DATE: 12/11/79
PROJECT: DDCMP MULTIPLEXER
CHARGE NO: M416-E0300-37622

SYSTEM DESIGN SPECIFICATION

AND

USERS GUIDE

REV 1.1
BY
D.E. KORF

TABLE OF CONTENTS

1.0 INTRODUCTION

1.1 PURPOSE

The purpose of this document is to provide a design overview of KMC DDCMP multiplexer and its interaction with a host PDP11 processor. It is also intended to provide the PDP11 user with the required knowledge to interface the PDP11 host to the KMC multiplexer.

1.2 REQUIREMENT SUMMARY

Detailed below is a summary of the Requirements, Goals and non goals of the DDCMP multiplexer. The Functional Specification should be referred to for a complete definition of multiplexer functionality.

KMC DDCMP MULTI-LINE CONTROLLER

* WHAT ARE THE MAJOR OBJECTIVES OF MULTIPLEXER?

- Provide 8 lines at full or half duplex with speeds up to 9600 BPS.
- DDCMP communications protocol implemented in firmware providing high reliability and throughput with low host processor overhead.
- Provide communications between KMC11 and other synchronous interfaces that support DDCMP protocol in a point to point environment.
- 16 Bit NPR (DMA) transfers for minimum interference with host processor operation.
- Firmware based on KMC11-B so that the firmware is loadable.

* WHAT ARE ITS GOALS?

- Line level compatability with present DMC11.
- Support greater than seven (7) outstanding messages.

* WHAT THE MULTIPLEXER IS NOT

- Not a DMC11 modified to JUST scan 7 more lines.
- Will not provide support of line speeds greater than 9600 BPS.
- Will not support local on board diagnostics. (Note: Since KMC is loadable, diagnostics can be separately loaded into the KMC).
- Will not support automatic recovery from powerfail.

* DESIGN GOALS AND CONSIDERATIONS

- CSR control is the same as COMM IOP-DUP. Note commands differ but interface control is the same.
- CSR transfers from KMC to host PDP11 take priority over CSR transfers from PDP11 to KMC.
- Body of all messages reside in PDP11 host.
- All message headers reside in KMC.
- All inquiry control messages reside in KMC.
- System is primarily a state system.
- The states in the system are controlled by a series of queues, tables and buffers.
- Data reception takes priority over data transmission.
- Attempt to make the system somewhat self regulating (It won't accept more work than what it can handle).

1.3 DESIGN GOALS AND CONSIDERATIONS

Listed below are some design considerations of the DDCMP multiplexer in relation to other products, interfaces and internal control. The speed calculations define the maximum slice of time or state size that may be executed for each line per character per second.

SPEED CALCULATIONS

KMC = 5,000,000 instructions per second

$$\text{CHARACTER RATE} = \frac{8 \text{ lines X } 9600 \text{ BPS}}{8 \text{ bits per character}} = 9600 \text{ characters/second}$$

$$\text{INSTRUCTIONS/CHAR/SEC} = \frac{5{,}000{,}000 \text{ instr/sec}}{9{,}600 \text{ char/sec}} = 520.8 \text{ instr/char/sec}$$

520.8 instructions/character/sec for 1/2 duplex line.

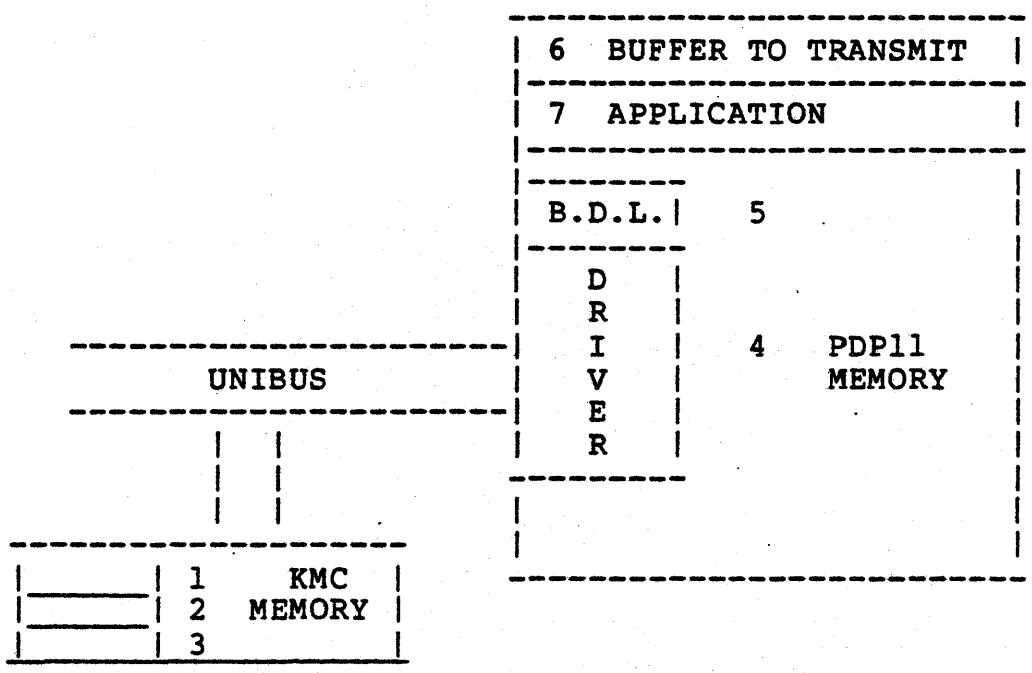
NOTE: The 520.8 instruction/character/sec is for a one way (1/2 duplex) line to arrive at a full duplex two way communication divide 520.8 by 2.

$$\text{FULL DUPLEX} = \frac{520.8}{2} = 260 \text{ instructions/character/second}$$

2.0 USERS VIEW OF SYSTEM

Below is a visual description of the system from the PDP11 users point of view. All major components of the system are identified.

VISUAL DESCRIPTION OF SYSTEM



1 - LINE STATUS TABLE IN KMC (LST)
CONTAINS CURRENT LINE DEFINITIONS

2 - LINE OUTPUT STATE TABLE IN KMC (LOST)
CONTAINS CURRENT STATE, HEADER AND
NEXT CHARACTER INFO

3 - INTERMEDIATE BUFFER IN KMC (IB)
CONTAINS BUFFER OF 8 CHARACTERS
TO BE TRANSMITTED

4 - DRIVER INTERFACE IN PDP11

5 - BUFFER DESCRIPTOR LIST AREA IN PDP11

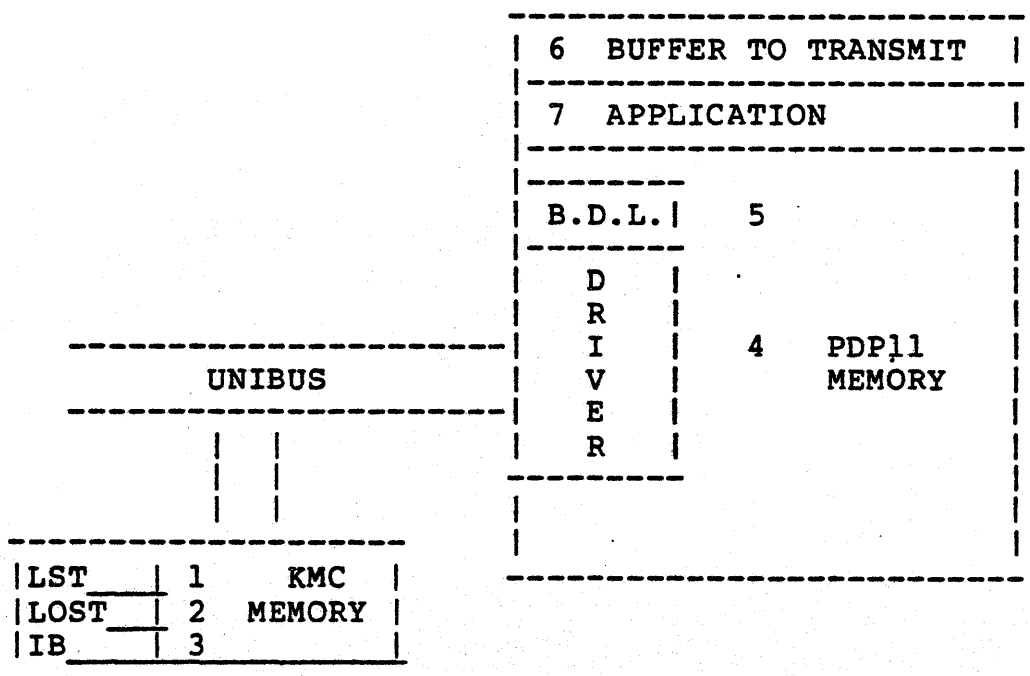
6 - DATA BUFFER TO BE TRANSMITTED

7 - APPLICATION PROGRAMS

2.1 USERS VIEW OF WHAT HAPPENS

The following is a description of the general processes that occur from a users point of view in the transmission of a message. The reception of a message has a similar process to that of transmission.

WHAT HAPPENS



Application program (7) passes to driver (4) a request for a message to be transmitted. It specifies the BDL MEMBER (5) that contains the address and character count of the buffer to be transmitted.

The driver (4) passes the information to the KMC.

The KMC first builds the header information in the L.O.S.T. Table (2).

The KMC then retrieves the BDL Member (5) which defines where in PDP memory the data buffer (6) is and how big it is.

The KMC then retrieves the first eight characters of the data buffer (6) and places them in the intermediate buffer (1).

The KMC then transmits the header information from the L.O.S.T. (6) table and updates the line status table (3).

After the header has been transmitted, a header CRC is transmitted.

Following the header CRC transmission, a character is retrieved from the intermediate buffer (1) and transmitted and a character from the data buffer (6) is requested to take its place (remember 7 other characters will be transmitted before the character requested is transmitted). The L.S.T. (3) is updated as is required.

The previous step is repeated until all characters have been transmitted. the data CRC is then generated and the LOST (2) is updated.

Upon reception of an ACK or NAK, (after seven retries), the BDL (5) member is set as free and a notification is sent to the driver (4) as to what has happened and the KMC tables and buffers 1,2,3 are updated.

The driver (4) then passes the KMC response status to the application program.

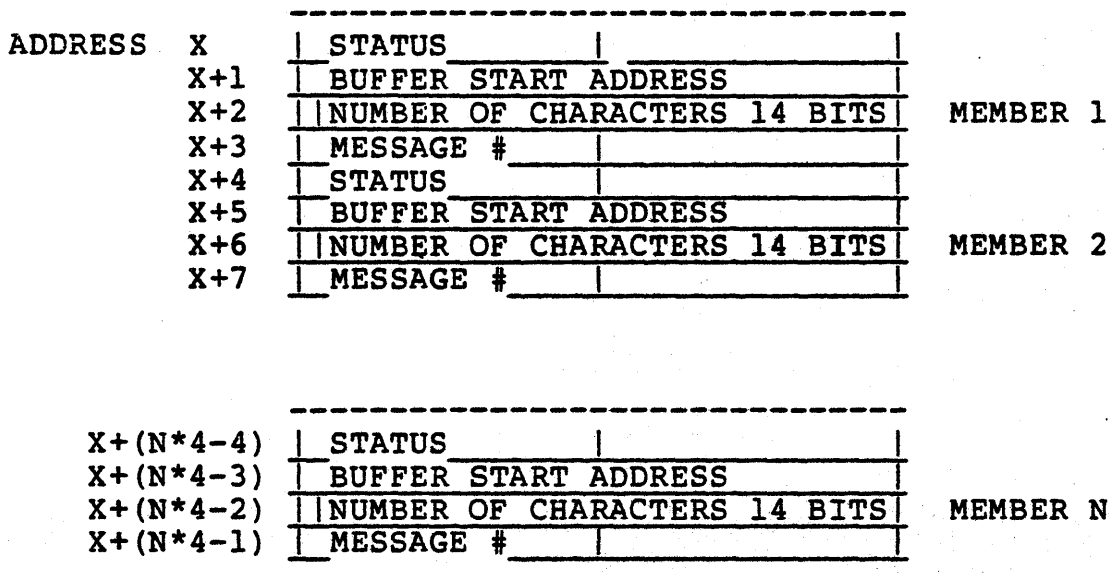
3.0 PDP11 INTERFACES

This section discusses the interfaces between the KMC and PDP11. There are three areas of interaction. The Buffer Descriptor List Definition Table in the PDP11, CSR Input Commands to the KMC from the PDP11 and CSR Response Commands from the KMC to the PDP11.

3.1 BUFFER DESCRIPTOR LIST (BDL) DEFINITIONS

In the PDP11 an area must be set aside to hold definitions of messages to be transmitted or received. What is a Buffer Descriptor List? It is a four word entry in PDP11 memory that defines to the KMC and the PDP11 where in PDP11 memory a data message is, how many characters are in the message, the line number and tributary of the line the message is to be transmitted or was received on, the DDCMP message number that was assigned by the KMC protocol handler and the status of the four word BDL entry (member). During initialization a BDL base address is passed to the KMC which defines where in memory the BDL Definition table starts. All references to this area of memory after setting the base is by member (four word entry) number. It is up to the user to define the size of this area. Its minimum size must be large enough to handle the number of input and output lines. For an eight line full duplex operation, the minimum size would be 64 words. That is eight lines for input at four words per line and eight lines for output at four words per line, assuming all lines are transmitting and receiving data at the same time. A definition of the Buffer Descriptor list and how it is organized follows.

PDP11 BUFFER DESCRIPTOR LIST DEFINITION TABLE



MESSAGE # = DDCMP Message Number from KMC Controller

STATUS = The status area is for use by the PDP11 for keeping track of which BDLs are in use and assigned to transmit and receive lines which are free.

BUFFER START ADDR = Start Address of Message Data to be Transmitted.

LENGTH = Character Count of Message to be Transmitted.

Length of BDL list is user defined. Maximum of 256 members.

3.2 CSR INPUT COMMAND TO KMC

The KMC and PDP11 communicate with each other via Control Status Registers (CSRs). Since both the KMC and PDP11 can read and write the CSRs, a control mechanism is required to prevent one CPU from writing while the other CPU is reading.

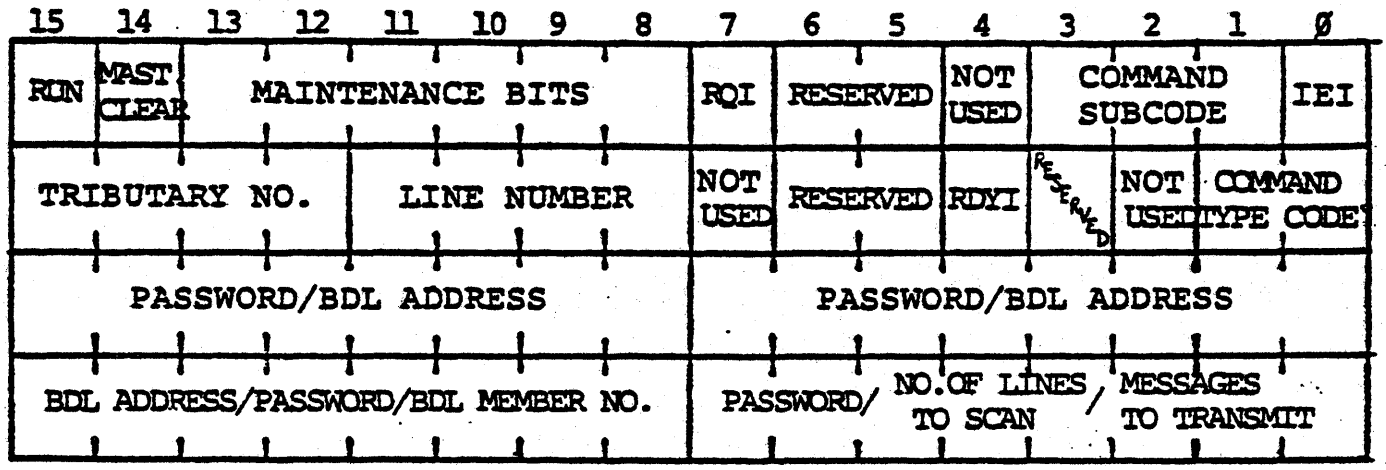
All input commands to the KMC are issued by an application program to a driver program which is completed in a series of steps. The driver program sets RQI to request the use of the CSR for transfer of data and then waits for the KMC to set RDYI. This wait can be implemented through a delay loop or can wait for an interrupt. The delay loop is not recommended. The KMC will then set RDYI when it is ready to accept an input. After the RDYI has been set by the KMC.

The PDP11 driver loads the CSRs with the command and its associated parameters. When the parameters have been loaded into the CSR, the RDYI is cleared to inform the KMC that the parameters may be read.

There are four (4) major input command types to the KMC. The command types and CSR structures are defined in the following figures.

KMC CONTROL INPUT COMMANDS

GENERAL FORMAT



COMMAND TYPES

COMMAND VALUE	COMMAND DESCRIPTION
0	Master Control
1	Line Control
2	Message Control
3	Status

MASTER CONTROL COMMANDS

COMMAND VALUE	COMMAND SUB CODE	COMMAND DESCRIPTION	PARAMETERS
0	0	Init System	N/A
0	1	Reset/Set to DDCMP Mode	LN #, TRIB #
0	2*	Enter MOP Mode	LN #, TRIB #, PASSWORD
0	3	Terminate Activity	LN #, TRIB #
0	4	Set Line Scan Length (No. of Lines to Scan)	LN #
0	5*	Set MOP Mode Password for Forced Entry	LN #, PASSWORD
0	6	Set BDL Base Address	BDL ADDRESS, TRANS., RCV.
0	7	Set Modem CSR Address (Note, also enables modem control for all lines.	CSR Address of modem control lower 18 bits

* Currently not planned for implementation.

LINE CONTROL COMMANDS

COMMAND VALUE	COMMAND SUBCODE	DESCRIPTION	PARAMETERS
1	0	Set Line Up	LN # (TRIB #)
1	1	Set Line 1/2 Duplex	LN #
1	2	Set Line Down	LN # (TRIB #)

MESSAGE CONTROL COMMANDS

COMMAND VALUE	COMMAND SUBCODE	DESCRIPTION	PARAMETERS
2	0	Transmit Message	LN #, BDL Member # of messages, TRIB #
2	1	BDL Member for use in Message Reception	BDL Member, LINE #

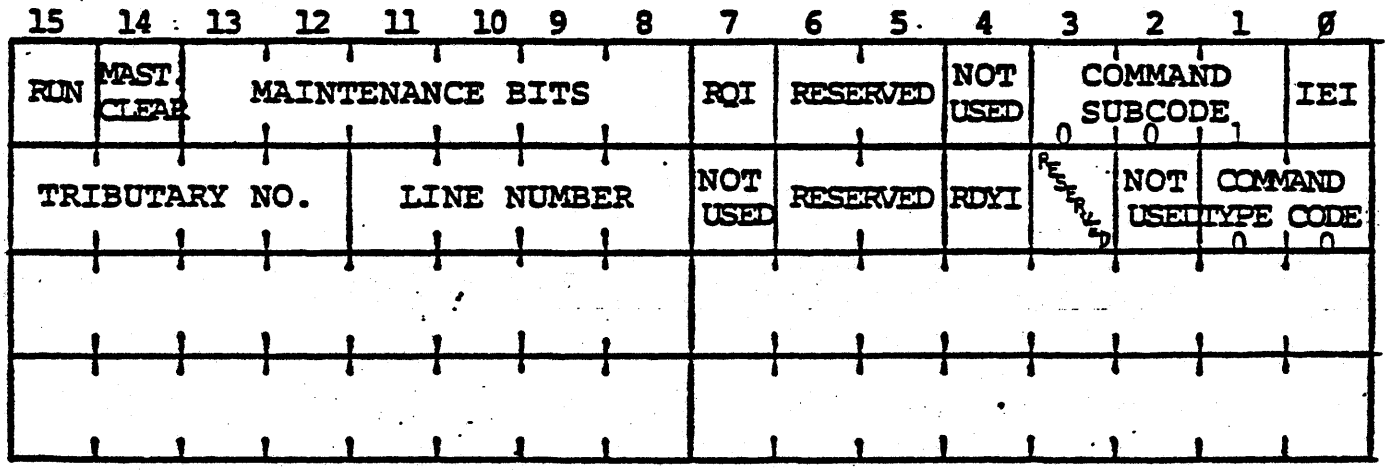
STATUS CONTROL COMMANDS

COMMAND VALUE	COMMAND SUBCODE	DESCRIPTION	PARAMETERS
3	0	Request Line Status	(LN #, (TRIB #1))
3	1	Read KMC Memory	KMC Data Memory Address
3	2	Write KMC Memory (Note an automatic read of the data store will take place for verification.	KMC Data Memory Address & Value to be stored.

*Currently not implemented.

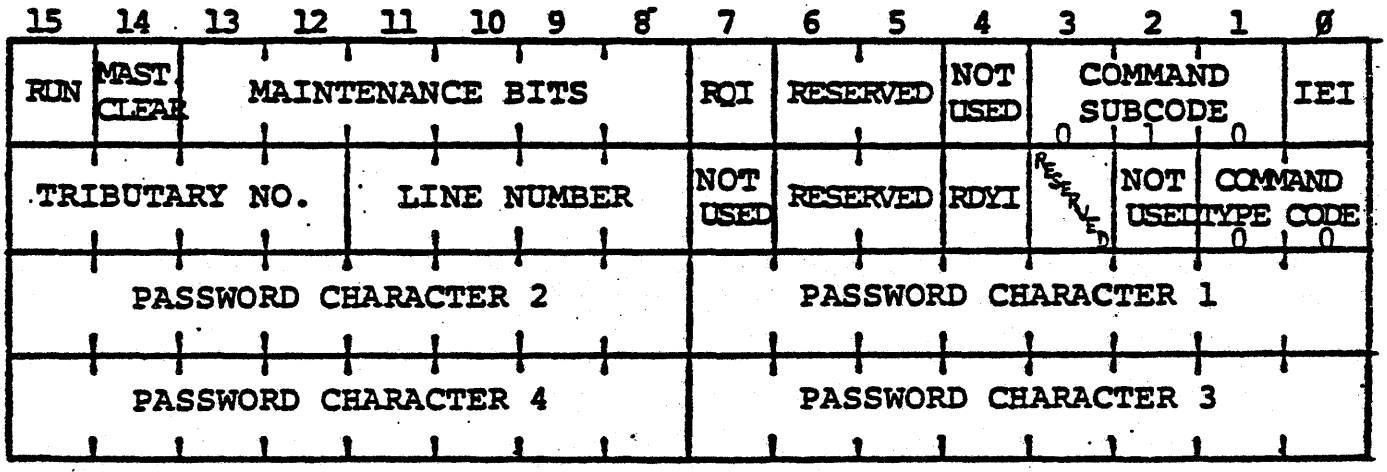
KMC CONTROL INPUT COMMANDS

RESET/SET DDCMP MODE



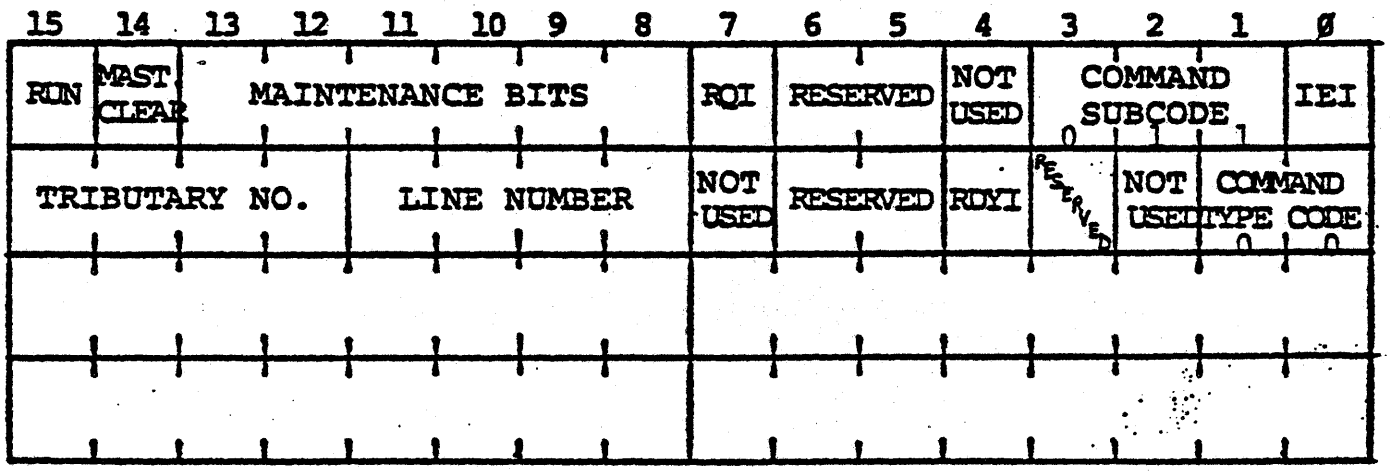
KMC CONTROL INPUT COMMANDS

ENTER MOP MODE



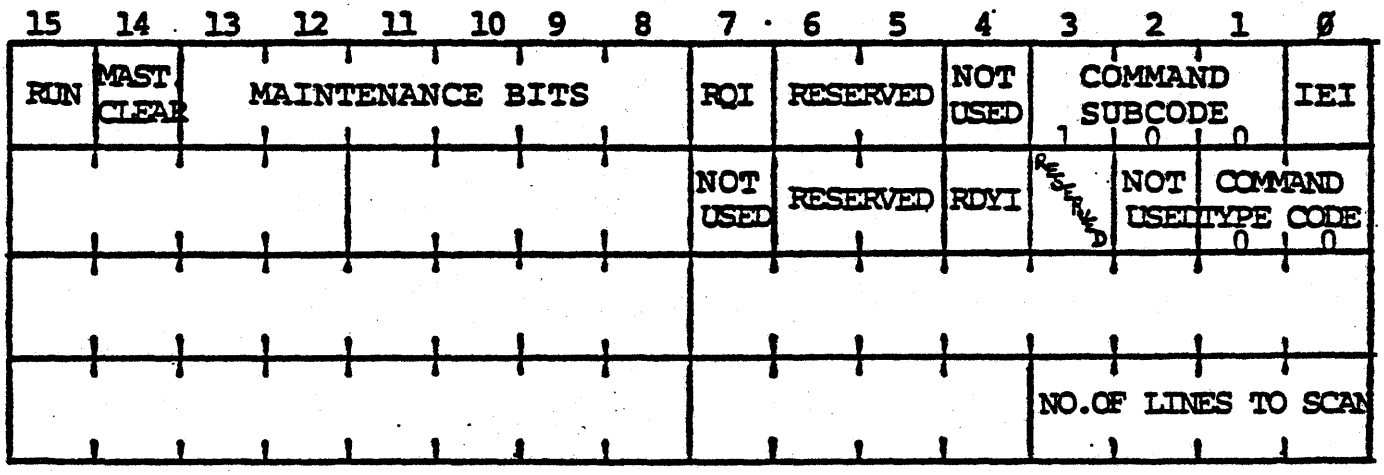
KMC CONTROL INPUT COMMANDS

TERMINATE ACTIVITY



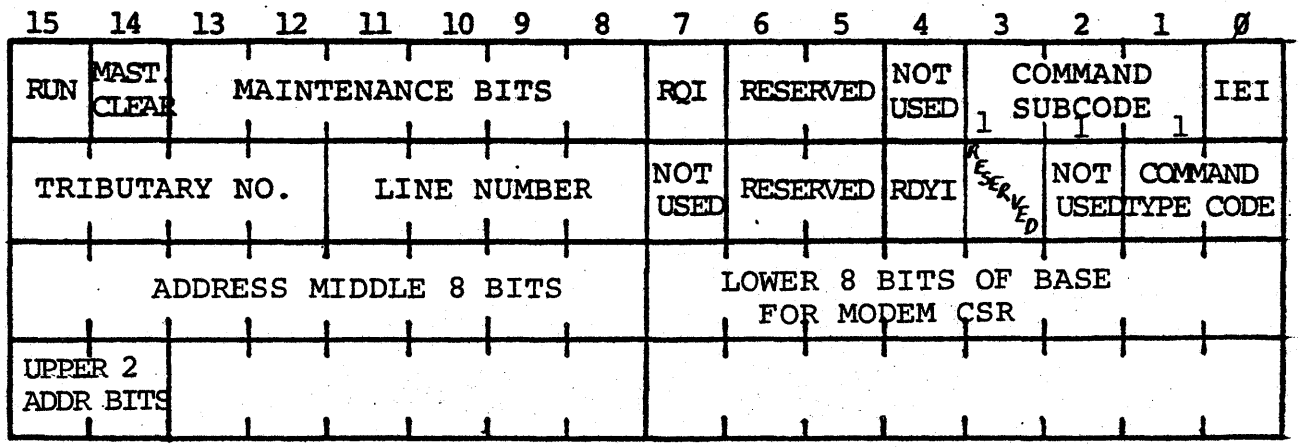
KMC CONTROL INPUT COMMANDS

SET LINE SCAN LENGTH



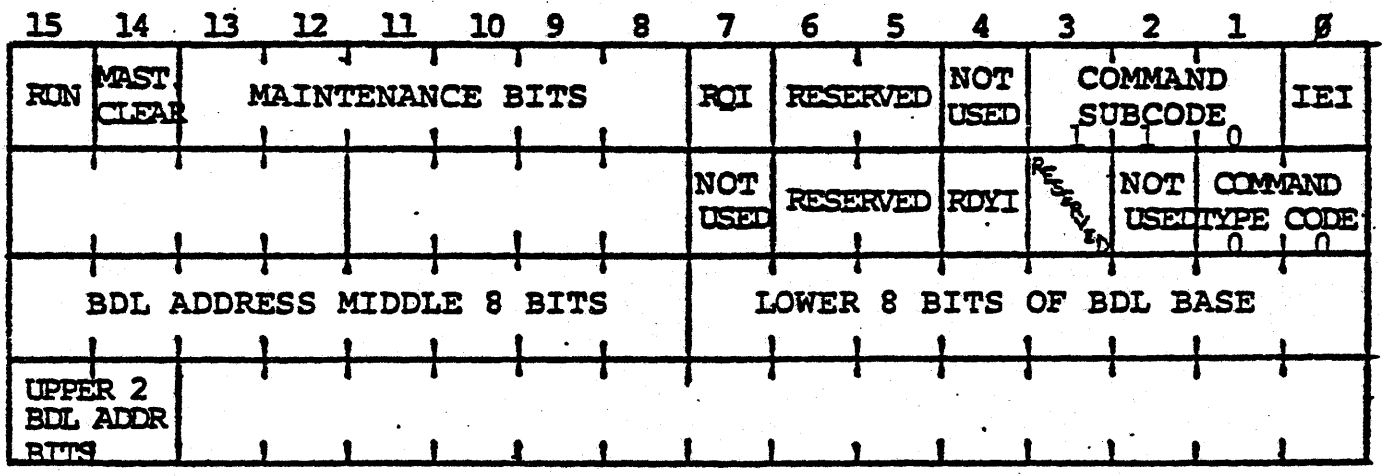
KMC CONTROL INPUT COMMANDS

MODEM CONTROL CSR ADDR



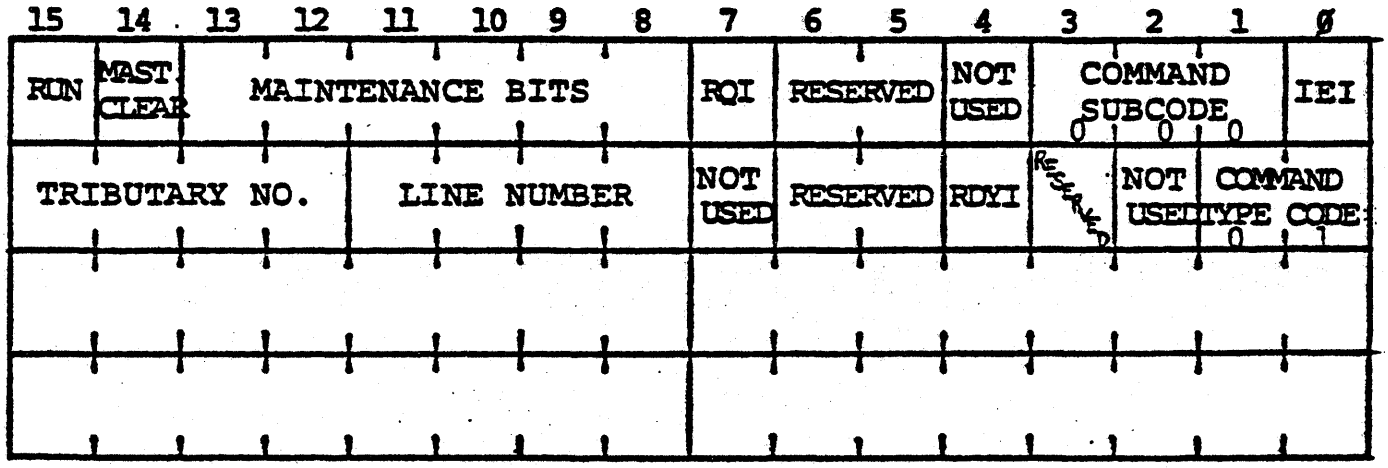
KMC CONTROL INPUT COMMANDS

BDL BASE ADDRESS



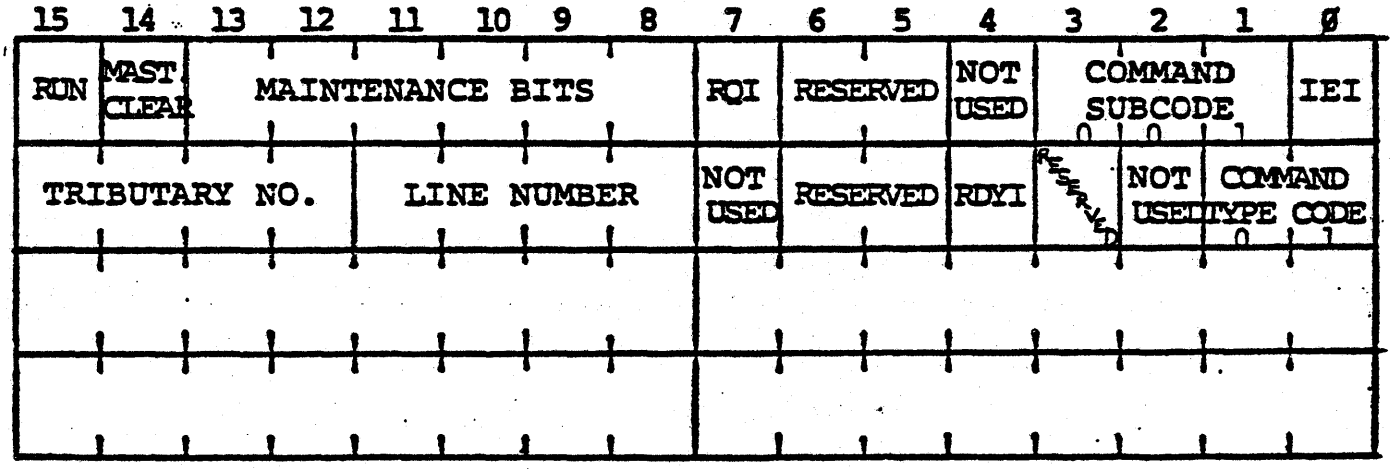
KMC CONTROL INPUT COMMANDS

SET LINE UP



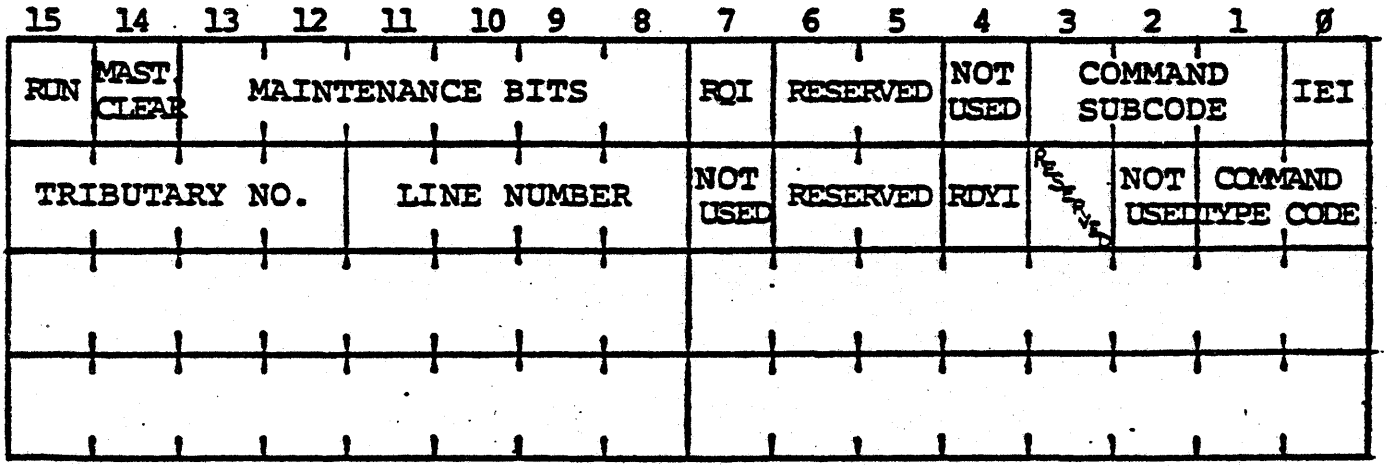
KMC CONTROL INPUT COMMANDS

SET LINE ½ DUPLEX



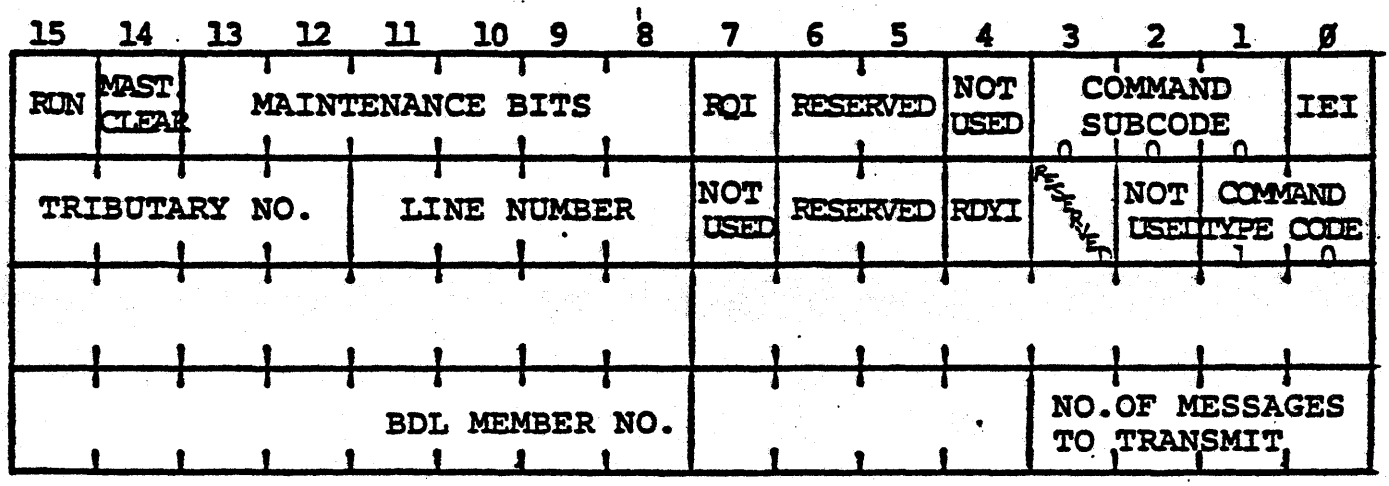
KMC CONTROL INPUT COMMANDS

SET LINE DOWN



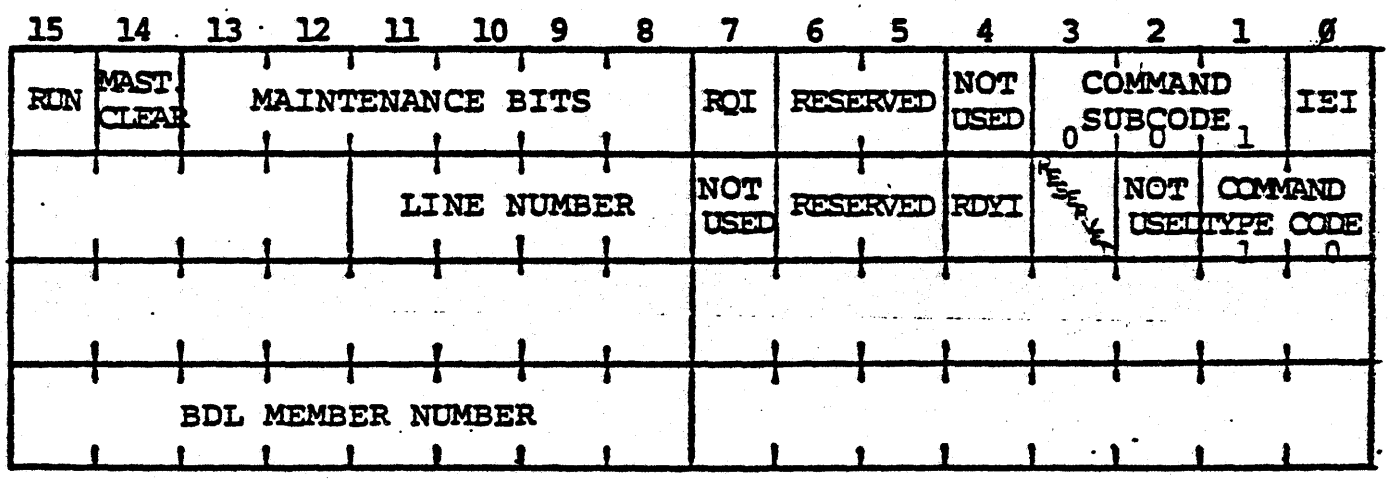
KMC CONTROL INPUT COMMANDS

TRANSMIT MESSAGE



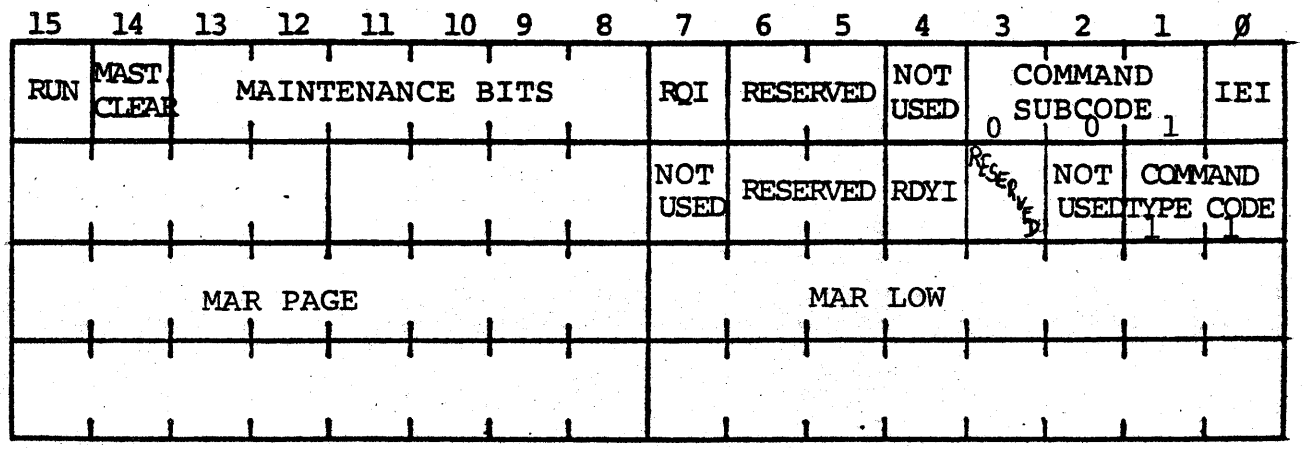
KMC CONTROL INPUT COMMANDS

BDL MEMBER FOR USE IN MESSAGE RECEPTION



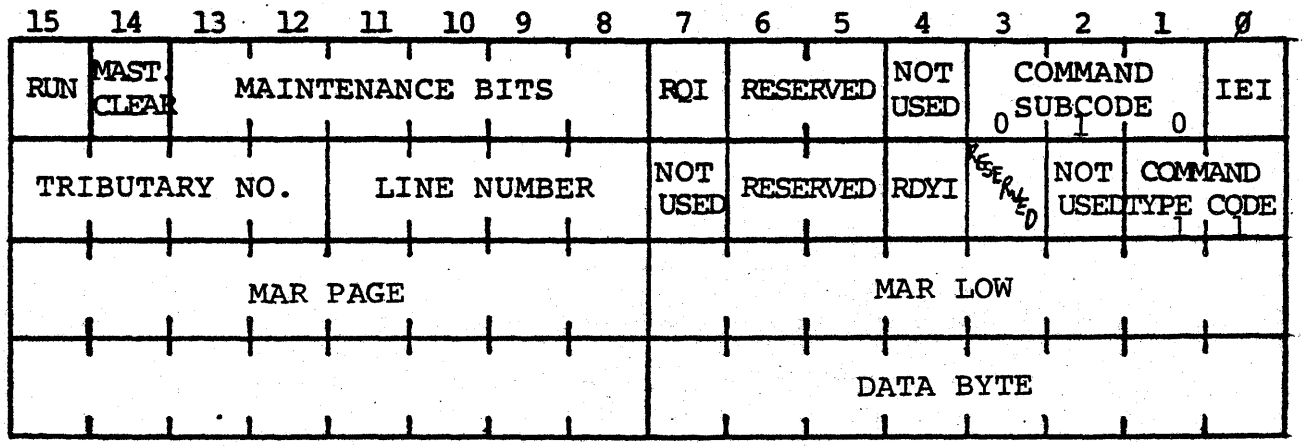
KMC CONTROL INPUT COMMANDS

READ KMC11 DATA MEMORY



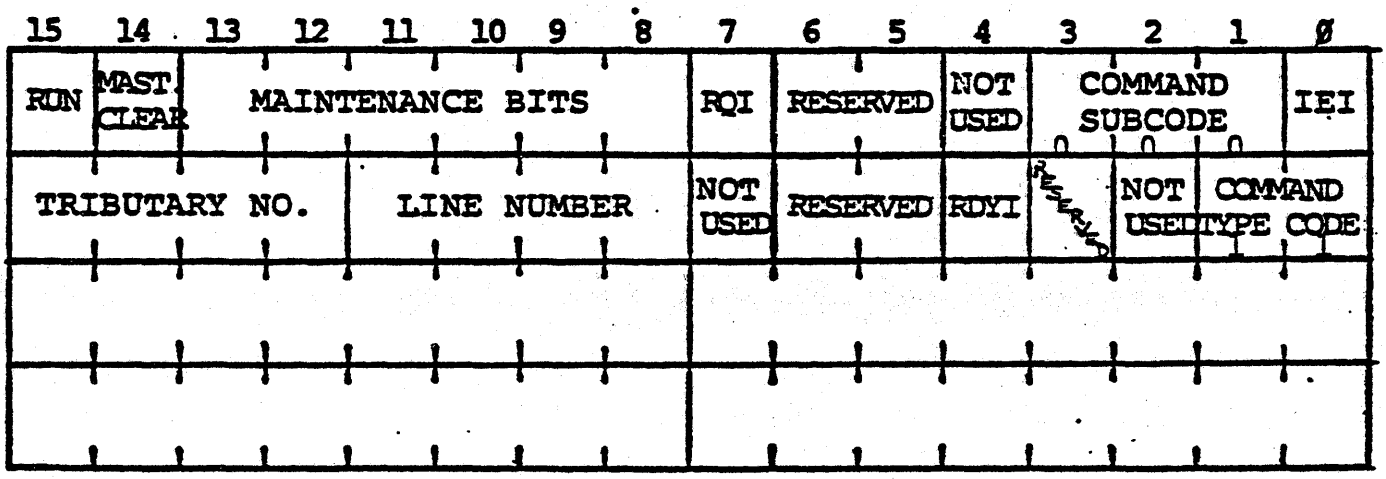
KMC CONTROL INPUT COMMANDS

WRITE KMC11 DATA MEMORY



KMC CONTROL INPUT COMMANDS

REQUEST LINE STATUS



3.3 CSR RESPONSE COMMANDS FROM KMC

Corresponding to the CSR input commands there are a series of output commands. Commands from the KMC to the PDP11 always take priority over an input command request. The method used for implementing output commands are as follows:

The KMC loads the CSRs with the parameters required for a given command;

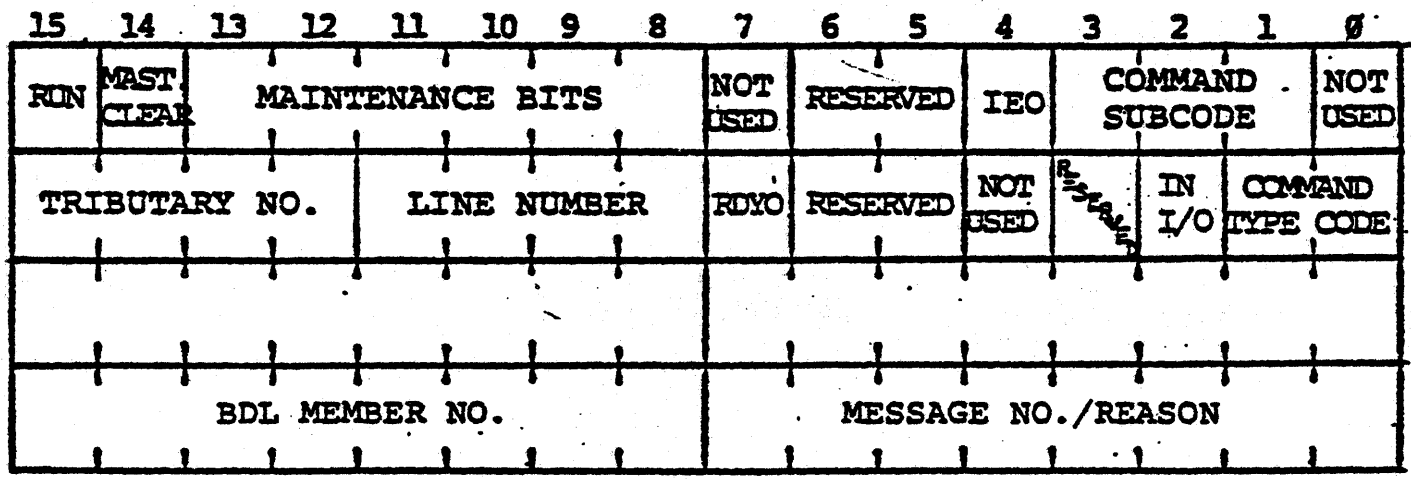
the RDYO bit is set and an interrupt is generated to the PDP11 indicating the PDP11 should read the CSRs;

the driver programs retrieves the CSR data and clears the RDYO bit to indicate the transfer is complete.

There are four (4) major output command types from the KMC. The command types and CSR structures are defined in the following figures.

KMC CONTROL OUT COMMANDS

GENERAL FORMAT



COMMAND TYPES

COMMAND VALUE	DESCRIPTION
0	Positive Responses Control
1	Negative Responses Control
2	Message Reception Control
3	Status Responses (Does Not Use General Format for Out Commands)

POSTIVIE RESPONSE CONTROL

COMMAND VALUE	COMMAND SUBCODE	DESCRIPTION	PARAMETERS
0	0	Init Complete	N/A
0	1	Set/Reset Complete	LN #, TRIB #
0	2	Activity Terminated	LN #, TRIB #
0	3	MOP Mode Entered*	LN #, TRIB 3
0	4	Message Acknolwedged	Message #, BDL #

NEGATIVE RESPONSE CONTROL

COMMAND VALUE	COMMAND SUBCODE	DESCRIPTION	PARAMETERS
1	3	Error Threshold Reached	LN #, TRIB #
1	5	Messaged Naked	LN #, TRIB #, BDL Member
1	6	Returned too many receiver buffers for line	LN #, MEMBER #

MESSAGE RECEPTION CONTROL

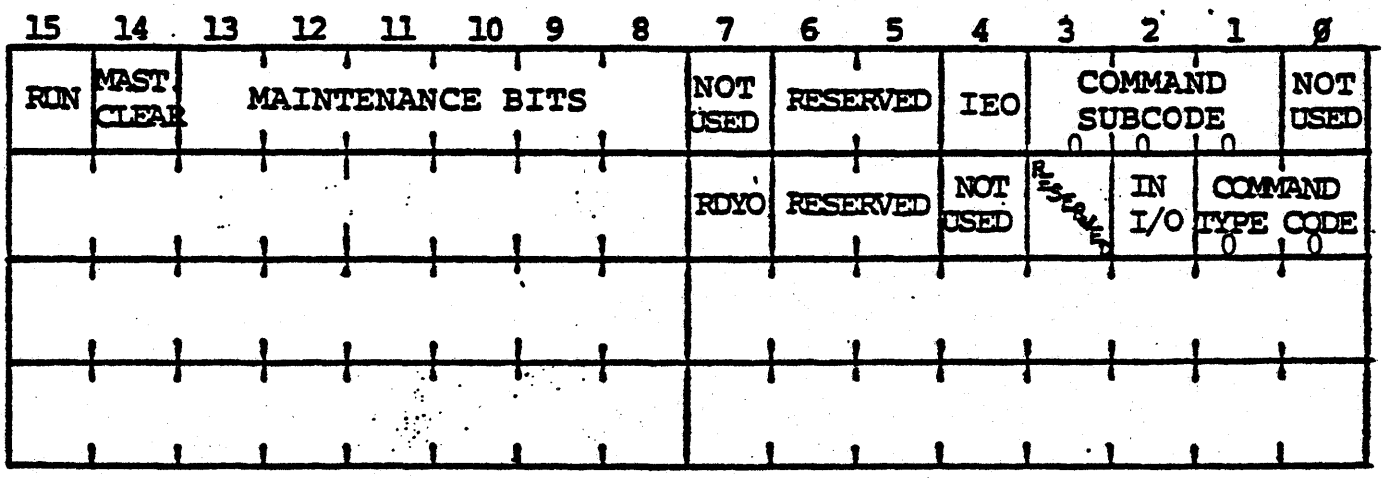
COMMAND VALUE	COMMAND SUBCODE	DESCRIPTION	PARAMETERS
2	0	Message Received	LN #, TRIB #, BDL MEMBER MSG #
2	1	Request BDL Member for Reception of Message	LN #

STATUS

COMMAND VALUE	COMMAND SUBCODE	DESCRIPTION	PARAMETERS
3	0	1st 4 bytes of line status table for specific line.	LIN# TRIB Bytes 0-3 of LST for line.
3	1	Last 4 bytes of line status table for specific line.	LIN# TRIB Bytes 4-7 of LST for line.
3	2	Dump of specified KMC Data Memory	Data Memory Address & Contents

KMC CONTROL OUT COMMANDS

INITIALIZATION COMPLETE

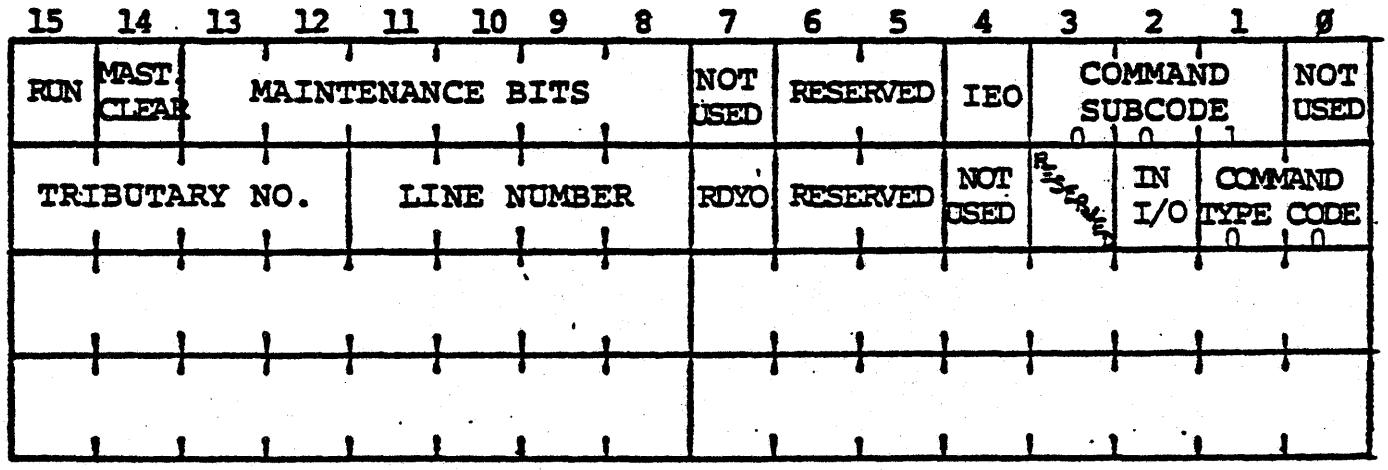


COMMAND TYPES

COMMAND VALUE	DESCRIPTION
Ø	Positive Responses Control
1	Negative Responses Control
2	Message Reception Control
3	Status Responses - (Does Not Use General Format for Out Commands)

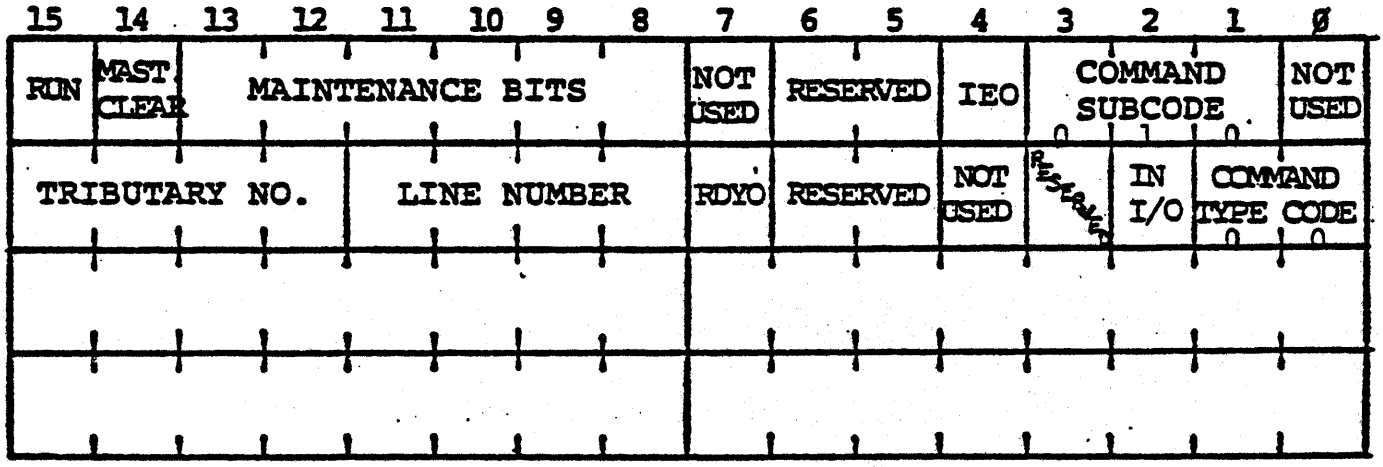
KMC CONTROL OUT COMMANDS

SET/RESET COMPLETE



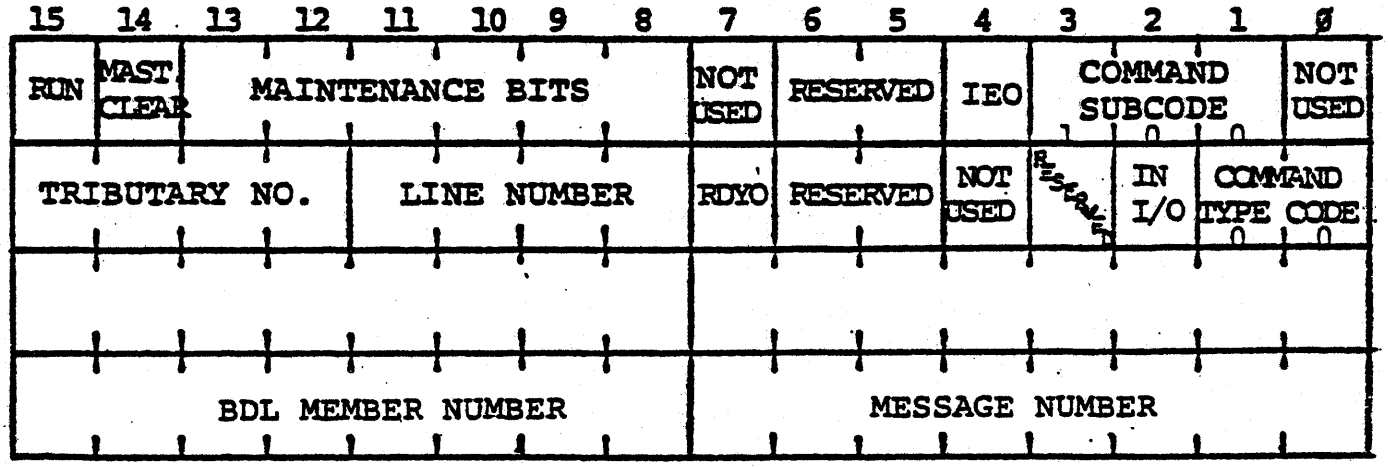
KMC CONTROL OUT COMMANDS

ACTIVITY TERMINATED



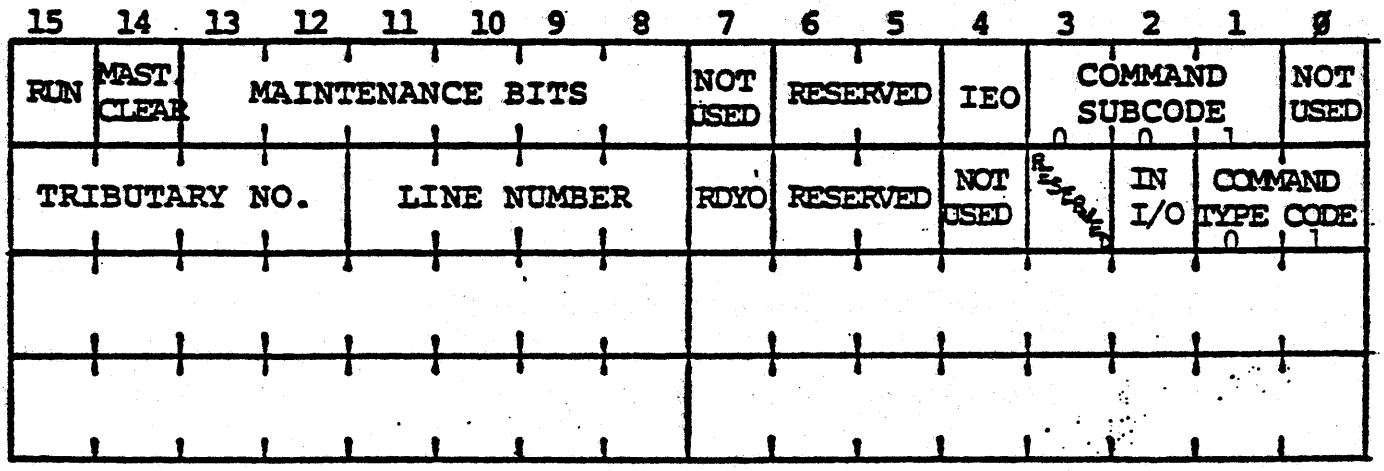
KMC CONTROL OUT COMMANDS

MESSAGE ACKNOWLEDGED



KMC CONTROL OUT COMMANDS

SET/RESET FAILED



KMC CONTROL OUT COMMANDS

ERROR THRESHOLD REACHED

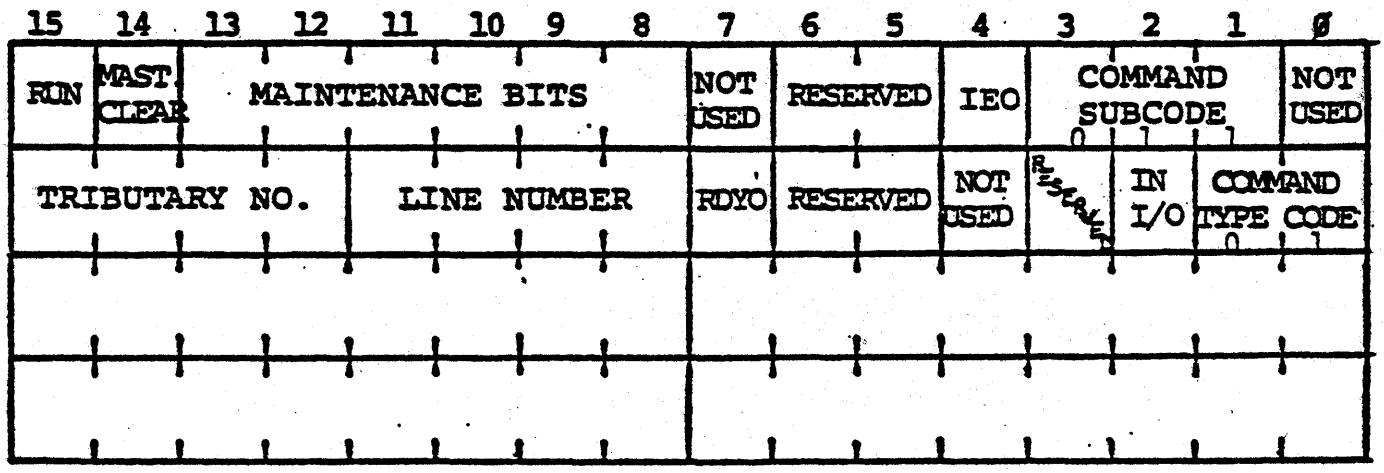
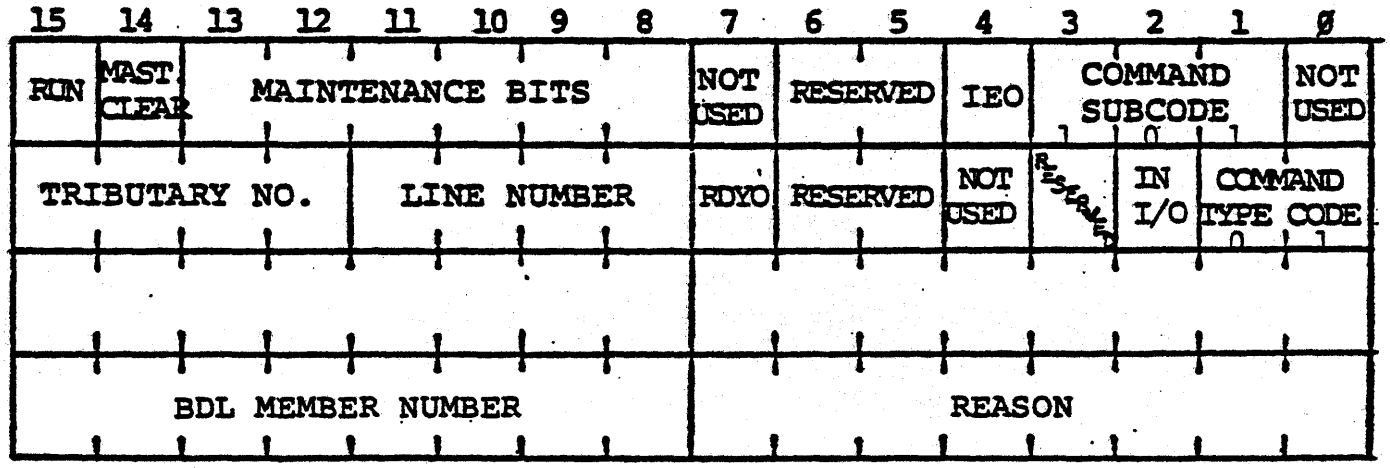


TABLE OF CONTENTS

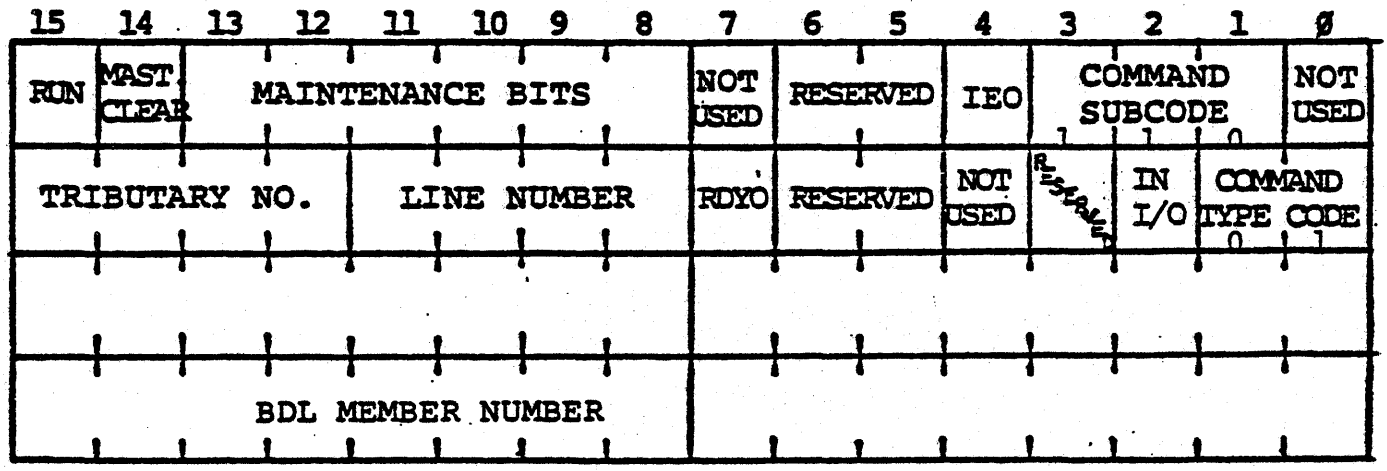
KMC CONTROL OUT COMMANDS

MESSAGE NACKED



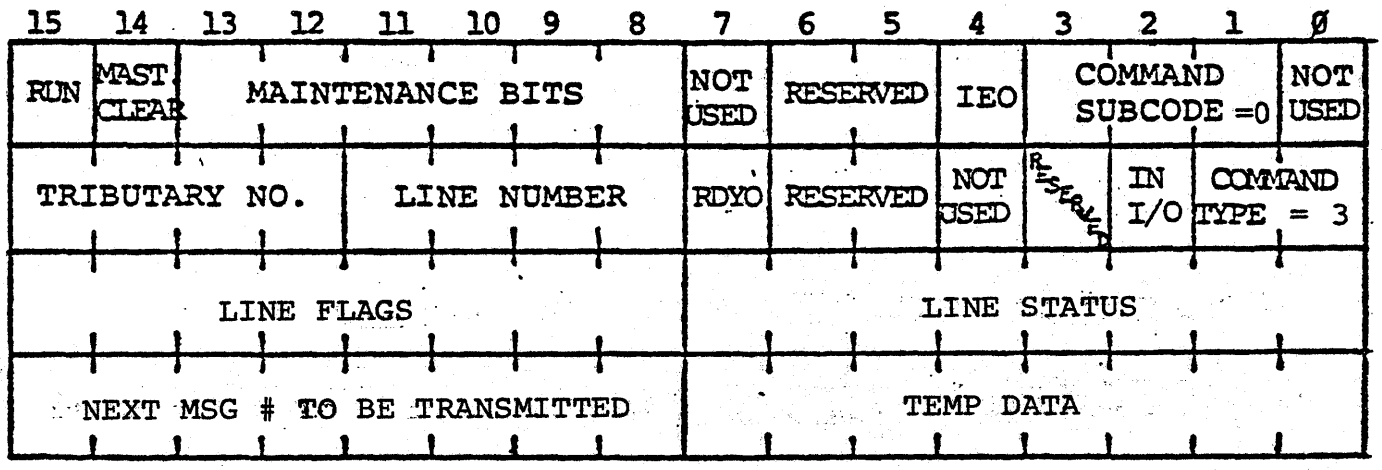
KMC CONTROL OUT COMMANDS

TO MANY RECEIVE BUFFERS

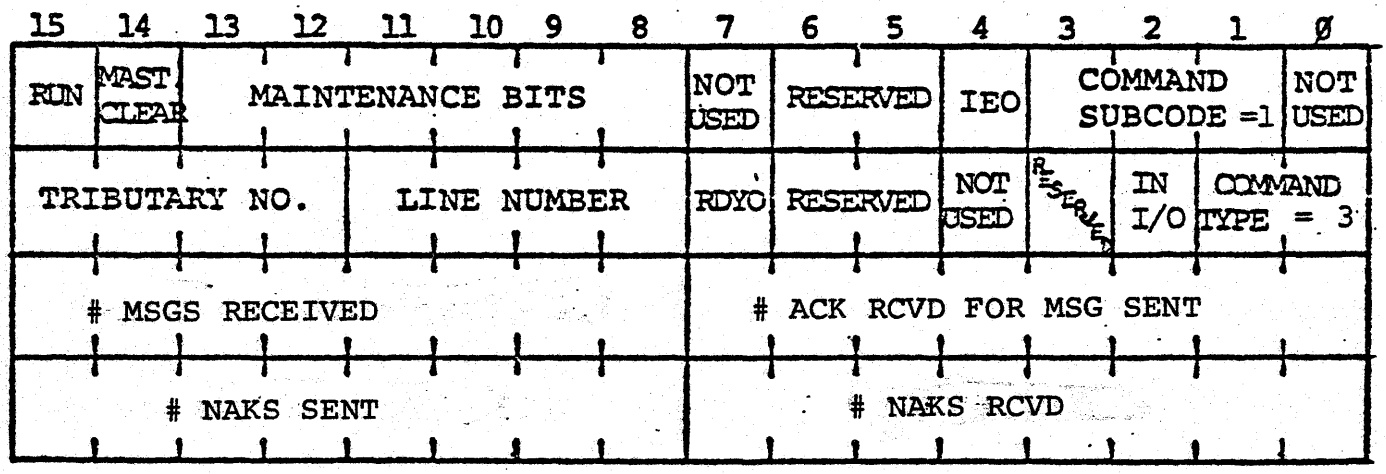


KMC CONTROL OUT COMMANDS

STATUS RESPONSES FORMAT 1

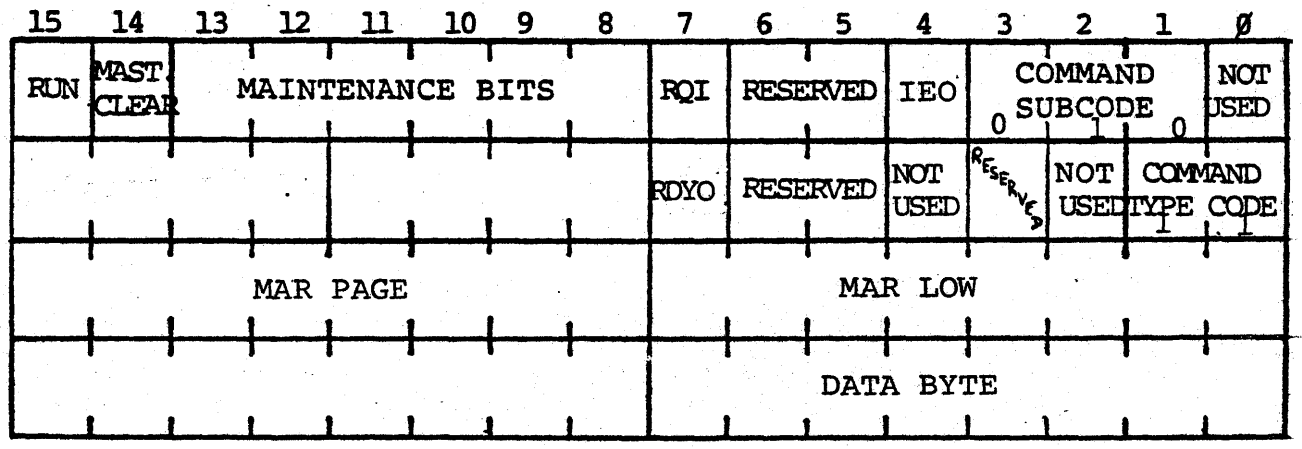


STATUS RESPONSES FORMAT 2



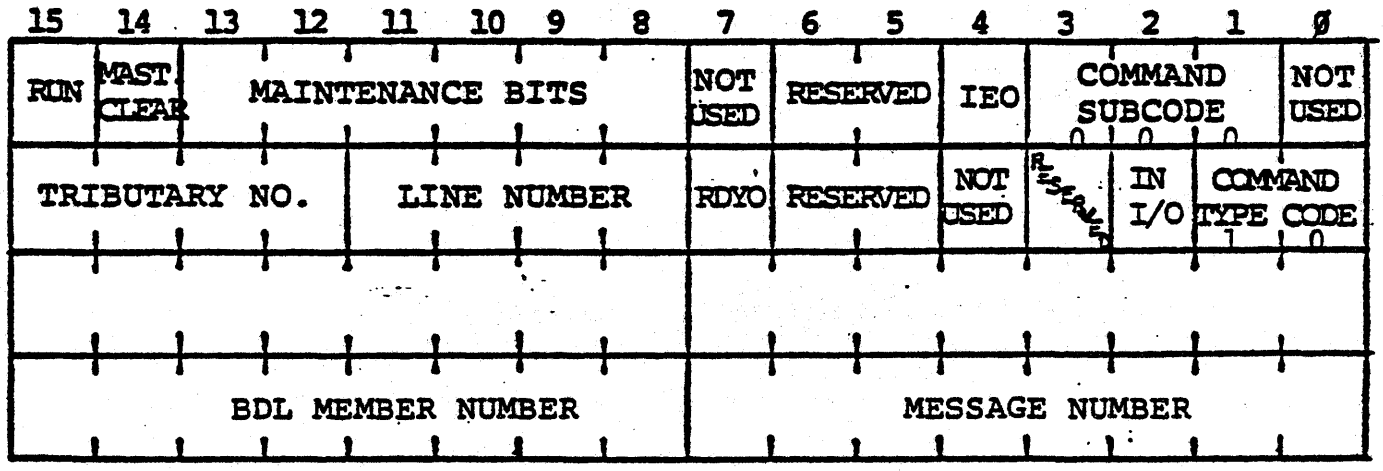
KMC CONTROL OUTPUT COMMANDS

READ KMC11 DATA MEMORY



KMC CONTROL OUT COMMANDS

MESSAGE RECEIVED



4.0 KMC DESIGN

4.1 The KMC DDCMP multiplexer is a state driven system, based on a series of queues, tables and buffers. Each state per line is less than or equal to 260 KMC instructions. The following sections describe the KMC memory organization, tables and queues used in the operation and control of the multiplexer.
The secion starts out with a detailed example of a message transmission and steps involved. This is followed by a memory map, and block diagram of the sytem and definition of tables. The last section contains a definition of various states the system may be in.

4.2 DETAILED EXAMPLE OF MESSAGE TO BE TRANSMITTED

1. Application program passes to driver BDL Base address.

2. PDP11 driver passes BDL base to KMC.

3. Application program request a transmission of a message providing PDP11 Driver with BDL member control address of message data and number of messages to be transmitted.

4. PDP11 Driver requests CSR transfer.

5. KMC acknowledges request when input que is available.

6. KMC retries CSR/Retransmit data and places data in input.

7. KMC command interpreter sees input data in queue. It determines from queue data line data is to be transmitted on. If line is busy, it leaves data in que and waits to check it next time around. It then dertermines if a control message is in the control out que. If so, it is prepared for transmission. Else it sets the line to output state 1 and places CSR data into output mesage character address buffer. The command interpreter state is then set back to 0.

8. The KMC then waits until select flag is set. If set, the KMC then builds the message header.

9. The KMC then builds a BDL address and places a read request into the output NPR queue and set BDL flag.

10. The KMC then performs THE NPR states which request the BDL Read.

11. After completion of the read, the message address contained in the BDL is placed in the Output message character address table.

12. An output request is placed in the NPR queue to set the BDL as in unuse by the KMC.

13. The KMC then requests that the first four words 8 character of the message be stored in the intermediate buffer.

14. After eight characters are buffered, the LST is updated, the header of the message is then transmitted.

15. After the header and CRC of the header is transmitted, the first character of the data message is transmitted.

16. After every even character of the message has been transmitted from the intermediate buffer, an NPR INPUT QUEUE request is made to get two more characters from the PDP11 for transmission until entire message has been retrieved.

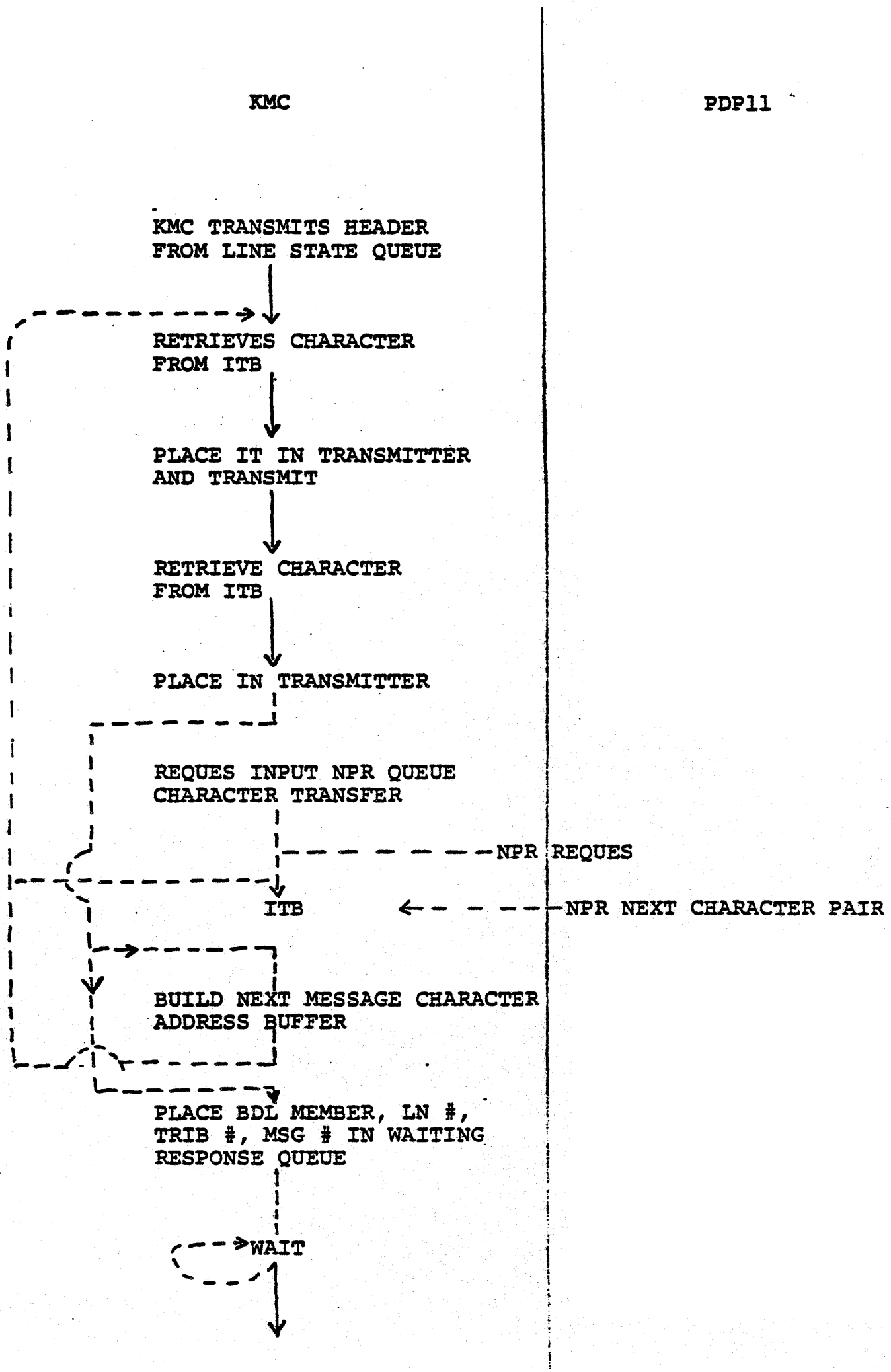
17. At this point in time, if multiple messages (pipelined messages) are to be transmitted, the next BDL address is calculated the BDL and next message flags are set and the address of the next message is retrieved. An NPR INPUT queue request is then made for a charcter pair of the next message while the current message is finishing its transmision.

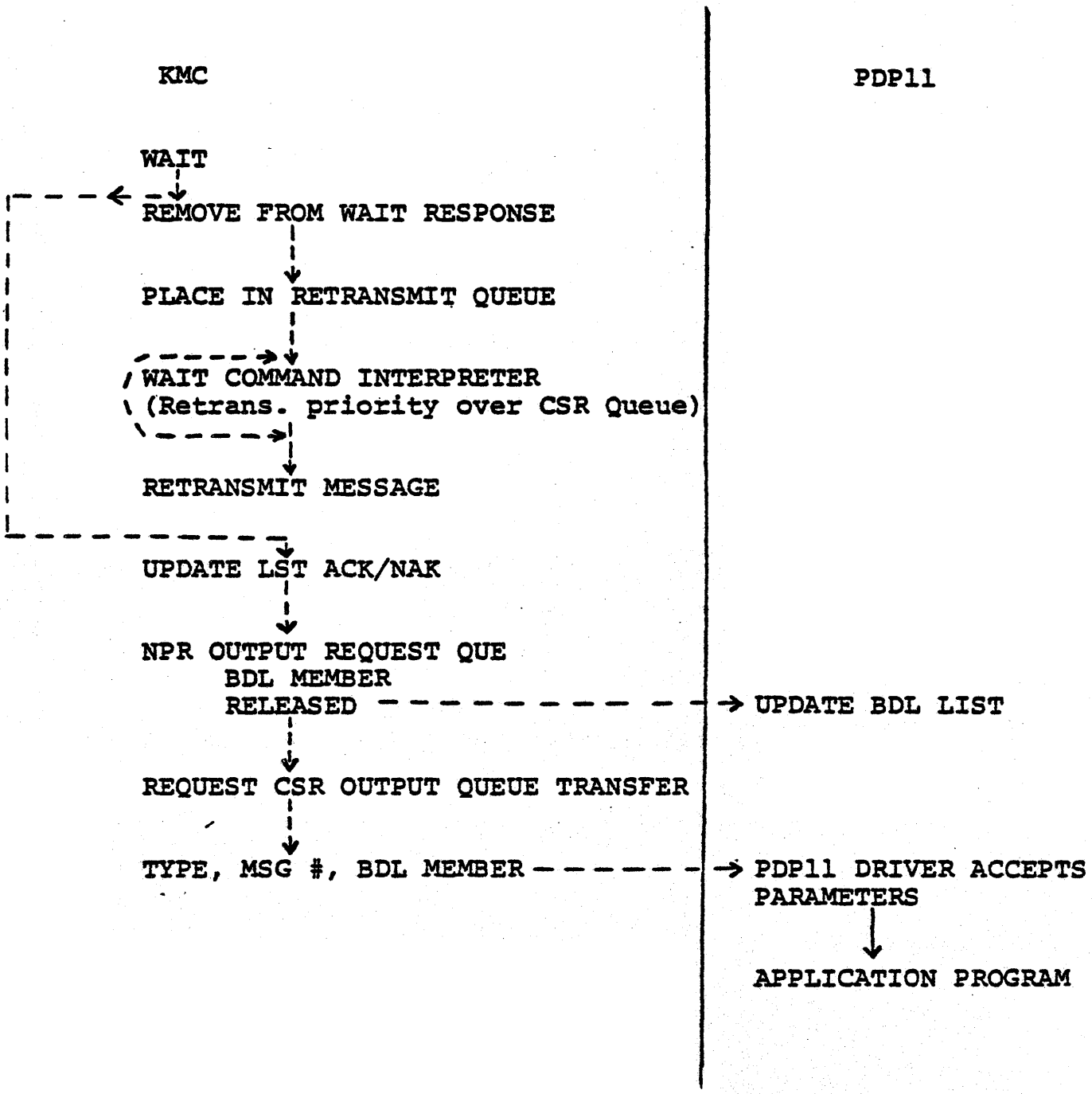
18. Upon completion of the transmission of the message, the message pending flag is set and the BDL member #, line # and tributary # are placed in the message waiting response que.

19. If the message is acked, then the entry in the message pending que is cleared. A request is placed in the OUT NPR Queue to release the BDL. A transmit complete CSR output request is placed in the CSR output que.

20. If the message is nacked, the entry in the message pending que is removed and placed in the retransmit que. An error counter is then incremented.

21. Before another CSR request is honored from the 11, the retransmit que is emptied and goes back for retransmission.

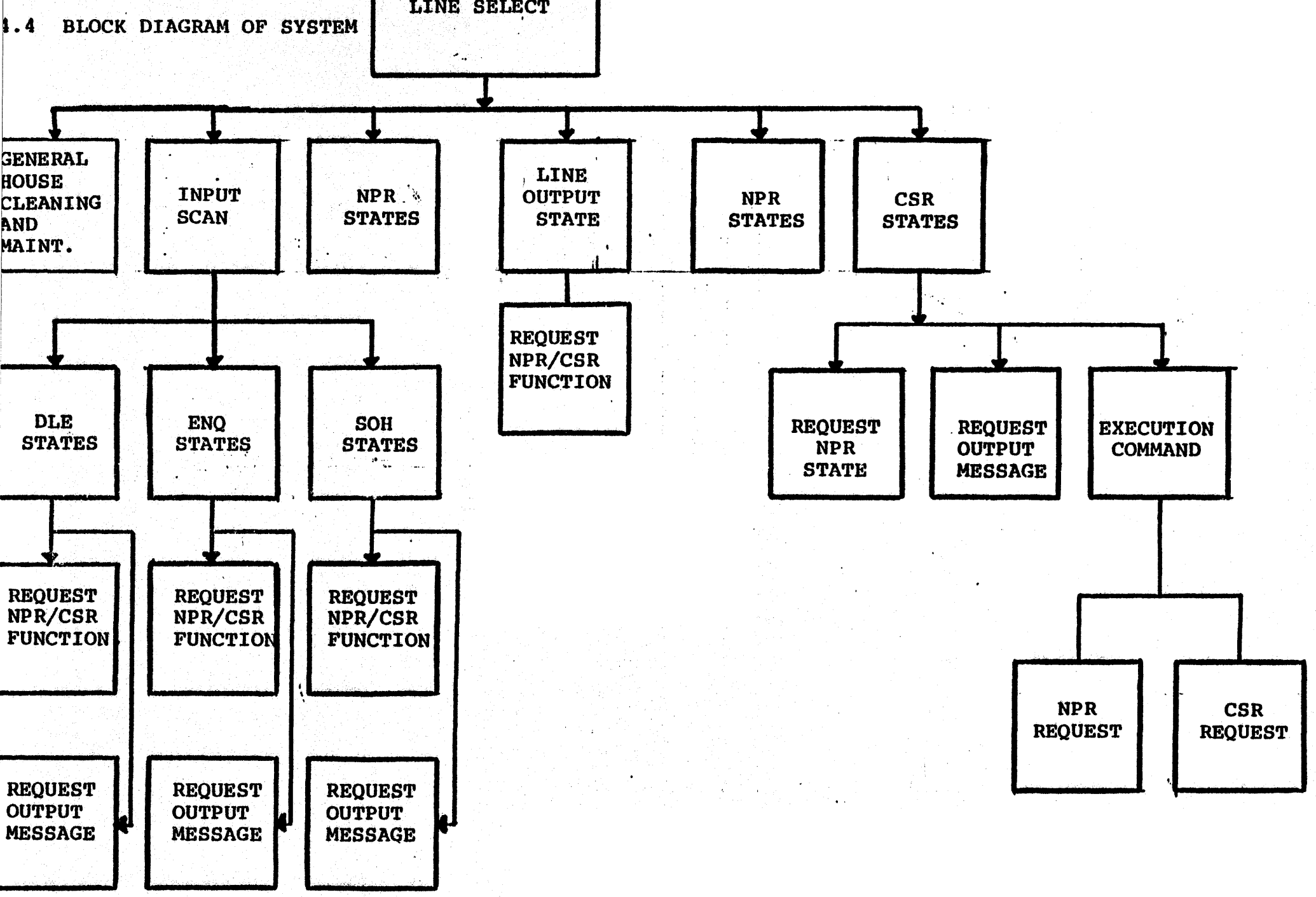

KMC
PDP11
KMC TRANSMITS HEADER FROM LINE STATE QUEUE
RETRIEVES CHARACTER FROM ITB
PLACE IT IN TRANSMITTER AND TRANSMIT
RETRIEVE CHARACTER FROM ITB
PLACE IN TRANSMITTER
REQUES INPUT NPR QUEUE CHARACTER TRANSFER
NPR REQUES
ITB
NPR NEXT CHARACTER PAIR
BUILD NEXT MESSAGE CHARACTER ADDRESS BUFFER
PLACE BDL MEMBER, LN #, TRIB #, MSG # IN WAITING RESPONSE QUEUE
WAIT


KMC
PDP11
WAIT
REMOVE FROM WAIT RESPONSE
PLACE IN RETRANSMIT QUEUE
WAIT COMMAND INTERPRETER
(Retrans. priority over CSR Queue)
RETRANSMIT MESSAGE
UPDATE LST ACK/NAK
NPR OUTPUT REQUEST QUE
BDL MEMBER
RELEASED
UPDATE BDL LIST
REQUEST CSR OUTPUT QUEUE TRANSFER
TYPE, MSG #, BDL MEMBER
PDP11 DRIVER ACCEPTS
PARAMETERS
APPLICATION PROGRAM

4.3 MEMORY DEFINITION

0	- 777	LINE STATUS TABLE (LST)
1000	- 1077	LINE OUTPUT STATE TABLE (LOST)
1100	- 1177	LINE INPUT STATE TABLE (LIST)
1200	- 1277	INTERMEDIATE TRANSMIT BUFFERS(ITB)
1300	- 1466	SPARE
1467	- 1477	CSR INPUT BUFFER
1500	- 1677	CONTROL MESSAGE OUTPUT QUE
1700	- 2077	CONTROL MESSAGE INPUT QUE
2100	- 2207	PRIMARY OUTPUT MSG ADDR BUFFER
2210	- 2317	SECONDARY OUTPUT MSG ADDR BUFFER
2320	- 2427	PRIMARY INPUT MSG ADDRESS BUFFER
2430	- 2547	SECONDARY INPUT MSG ADDRESS BUFFER
2550	- 2777	SPARE
3000	- 3377	NPR OUTPUT REQUEST QUE
3400	- 3777	MSG WAITING RESPONSE QUEUE 1ST HALF
4000	- 4377	NPR INPUT REQUEST QUEUE
4400	- 4777	CSR OUTPUT QUEUE
5000	- 5110	TIMERS
5111	- 6977	SPARE
6400	- 6777	MESSAGES WAITING RESPONSE QUEUE 2ND HALF
7000	- 7640	RETRANSMIT QUE
7640	- 7661	KMC DATA VARIABLES

4.4 BLOCK DIAGRAM OF SYSTEM



MEMORY LAYOUT

LINE STATUS TABLE

	MEM ADDR TRIB # LINE #			
LINE 0	0	0	0	1 2 3 4 5 6 7
TRIB 1	0	1	0	
TRIB 2	0	2	0	
TRIB 3	0	3	0	
TRIB 4	0	4	0	
TRIB 5	0	5	0	
TRIB 6	0	6	0	
TRIB 7	0	7	0	
LINE 1	1	0	0	1 2 3 4 5 6 7
TRIB 1	1	1	0	
TRIB 2	1	2	0	
TRIB 3	1	3	0	
TRIB 4	1	4	0	
TRIB 5	1	5	0	
TRIB 6	1	6	0	
TRIB 7	1	7	0	
	.			
	.			
	.			
	.			
LINE 7	7	0	0	1 2 3 4 5 6 7
TRIB 1	7	1	0	
TRIB 2	7	2	0	
TRIB 3	7	3	0	
TRIB 4	7	4	0	
TRIB 5	7	5	0	
TRIB 6	7	6	0	
TRIB 7	7	7	0	

LINE TRIBUTARY ENTRIES

0 = LINE/TRIB STATUS
1 = LINE/TRIB FLAGS
2 = FLAGS
3 = NEXT MESSAGE TO BE TRANSMITTED
4 = NUMBER OF ACKS RECEIVED
5 = NUMBER OF MESSAGES RECEIVED
6 = NUMBER OF NAKS TRANSMITTED
7 = NUMBER OF NAKS RECEIVED

LINE STATUS BYTE MEANINGS

BYTE	NAME	BIT #	DEFINITION	
0	LINE STATUS	BIT 0	0 = LINE DOWN	1 = LINE UP
		BIT 1	0 = DDCMP MODE	1 = MOP
		BIT 2	0 = XMIT NOT ACTIVE	1 = ACTIVE
		BIT 3	0 = RECEPTION NOT ACTIVE	1 = ACTIVE
		BIT 4	0 = NO START PENDING	1 = START PENDING
		BIT 5	0 = SPARE	1 = SPARE
		BIT 6	0 = FULL DUPLEX	1 = 1/2 DUPLEX
		BIT 7	SELECT FLAG FOR 1/2 DUPLEX TURN AROUND 0 = NO XMIT	
1	LINE FLAGS 1	BIT 0	0 = NO START RCVD	1 = START RCVD
		BIT 1	0 = NO NAK RCVD	1 = NACK RCVD
		BIT 2	0 = NO REP RCVD	1 = REP RCVD
		BIT 3	0 = NO XMIT CONTRO MSG	1 = XMIT CONTROL MSG
		BIT 4	0 = NO ACK RCVD	1 = ACK RCVD
		BIT 5	0 = NO STACK RCVD	1 = STACK RCVD
		BIT 6		
		BIT 7	0 = NO START PENDING	1 = START PENDING
2	TEMPORARY DATA		MSG # FROM ACK OR NAK	
3	NBRNXT		NEXT MESSAGE NUMBER TO BE TRANSMITTED	
4	NBRACK		NUMBER OF ACKs RECEIVED FOR MESSAGE SENT	
5	NBRRCV		NUMBER OF MESSAGES RECEIVED	
6	NAKRCV		NUMBER OF NAKs RECEIVED (NO OVERFLOW)	
7	NAKXMT		NUMBER OF NAKs TRANSMITTED (NO OVERFLOW)	

4.5 TABLES AND QUES AND BUFFERS

The following sections provide definitions of the Tables, Queues and Buffers used internally by the KMC11.

4.5.1 LINE STATUS TABLE (LST)

The line status table is used to reflect the current status and definition for each line. The table is configured of 64 entries at 8 bytes of information per entry. The entries consist of 8 lines with 8 tributaries per line. The line table entries can be thought of as a series of row entries. The line number and line tributary number make up the memory address of where the table entry for a given line is to be found. The line status table contains the following information:

* Current line status up/down
* Current mode DDCMP/MOP
* Transmitter state
* Receive state
* Full or 1/2 duplex line
* Select flag
* Next message number to be sent
* Number of ACKs received for message sent
* Number of messages sent
* Number of NAKs transmitted
* Number of NAKs received

4.5.2 LINE OUTPUT STATE TABLE (L.O.S.T.)

The Line Output State Table consists of eight subtables. One for each output line. Each subtable contains eight entries which contain the current state of the line and the header of the current output message except CRC values. The first byte of each subtable is the current state of the line. This state value is used as an indexed branch to execute the =next series of instructions to keep the line operating properly. The remainder of the table contains the header data of the current message being transmitted.

LINE OUTPUT STATE TABLE

	LINE TRIB							
		STATE	BASE BDL	BASE LOW	BASE UP	WORK BDL	BDL	# MSGS
ADDR	1000	1	2	3	4	5	6	7
	1010	STATE	HDR1	HDR2	HDR3	HDR4	CRC1/NEXT CHAR IN	CRC2/NEXT CHAR AVAIL.
	.							
	.							
	.							
	1077	"	"	"	"	"	"	"

LINE OUTPUT STATE TABLE CONTROL MSG

1	2	3	4	5	6	7
STATE	ACK #	LSTL	LSTH	INDEX	NAK COUNT	

4.5.3 LINE INPUT STATE TABLE (L.I.S.T.)

The Line Input State Table is configured exactly as the Line Output State Table except that it is used for input rather than output.

LINE INPUT STATE TABLE

STATES								
ADDR	1100	1	2	3	4	5	6	7
		STATE	HDR1	HDR2	HDR3	HDR4	CRC1/NEXT CHAR IN	CRC2/NEXT CHAR AVAIL.
	.							
	.							
	.							
	1170	"	"	"	"	"	"	"

4.5.4 CSR QUEUE

The CSR functions have been broken down into Input and Output States and Ques. There can be only one CSR state at any given time. If there is both an input request and an output request, the KMC output request takes priority. There are eleven (11) output queue entries and one input que entry. See Section 3.2.3.3 on control in and control out commands for specific CSR control information.

CSR OUTPUT QUE ENTRIES

BYTE 0 = SUBCOMMAND
1 = LINE & TRIB #
2 = DATA BSEL 3
3 = DATA BSEL 4
4 = DATA BSEL 5
5 = DATA BSEL 6
6 = COMMAND TYPE

CSR INPUT QUEUE

BYTE 0 = COMMAND (BITS 0,1)
1 = SUBCOMMAND (BITS 2,3,4)
2 = LINE # (BITS 0-3) & TRIB # (BITS 4-7)
3 = BDL ADDRESS MIDDLE BYTE/DATA
4 = BDL ADDRESS LOWER BYTE/DATA
5 = BDL ADDRESS UPPER 2 BITS (6,7)/MEMBER NO. BITS 0-7
6 = # OF MESSAGES TO BE SENT.

4.5.5 NPR INPUT REQUEST

The NPR input request queue is the only interface that allows NPR input transfers between the KMC and the PDP11. There are two basic types of input NPR transfers in the system: message data and B.D.L. data. To request a BDL transfer, the line number is entered into the NPR Que with BDL bit set (bit 4) and with either the primary or secondary message character buffer bit set (bit 5). Bits 0-3 indicate the line number that is requesting the transfer and point to the message character buffer used. Bit 4 if on requests that the contents be placed not in the intermediate buffer for the line, but in the message character address buffer. Bit 5 indicates whether the BDL data is to be in the primary or secondary message character address buffer. For a simple message character input only, the line number is entered and the retrieved data characters are placed in the intermediate buffer for the requesting line.

4.5.6 NPR OUTPUT REQUEST QUEUE

The NPR output request que operates similar to that of the input request. The queue is structured rather differently; there are three bytes per entry verses the one (1) in the Input Request Que. The first byte the line number has the same definition as that of an Input Request. The second two bytes are the character pair or data that is to be stored.

NPR CHARACTER INPUT REQUEST QUEUE

LINE #	LINE #	LN#	LN#	LINE #
ENTRY 1	ENTRY 2	ENT.3	ENT. 15	ENTRY 16

NPR CHARACTER OUTPUT REQUEST QUEUE

	CHARACTER A	CHARACTER B	ADDR
ENTRY 1	"	"	"
ENTRY 2	"	"	"
ENTRY 3			
ENTRY 16	CHARACTER A	CHARACTER B	ADDR

4.5.7 INPUT/OUTPUT MESSAGE CHARACTER ADDRESS BUFFERS

The Output Message Character Address Buffers are the main interface to the NPR queue. These buffers contain the address of where the next two characters for transfer from the PDP11 are stored. It also contains the character count of the message and the count of the number of characters retrieved from the PDP11. The tributary number and current message transmit number are also stored in this buffer. An example of its use is after a character pair is transmitted, the line number is placed in the input NPR queue. The input NPR queue accesses the output message character address buffer to find the address of the next character pair to be transmitted.

There are four message character address buffers - two (2) for input a`nd two (2) for output. Th`=e reason for redundancy is to save time during pipelined input or output messages. The "next" Input/Output Message Character Address buffer is readied during use of the primary buffer. As soon as the primary buffer is complete, a buffer flip flop takes place and the secondary is now the primary buffer interface for transmitting or receiving characters.

OUTPUT DATA CHARACTER ADDRESS BUFFER

BYTE	0	1	2	3	4	5	6	7	8
	TRIB 3	BDL ADDR		CHAR COUNT		# OF CHAR TRANSFERRED		BDL #	# OF MSGS
LINE 1									
LINE 2									
LINE 3									
.									
LINE 8									

INPUT DATA CHARACTER ADDRESS BUFFER

BYTE	0	1	2	3	4	5	6	7	8
	BDL #	BDL ADDR			CHAR COUNT		# OF CHAR RECEIVEDS		ODD BYTE RECEIVED
LINE 1									
LINE 2									
LINE 3									
.									
LINE 8									

WAITING RESPONSE BUFFER

	LINE #	TRIB #	MSG #	BDL MEMBER
ENTRY 1				
ENTRY 2				
ENTRY 3				
.				
.				
.				
ENTRY 100	LINE #	TRIB #	MSG #	BDL MEMBER

4.5.8 MESSAGE WAITING RESPONSE QUEUE

After a message has been transmitted, its line number, tributary number, message number and BDL number are entered into the waitin response queue. This queue is a holding area containing all pertinent information concerning the message until the message disposition can be determined. If the message is acked, the BDL member is released, the queue entry cleared, and the PDP11 is notified of the transmit. If the message was naked and seven retries have not occurred, the queue entry will be cleared and th data will be entered into the retransmit que.

4.5.9 RETRANSMIT QUE

This queue contains information required to retransmit a message that was NAKED or failed to get an acknowledgement. This queue i examined and takes priority over the CSR input request que from the PDP11.

4.5.10 INTERMEDIATE TRANSMIT BUFFER

There is one intermediate buffer per line in the system. This buffer area provides an input buffering function between characters being sent from the PDP11. The ITB keeps eight (8) characters buffered ahead of the transmitter such that if a transfer of a character is late in coming from the PDP11, the KMC will not have to wait and will transmit the seven (7) previously buffered characters before the one that arrived late. The ITB is used in conjunction with the NPR INPUT request que.

INTERMEDIATE TRANSMIT BUFFER

ADDR	120	0	1	2	3	4	5	6	7
	121	DATA	DATA	DATA	DATA	DATA	DATA	DATA	DATA
	122		"	"	"	"	"	"	"
	.								
	.								
	.								
	.								
	127		"	"	"	"	"	"	"

INTERMEDIATE RECEIVE BUFFER

ADDR	130	0	1	2	3	4	5	6	7
	131	DATA	DATA	DATA	DATA	DATA	DATA	DATA	DATA
	132								
	.								
	.								
	.								
	.								
	137								

4.5.11 CONTROL MESSAGE QUEUE

The control message queues are used to hold control messages that are to be transmitted as soon as the line is free. There are two (2) queues - one for input message reception and the second is for message transmission. Position in the queue defines line and tributary, thus entry nine in the que is for line one, tributary one. To determine the line and tributary number, the queue entry is divided by eight. The integer from the division gives the line number and the remainder is the tributary number. Each queue entry is two bytes long. The first byte defines the message type and the second the subtype.

LINE 0 Type	LINE 0 Q,S Bits SubType	LINE 1 Type	LINE 1 SubType	LINE 2 Type	LINE 3 SubType	LINE 4 Type
LINE 4 SubType	LINE 5 Type	LINE 5 SubType	LINE 6 Type	LINE 6 SubType	LINE 7 Type	LINE 7 SubType

4.6 SYSTEM STATES

The following sections contain a list of the various states the multiplexer may be in. It should be noted that it is possible and very likely that the KMC will be in several different states at any given point in time.