupt from that line, the address received along with the interrupt, due to the forced sequence, is that of the "Line Control Information" PCB and not the address that was in the present command at the time of termination.

COMPUTER-GENERATED INDEX

HONEYWELL EDP TECHNICAL PUBLICATIONS
USERS' REMARKS FORM

TITLE: SERIES 200 TYPE 286-4 AND TYPE 286-5 MESSAGE-MODE, MULTI-CHANNEL COMMUNICATION CONTROLS HARDWARE BULLETIN

DATED: JUNE, 1966

FILE NO: 112.0005.1124.0-270

ERRORS NOTED:

Fold

SUGGESTIONS FOR IMPROVEMENT:

Fold

FROM: NAME _____ DATE _____

COMPANY _____

TITLE _____

ADDRESS _____

_____

Cut Along Line

FIRST CLASS
PERMIT NO. 39531
WELLESLEY HILLS
MASS.

BUSINESS REPLY MAIL

No postage stamp necessary if mailed in the United States

POSTAGE WILL BE PAID BY

HONEYWELL
ELECTRONIC DATA PROCESSING DIVISION
60 WALNUT STREET
WELLESLEY HILLS, MASS. 02181

ATT'N: TECHNICAL COMMUNICATIONS DEPARTMENT

Cut along Line

Honeywell
ELECTRONIC DATA PROCESSING