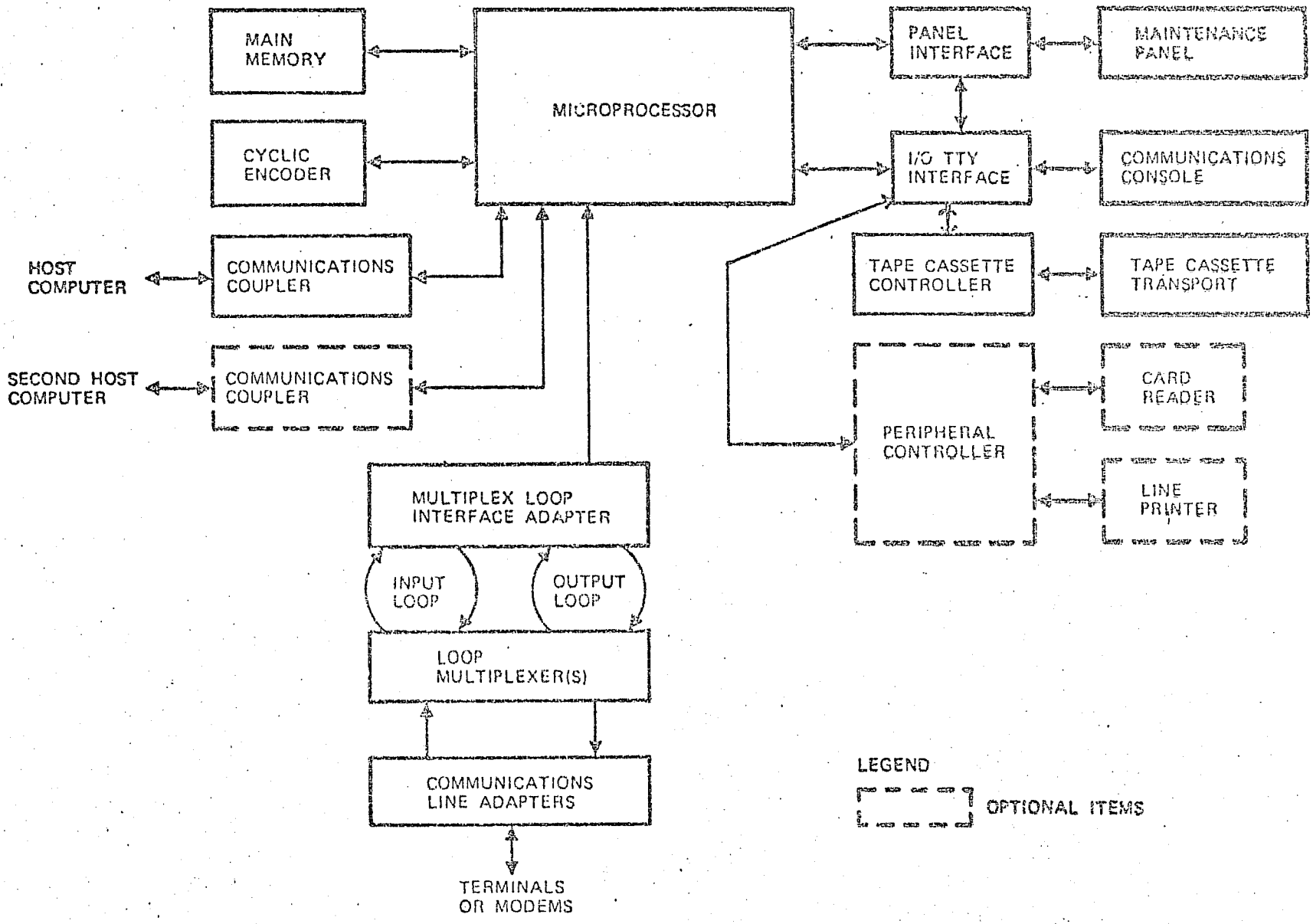
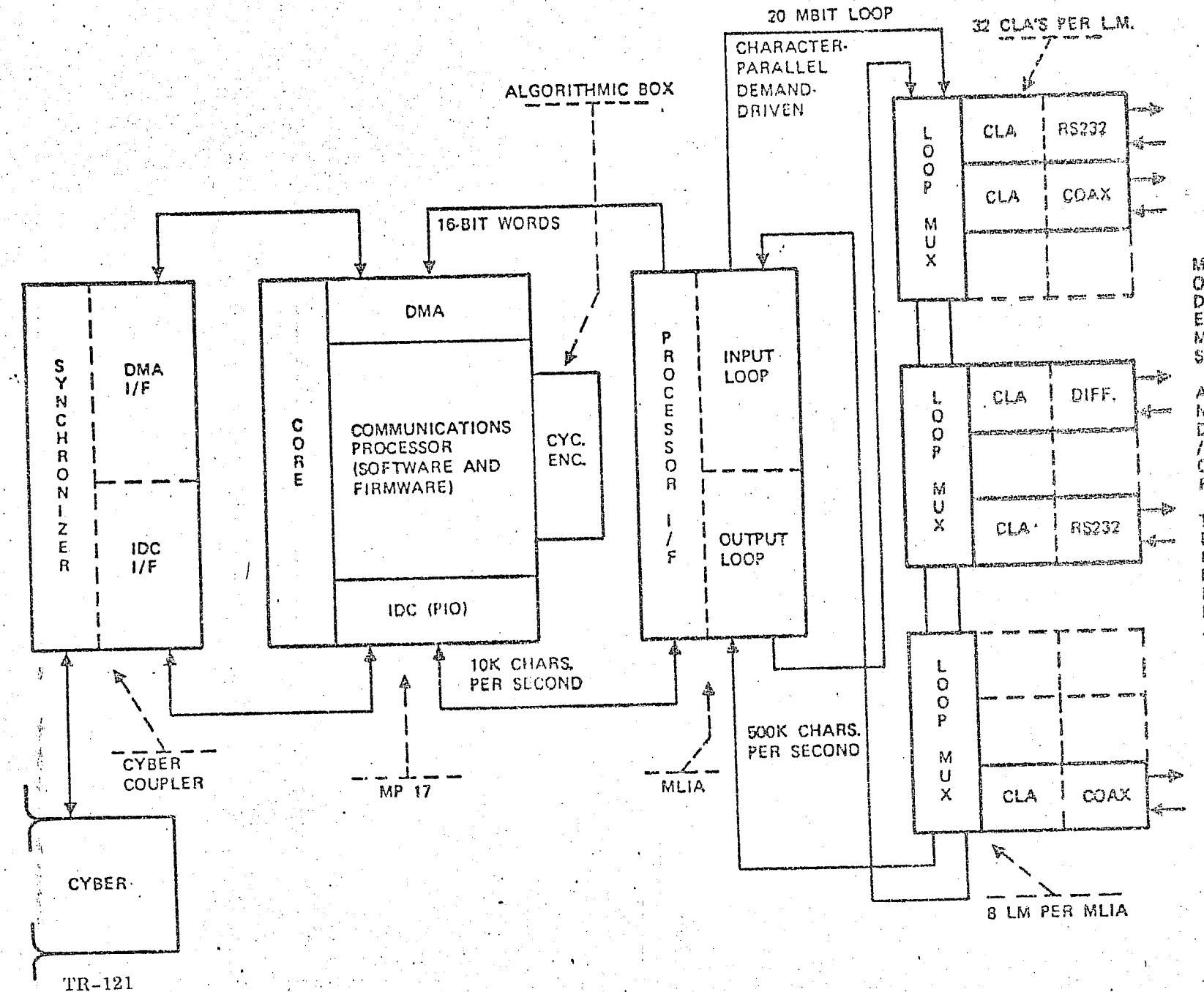
2550 HOST COMMUNICATION PROCESSOR

[redacted]

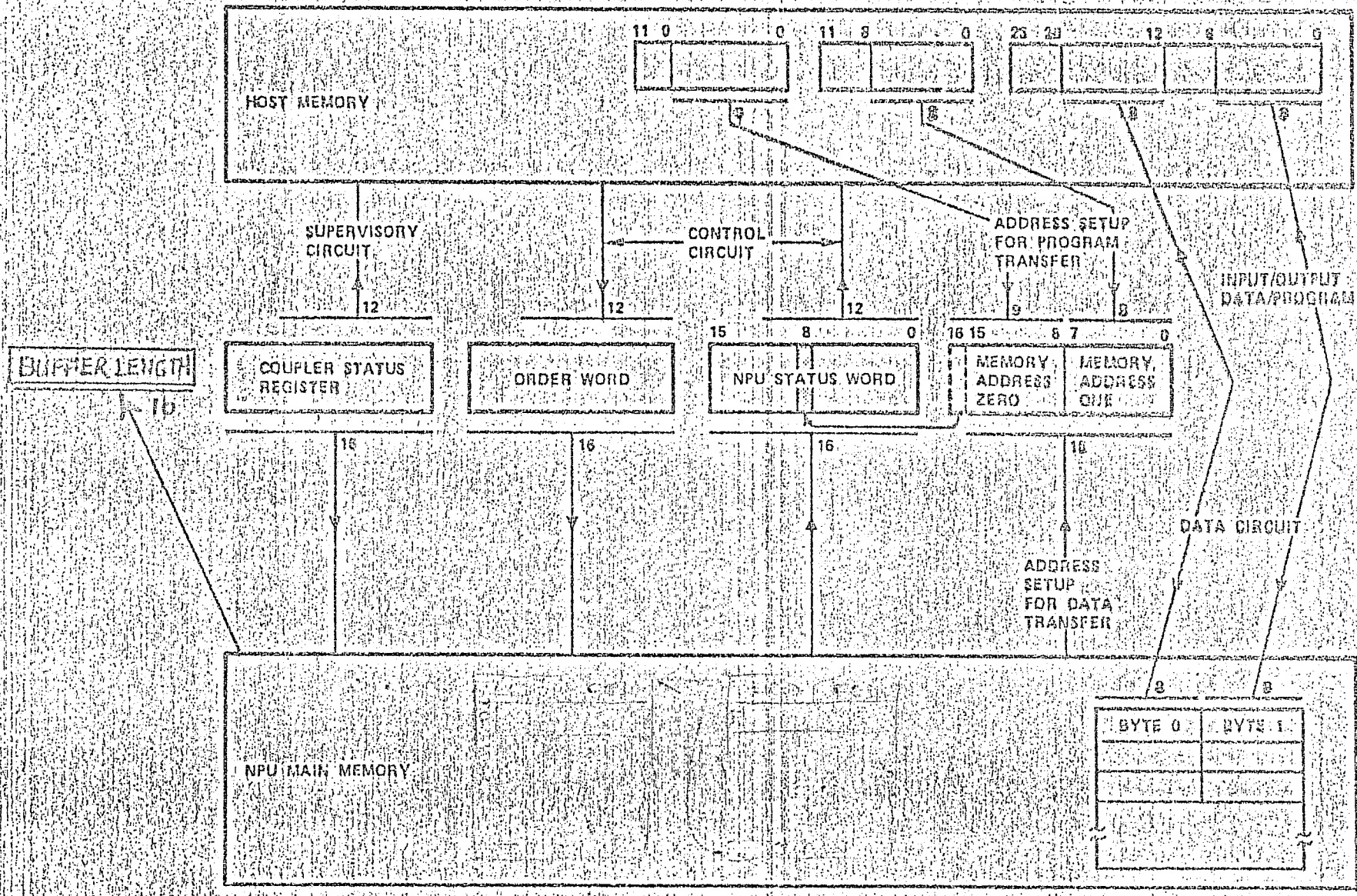


T8010

Main Data Paths – 2550-2 HCP


20 MBIT LOOP
32 CLA'S PER L.M.
CHARACTER-
PARALLEL
DEMAND-
DRIVEN
ALGORITHMIC BOX
16-BIT WORDS
LOOP MUX
CLA
RS232
CLA
COAX
SYNCHRONIZER
DMA I/F
IDC I/F
CORE
DMA
COMMUNICATIONS PROCESSOR (SOFTWARE AND FIRMWARE)
CYC. ENC.
IDC (PIO)
PROCESSOR I/F
INPUT LOOP
OUTPUT LOOP
LOOP MUX
CLA
DIFF.
CLA
RS232
LOOP MUX
CLA
COAX
MODEMS AND/OR TERMINALS
10K CHARS. PER SECOND
500K CHARS. PER SECOND
CYBER COUPLER
MP 17
MLIA
CYBER
8 LM PER MLIA
TR-121
T8010

CYBER Coupler Registers



TR-129

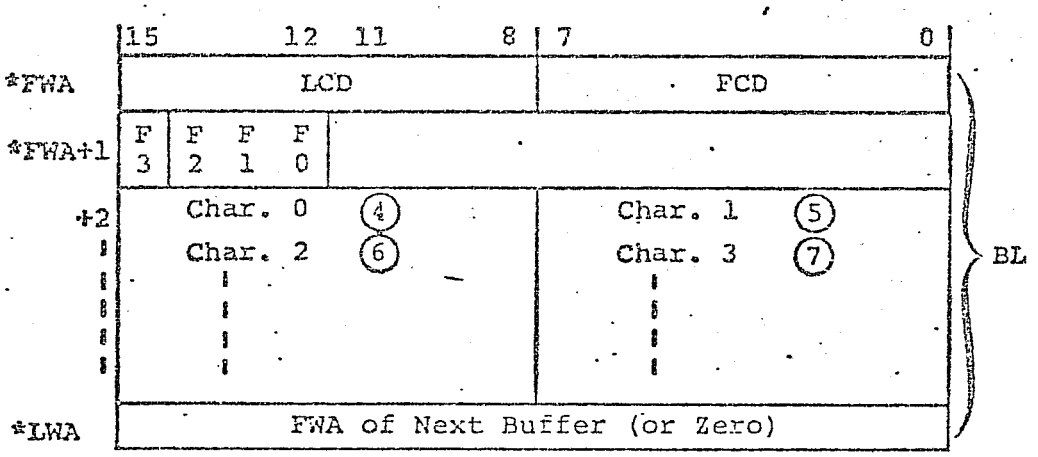
T8010

I/O PROGRAMMING FORMATS

COMMUNICATIONS COUPLER PPU FUNCTION CODES

11 10 9	8 7 6	5	4	3	2 1 0	
Equip Code (=7)	1 0 0	0	0	X	X X X	Clear Coupler
	0 1 0	0		X	X X X	Clear CP
	0 0 1	0		X	X X X	Stop CP
	0 0 0	1		X	X X X	Start CP
	X X X	X		0	0 0 0	Input Memory Address Zero*
	X X X	X		0	0 0 1	Input Memory Address One*
	X X X	X		0	0 1 0	
	X X X	X		0	0 1 1	Input Data
	X X X	X		0	1 0 0	Input CP Status
	X X X	X		0	1 0 1	Input Coupler Status
	X X X	X		0	1 1 0	Input Order Word*
	X X X	X		0	1 1 1	Input Program
	X X X	X		1	0 0 0	Output Memory Address Zero (upper byte)
	X X X	X		1	0 0 1	Output Memory Address One (lower byte)
	X X X	X		1	0 1 0	
	X X X	X		1	0 1 1	
	X X X	X		1	1 0 0	Output Data
	X X X	X		1	1 0 1	Output Program
	X X X	X		1	1 1 0	Output Order Word
	X X X	X		1	1 1 1	

*Hardware Maintenance feature



FWA = First Word Address of Buffer (must be multiple of BL)

LWA = Last Word Address of Buffer (LWA = FWA + (BL-1))

LCD = Last Character Displacement (relative to FWA)

FCD = First Character Displacement (relative to FWA)

F3 = Last Buffer Flag

F2-F0 = Last Character Flag (place succeeding character in next buffer(s))

BL = Buffer Length (words) $BL=2^N$, $2 \leq N$ (integer) ≤ 7

CD = Character Displacement ($4 \leq CD \leq 253$)

*Control words placed in core before data I/O

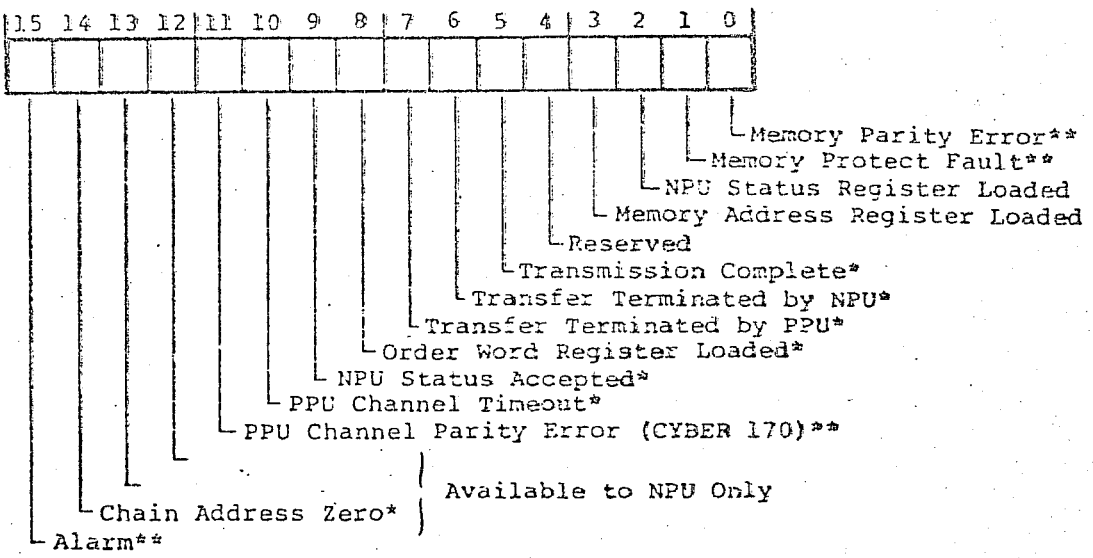
NOTE: LCD$\geq$FCD. If LCD<FCD, buffer is skipped and operation points to the next buffer chain.

15 14 13 12	11	10 9 8	7	6 5 4	3 2	1 0
0 0 0 0	0	1 E E E				
		Equipment Address				Not Used

First Coupler = 1100 ('C') (Standard) Second Coupler = 1101 ('D') (Optional)	Sample (Read)	0 0 0	0 0	Input Memory Address Zero*
		0 0 1	0 0	Input Memory Address
		0 1 0	0 0	
		0 1 1	0 0	Input First/Present Character Displacement
		1 0 0	0 0	Input CP Status*
		1 0 1	0 0	Input Coupler Status
		1 1 0	0 0	Input Order Word
		1 1 1	0 0	Input IO*
		0 0 0	0 1	Input Last Word From Data Channel*
		0 0 1	0 1	Input FDMAR0/FDMAR1*
		0 1 0	0 1	Input FDMAR0/Flag Mux*
		0 1 1	0 1	Input FDMAR1/Flag Mux/Flag Register*
		1 0 0	0 1	
		1 0 1	0 1	Input Switch Status
		1 1 0	0 1	
		1 1 1	0 1	Input Character*
	Set (Write)	0 0 0	1 0	Output Memory Address Zero*
		0 0 1	1 0	
		0 1 0	1 0	
		0 1 1	1 0	Output FCD, PCD, LCD*
		1 0 0	1 0	Output CP Status
		1 0 1	1 0	Output Buffer Length
		1 1 0	1 0	Output Order Word*
		1 1 1	1 0	
		0 0 0	1 1	Clear Coupler
		0 0 1	1 1	Terminate Transfer
		0 1 0	1 1	
		0 1 1	1 1	
		1 0 0	1 1	Output Test*
		1 0 1	1 1	Input Test*
		1 1 0	1 1	Output Memory Address
		1 1 1	1 1	Output Character*

*Hardware maintenance feature

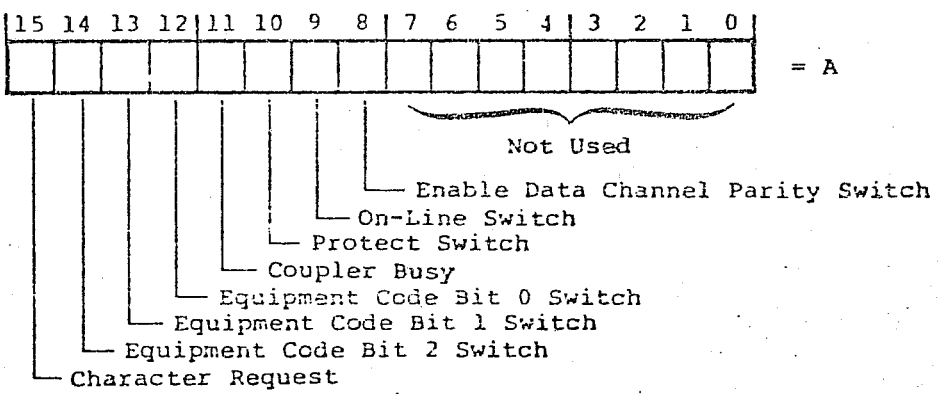
COMMUNICATIONS COUPLER STATUS FORMAT



**Alarm Condition (all alarms generate NPU Interrupt)
*NPU Interrupt Condition

NOTE: All non-alarm interrupt conditions except OWRL are cleared by input coupler status command, clear coupler function, clear coupler command. OWRL interrupt condition is cleared by input OW command, clear coupler function and command, and Master Clear. Alarm interrupt conditions are cleared only by clear coupler functions and commands and Master Clear.

COMMUNICATIONS COUPLER INPUT SWITCH STATUS FORMAT



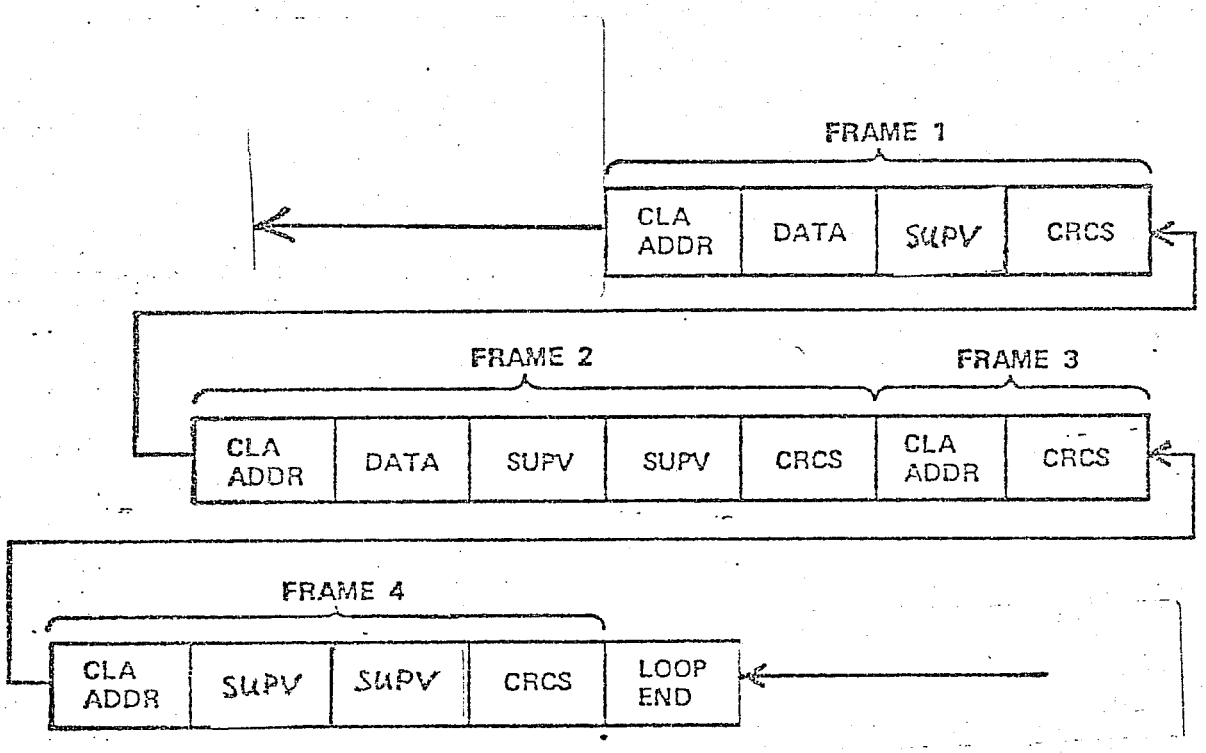
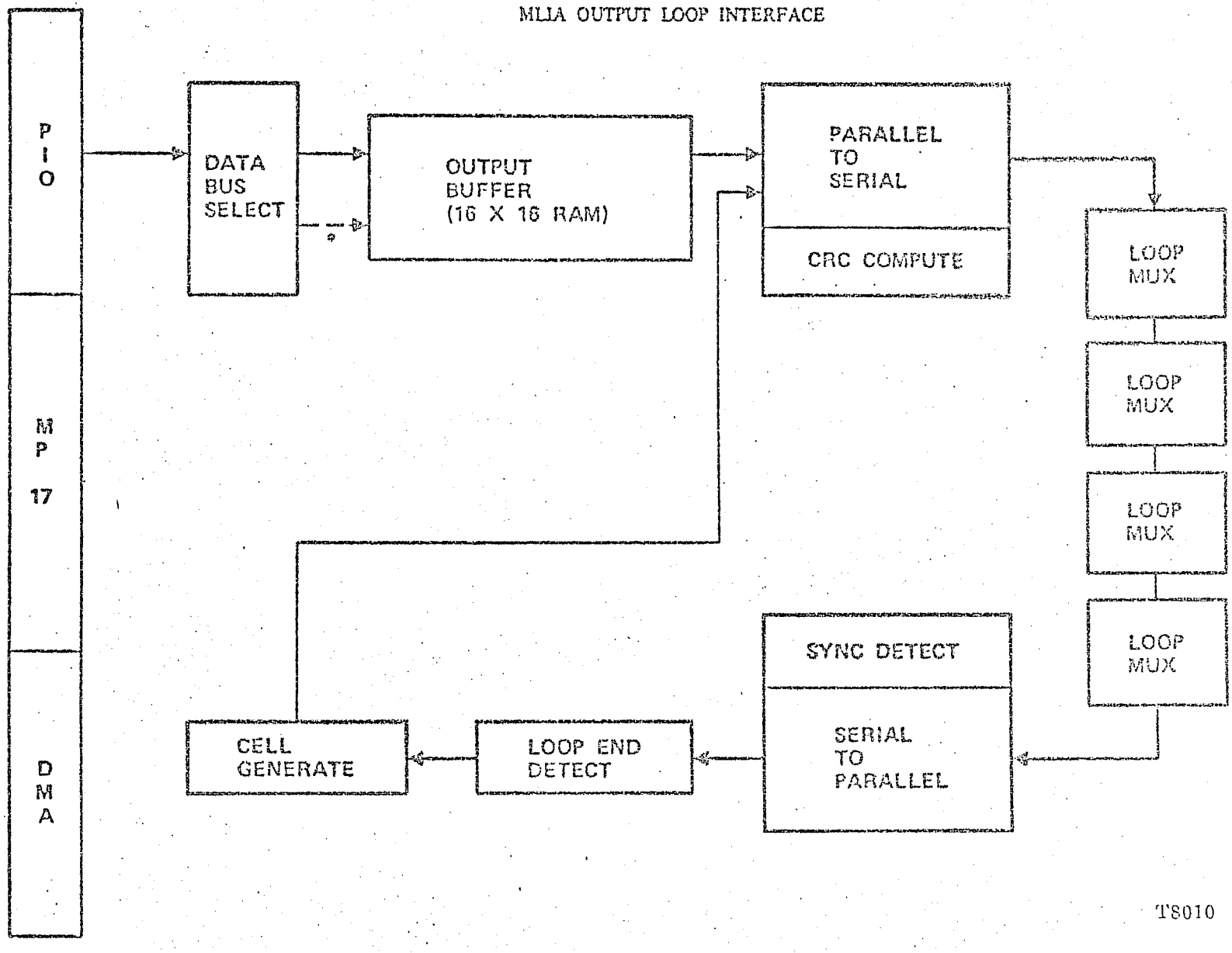


Figure Loop Batch and Line Frame Structure

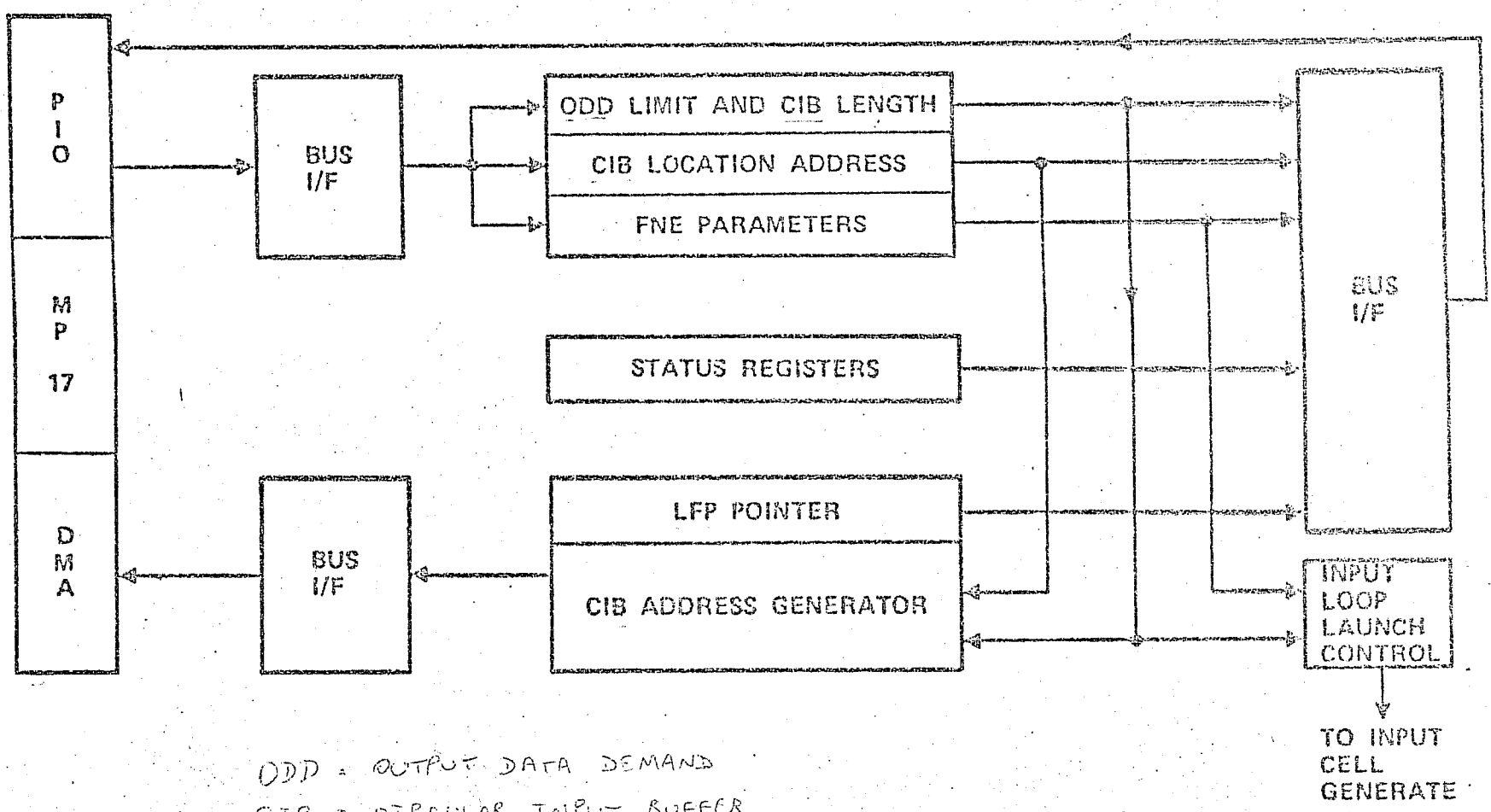
MLIA OUTPUT LOOP INTERFACE



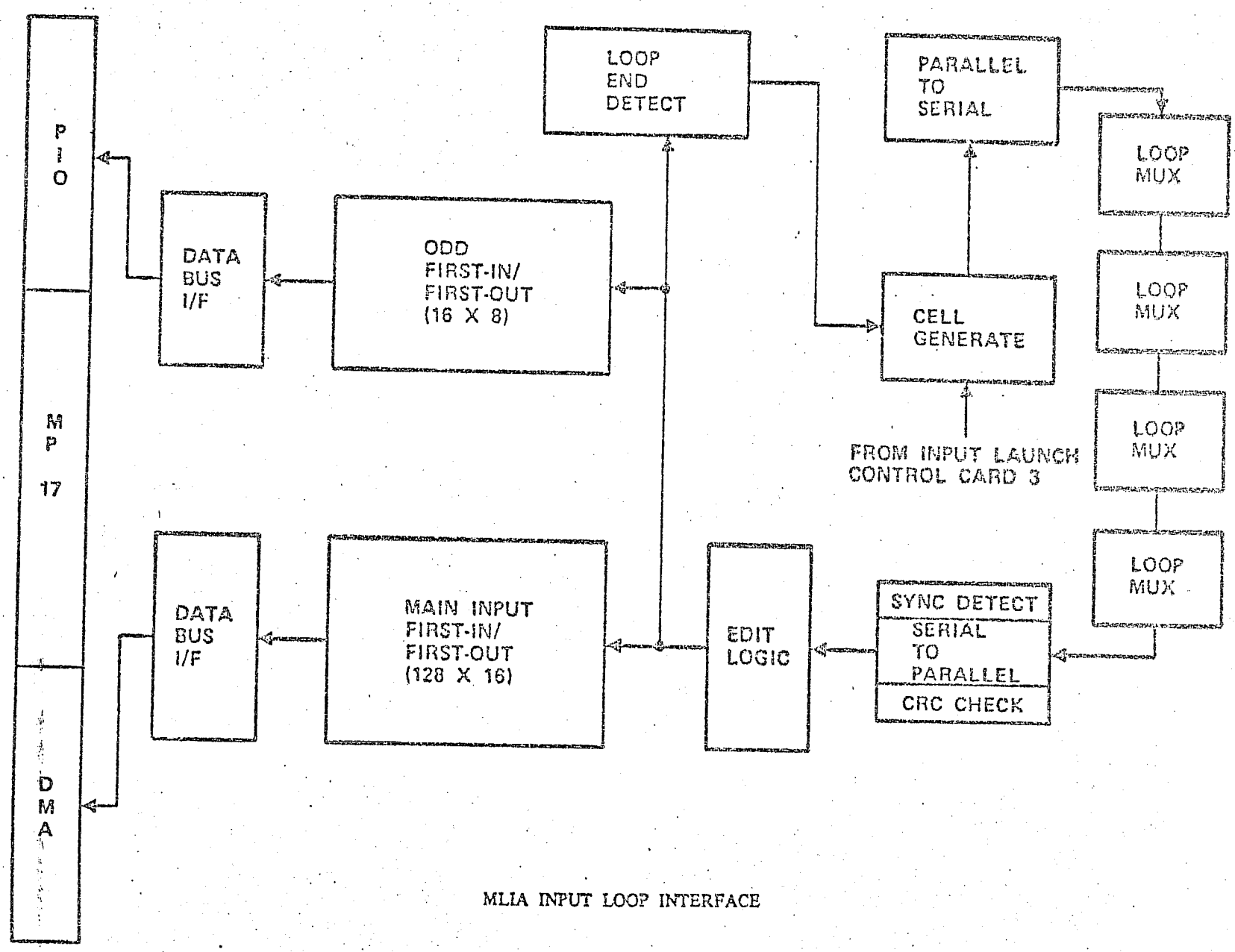
TR-124

TS010

MLIA PROCESSOR INTERFACE



ODD = OUTPUT DATA DEMAND
CIB = CIRCULAR INPUT BUFFER
FNE = NO. OF EMPTYS & NULLS IN LOOP BATCH
LFP = LAST FRAME POSITION



MLIA INPUT LOOP INTERFACE

MLIA

COMMAND CODES

**Q-FIELD	OPERATION
0519	RESET
0529	IDLE
050A	WRITE LOC
051A	WRITE LIM/LGH
0578	WRITE FNE (RESET LFP)
0539	NORMAL
0576	READ STATUS
0500	READ ODD
0560	READ LFP
0502	READ LOC
0512	READ LIM/LGH
0570	READ FNE
0508	WRITE OUTPUT CELL
0549	CLEAR OUTPUT BUFFERS
057D	SET CASCADE LOOPBACK
056D	SET OUTPUT LOOPBACK
055D	SET INPUT LOOPBACK
054D	SET TRANSPARENT MODE
053D	CLEAR CASCADE LOOPBACK
052D	CLEAR OUTPUT LOOPBACK
051D	CLEAR INPUT LOOPBACK
050D	CLEAR TRANSPARENT MODE

**HEXADECIMAL NOTATION

LOC:

LIM:

LGH:

A-REGISTER FORMATS

CIB LOCATION

16-BIT ABSOLUTE CORE ADDRESS

ODD LIMIT | CIB LENGTH

F1 | F2 | F3 | N | E

SECONDARY LINE TOGGLE

RESERVED

RESET LFP

15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0

STATUS BIT SET

0 = NOT RESET AND LOOPS ACTIVE
2 = ERROR INTERRUPT*
3 = NORMAL AND NO DIAGNOSTIC
5 = INPUT BUFFER FULL*
6 = OUTPUT LOOP ERROR*
9 = INPUT LOOP ERROR*
*CAUSES INTERRUPT

0 0 0 0 0 0 0 0 | CLA ADDRESS

0 0 0 0 | LFP POINTER

F1 | F2 | F3 | LOOP CELL

BUFFER DESTINATION FLAG

END OF BATCH

END OF FRAME

Figure 5-5. MLIA Commands

CELL MARKER
CELL IDENTIFICATION
CELL INFORMATION

TYPE	FORMAT NUMBER	M	ID			INFORMATION								CELL DESIGNATION
		B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	
LOOP MANAGEMENT FORMAT	0	1	0	0	0	0	0	0	0	0	0	0	0	EMPTY
	0	1	0	0	0	1	0	1	0	1	0	1	0	NULL
	1	1	0	0	1	F3	F2	1	1	1	0	0	0	LOOP END
	1	1	0	0	1	F3	F2	0	0	0	1	1	1	LOOP END RESTART
RESERVED FORMATS	2	1	0	1	0	R	R	R	R	R	R	R	R	UNDEFINED
	3	1	0	1	1	R	R	R	R	R	R	R	R	UNDEFINED
LOOP USAGE FORMATS	6/7 E+F	1	1	1	F1	A1	A2	A3	A4	A5	A6	A7	A8	CLA ADDRESS
	4 C	1	1	0	0	D1	D2	D3	D4	D5	D6	D7	D8	DATA
	5 D	1	1	0	1	S1	S2	S3	S4	S5	S6	S7	S8	SUPERVISION
	6	1	1	1	0	C1	C2	C3	C4	C5	C6	C7	C8	CRCS

INPUT LOOP CODING

F1 = 0 NO OUTPUT DATA DEMAND
F1 = 1 OUTPUT DATA DEMAND
F2 = X UNDEFINED
F3 = 0 PRIMARY DECODES LOOP END; SECONDARY IGNORES LOOP END
F3 = 1 PRIMARY IGNORES LOOP END; SECONDARY DECODES LOOP END

OUTPUT LOOP CODING

F1 = 1
F2 = X UNDEFINED
F3 = X UNDEFINED

CODING ON BOTH LOOPS

A1 TO A8 = BINARY CODE OF CLA ADDRESS; A1 IS MOST SIGNIFICANT BIT
D = DATA
C = CYCLIC REDUNDANCY CHECK CHARACTER (CRCS), C1 IS MOST SIGNIFICANT BIT
R = X RESERVED AND UNDEFINED

Figure 5-3. Format of Loop Cells

(CRC) character. On the input loop, the address cell also carries the output-data-demand (ODD).

Other cells within the frame may contain data and/or supervision (status or commands). All cells are passed unmodified between the multiplex loop and the CLA, except the check character which is removed from the output loop and added to the input loop by the loop multiplexer.

LINE FRAME FORMATS

The ACLA uses the following general line frame format on the input loop:

E C D D

ACLA ADDR	INPUT DATA	STATUS WORD 1	STATUS WORD 2	CRC CHAR

The ACLA address cell is always present. It may contain an output data demand. The data cell appears next (if present) and contains input data. Two supervision cells follow (if present) and contain status word 1 and status word 2. If any status is to be reported, both status words always appear. The CRC character is added by the loop multiplexer.

The following general line frame format is required by the ACLA on the output loop:

E C D D D D

ACLA ADDRESS	OUTPUT DATA	COMMAND WORD 1	COMMAND WORD 2	COMMAND WORD 3	COMMAND WORD 4	CRC CHARACTER

The ACLA address cell is always present. Either the data cell or command words may appear next. The data cell (if present) contains output data. One to four supervision cells (if present) contain command words 1, 2, 3 & 4. Allowable combinations of command words are: none; 1; 1 and 2; 1, 2 and 3; or 1, 2, 3 and 4. The CRC character is removed by the loop multiplexer.

CELL AND WORD FORMATS

ADDRESSING CELL

Each ACLA is set to a different 8-bit binary address. Thus, when data, input-supervision, or output-data-demand are presented to the loop multiplexer from the ACLA, the first operation of the ACLA is to present its particular address.

Data or output supervision presented to the ACLA from the LM is preceded by an address. The ACLA compares this address with the internally preset address. If the two agree, the ACLA accepts the data or supervision. The addressing code format is shown in table 2-2.

In table 2-2, bit position IF3 is the output-data-demand bit in the address code. When the address is presented to the LM from the ACLA, the bit is a

TABLE 2-2. ADDRESSING CELL FORMAT

11	10	9	8	7	6	5	4	3	2	1	0	Processor Word Bit Position
B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	Loop Cell Bit Position
X	OF1	OF2	OF3	IO1	IO2	IO3	IO4	IO5	IO6	IO7	IO8	LM to ACLA Interface
X	IF1	IF2	IF3	II1	II2	II3	II4	II5	II6	II7	II8	ACLA to LM Interface
1	1	1	0/1	A1	A2	A3	A4	A5	A6	A7	A8	Bit Definition

F+E

O D Dp

logic one if an ODD is present and is a logic zero if no ODD is present. This bit must be a logic one in an output loop address cell (OF3).

Address position A1 is the most significant bit and A8 is the least significant bit in the binary coded address.

DATA CELL

The data cell transfers information into or out of the ACLA via the loop multiplexer. The data cell format is presented in table 2-3. Bit D1 is always the first bit received from or transmitted to the modem by the ACLA.

SUPERVISION CELLS

The supervision cell on output gives information to the ACLA in the form of commands. On input it gives information to the processor in the form of status words from the ACLA.

STATUS WORDS

Most status changes, error conditions, or a supervision-report command causes status to be reported, and two words are sent to the processor. The status word 1 and status word 2 formats are shown in tables 2-4 and 2-5, respectively. In both tables a logic one indicates that the associated modem signal or status condition is active (on), and a logic zero indicates not active (off).

TABLE 2-3. DATA CELL FORMAT

11	10	9	8	7	6	5	4	3	2	1	0	Processor Word Bit Position
B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	Loop Cell Bit Position
X	OF1	OF2	OF3	IO1	IO2	IO3	IO4	IO5	IO6	IO7	IO8	LM to ACLA Interface
X	IF1	IF2	IF3	II1	II2	II3	II4	II5	II6	II7	II8	ACLA to LM Interface
1	1	0	0	D8	D7	D6	D5	D4	D3	D2	D1	8- or 9-bit char
1	1	0	0	0	D7	D6	D5	D4	D3	D2	D1	7-bit character
1	1	0	0	0	0	D6	D5	D4	D3	D2	D1	6-bit character
1	1	0	0	0	0	0	D5	D4	D3	D2	D1	5-bit character

TABLE 2-4. STATUS WORD 1 FORMAT

11	10	9	8	7	6	5	4	3	2	1	0	Processor Word Bit Position
B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	Loop Cell Bit Position
X	IF1	IF2	IF3	II1	II2	II3	II4	II5	II6	II7	II8	ACLA to LM Interface
1	1	0	1	CTS	DSR	DCD	RI	SDCD		ILE	OLE	Bit Definition

Bit Definitions

CTS = Clear-to-send. This status bit indicates the state of the modem signal clear-to-send. When active, it indicates the modem is ready to accept data from the ACLA. It must be active to enable data output from the CLA. If this signal

changes from a logic one to a logic zero during character output, the current character is completed and the transmit-data line is set to marking. A change of state of this signal causes status to be reported.

DSR = Data-set-ready. This status bit indicates the state of the modem signal data-set-ready. When active, it indicates that power is applied to the modem and that it is connected to the communications line. Any change of state of this signal causes status to be reported.

DCD/ RLSD = Data-carrier-detector. This status bit indicates the state of the modem signal receive-line-signal-detector (RLSD). When active, it indicates that a carrier signal is being received by the modem. Any change of state of this signal causes status to be reported.

RI = Ring-indicator. This status bit is set and status reported each time the modem signal ring-indicator goes from an on state to the off state. This indicates that the modem is receiving an incoming call from a remote station. The status bit is reset when the status words are sent to the processor.

SDCD/ SRLSD = Secondary-data-carrier-detect. Indicates state of modem signal secondary-receive-line-signal-detector (secondary-receive-data on some modems). On modems equipped with a reverse channel receiver, supervisory information can be received from a remote station while the ACLA is transmitting data to the station over a half-duplex, 2-wire circuit. Any change of state of this signal causes status to be reported.

ILE = Input-loop-error. When set to a one, the bit indicates that the LM has detected a loop error while the ACLA was using the input loop. This status is reset when the status words are sent to the processor.

OLE = Output-loop-error. When set to a one, the bit indicates that the LM has detected a loop error while the ACLA was using the output loop. This status is reset when the status words are sent to the processor.

TABLE 2-5. STATUS WORD 2 FORMAT

11	10	9	8	7	6	5	4	3	2	1	0	Processor Word Bit Position
B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	Loop Cell Bit Position
X	IF1	IF2	IF3	II1	II2	II3	II4	II5	II6	II7	II8	ACLA to LM Interface
1	1	0	1	PES	DTO	FES						Bit Definition

Bit Definitions

PES = Parity-error-status. This status is generated when the ACLA has been instructed to check for even or odd character parity and a character is received with incorrect character parity. The status always appears in the same line frame as the character. This status is reset when the next character with correct parity is received.

DTO = Data-transfer-overrun. This status is generated by the ACLA when it has a data char-

acter that is ready for transfer to the LM before the LM has picked up the previously assembled character. The previously assembled character is lost. This status is reset when the status words are sent to the processor.

FES = Framing-error-status. This status is posted by the ACLA when a character is received from the modem without the presence of a stop bit. The status always appears in the same line frame as the character. This status is not reset until another character with proper stop bit is received.

NOTE

FES can be set independently of the state of the ION (input-on) command. The character received that caused FES is always transferred. See Programming Notes for use of FES in break detection.

COMMAND WORDS

The command cells are instruction commands from the processor in the form of command words 1, 2, 3 and 4--these must be received in sequence. For example, words 1 and 2 must be received before word 3, and word 2 must always be preceded by command word 1. However, command word 1 can be received as a single word command. Formats for command word 1 and command word 2 are shown in tables 2-6 and 2-7. A logic one, in the position indicated, activates the associated signal, while a logic zero deactivates the signal. The commands operate independently of each other.

Each time a command is given to the ACLA, each bit (IO1-IO8) must be set to the condition desired on the associated function. Failure to do this will result in the condition being cleared or set to a different condition. Each time a command is given, the ACLA monitors each bit and takes the action dictated.

NOTE

When IO1 thru IO8 in command words 1 and 2 are all set to logic zero, the ACLA is cleared to a known state which disables any input (output-data-demand, status, or data) from the ACLA to HCP.

TABLE 2-6. COMMAND WORD 1 FORMAT

11	10	9	8	7	6	5	4	3	2	1	0	Processor Word Bit Position
B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	Loop Cell Bit Position
X	OF1	OF2	OF3	IO1	IO2	IO3	IO4	IO5	IO6	IO7	IO8	LM to ACLA Interface
1	1	0	1	RTS	SRTS	OM	LM	DTR	TB	ION	OON	Bit Definition

Bit Definitions

RTS = Request-to-send. A logic one activates the request-to-send line to the modem and a logic zero in this position deactivates RTS.

SRTS = Secondary request-to-send. A logic one activates the secondary-request-to-send line to the modem (referred to as secondary-send-data on some modems). On modems equipped with a reverse channel transmitter, supervisory information can be sent to a remote station while the ACLA is receiving data from the station over a half-duplex, 2-wire circuit. Typical uses include circuit assurance, error control, and interrupt

(break). A logic zero deactivates SRTS.

OM = Originate mode/auxiliary. A logic one in this position causes the ACLA to notify the modem equipment that it is in the originate mode. A logic zero indicates answer mode. This line is an auxiliary signal line and may be used for other functions as designated by system design.

LM = Local mode/auxiliary. A logic one in this position causes the ACLA to notify the modem (when equipped) to loop back the analog signal on the modem. A logic zero disables the loopback. This line is an auxiliary signal line and may be used for other functions as designated by system design.

DTR = Data-terminal-ready. A logic one in this location causes the ACLA to notify the modem that the system is ready to communicate with the modem. A logic zero causes a not-ready signal to be reported.

TB = Terminal-busy. A logic one causes the ACLA to notify the modem to busy-out-the-line. A logic zero disables this function.

ION = Input-on. When this bit is a logic zero, the input section of the ACLA will not receive data characters nor transfer data to the LM. A logic one causes normal input operation.

OON = Output-on. A logic one causes the output section of the ACLA to report output-data-demand initially when command is received if clear-to-send is active, and enables the output to report output-data-demand whenever the output buffer is empty. A logic zero inhibits reporting of output-data-demand.

TABLE 2-7. COMMAND WORD 2 FORMAT

11	10	9	8	7	6	5	4	3	2	1	0	Processor Word Bit Position
B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	Loop Cell Bit Position
X	OF1	OF2	OF3	IO1	IO2	IO3	IO4	IO5	IO6	IO7	IO8	LM to ACLA Interface
1	1	0	1	Break	ISR	ISON	DLM	RSR1	RSR2	TSR1	TSR2	Bit Definition

Bit Definitions

BREAK = Break-mode. A logic one in this position causes the ACLA to place the transmit-data line in a spacing condition (zero state). A logic zero inhibits the break operation and returns the line to marking condition (one state).

ISR = Input-supervision-report. A logic one in this position causes the ACLA to report the status of the RS-232 interface lines and any other ACLA status that may be active once each time the command is received. The ISR command is honored only when the ACLA has previously received a logic one in the ISON position. This is a momentary nonstored command. (If ISR = 1 and ISON = 1 appear in the same line frame, status is reported.)

ISON = Input-supervision-on. When a logic one is placed in this bit position, the ACLA monitors the modem interface and reports input supervision. A logic zero inhibits monitoring and reporting. Status is not reported automatically when this command bit is first received by the ACLA.

The ACLA must receive either an ISR command or status change to report input supervision.

DLM = Data-line-monitor. A logic one in this position causes the ACLA to monitor the receive-data line that is in a break condition for one character time after reception of the command. This command is used to allow the processor to determine the length of a break condition on data input. This is a momentary, nonstored command.

RSR1, RSR2 = Receive-speed-range 1 and 2. This code causes the ACLA to select a reference frequency from the loop multiplexer to provide a range of baud rates selectable by bits IO1-IO4 in command word 4. See table 2-8.

TSR1, TSR2 = Transmit-speed-range 1 and 2. This code causes the ACLA to select a reference frequency from the loop multiplexer to provide a range of transmit baud rates selectable by bits IO5-IO8 in command word 4. Table 2-8 shows the code and related reference frequencies.

In table 2-8, the special frequency is provided by an optional 115.2 kHz oscillator attached to the LM backplane. This special frequency can be used to accommodate a baud rate that is not available from the range of transmit baud rates shown in table 2-9.

TABLE 2-8. CODE BITS AND REFERENCE FREQUENCIES

RF No.	Code				Reference Frequency	Baud Rate
	IO5	IO6	IO7	IO8		
A	0	0	0	0	9.6 kHz	45 to 100
B	1	0	1	0	19.2 kHz	100 to 600
C	0	1	0	1	153.6 kHz	600 to 9600
D	1	1	1	1	Special	Special

TABLE 2-9. COMMON BAUD RATES AND COMMAND CODES

Baud Rate	Speed - Command Word 4								Range - Command Word 2				
	Input				Output				RF No.	Input		Output	
	I1	I2	I3	I4	I5	I6	I7	I8		I5	I6	I7	I8
9600	1	1	1	1	1	1	1	1	C	0	1	0	1
7200 (Special)	1	1	1	1	1	1	1	1	D	1	1	1	1
4800	0	1	1	1	0	1	1	1	C	0	1	0	1
3600 (Special)	0	1	1	1	0	1	1	1	D	1	1	1	1
2400	0	0	1	1	0	0	1	1	C	0	1	0	1
1800 (Special)	0	0	1	1	0	0	1	1	D	1	1	1	1
1600	0	1	0	1	0	1	0	1	C	0	1	0	1
1200	0	0	0	1	0	0	0	1	C	0	1	0	1
1050	1	1	1	0	1	1	1	0	C	0	1	0	1
800	0	0	1	0	0	0	1	0	C	0	1	0	1
600	0	1	1	1	0	1	1	1	B	1	0	1	0
300	0	0	1	1	0	0	1	1	B	1	0	1	0
150	0	0	0	1	0	0	0	1	B	1	0	1	0
133.3	1	1	1	0	1	1	1	0	B	1	0	1	0
120	0	1	1	0	0	1	1	0	B	1	0	1	0
110	1	0	1	0	1	0	1	0	B	1	0	1	0
100	0	0	1	0	0	0	1	0	B	1	0	1	0
75	0	0	0	0	0	0	0	0	B	1	0	1	0
66.67	1	1	1	0	1	1	1	0	A	0	0	0	0
50.0	0	0	1	0	0	0	1	0	A	0	0	0	0

COMMAND WORD 3. The format for this command word is shown in table 2-10.

TABLE 2-10. COMMAND WORD 3 FORMAT

11	10	9	8	7	6	5	4	3	2	1	0	Processor Word Bit Position
B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	Loop Cell Bit Position
X	OF1	OF2	OF3	IO1	IO2	IO3	IO4	IO5	IO6	IO7	IO8	LM to ACLA Interface
1	1	0	1	PSET	PI	CO1	CO2	SB		ECHO	LIT	Bit Definition

10 11

Bit Definitions

PSET = Parity-set. When this bit is a logic one, concurrent with PI set to a logic zero, the ACLA generates and checks for even parity. A logic zero concurrent with PI set to a logic zero causes the ACLA to generate and check for odd character parity.

PI = Parity-inhibit. A logic zero in this position causes the ACLA to check character parity on input and generate character parity on output. A logic one causes the ACLA to ignore parity.

NOTE

When parity is selected (PI=0), the ACLA does not transfer the received parity bit to the loop multiplexer in the data cell. A logic zero occupies the parity bit position within the character.

CO1, CO2 = Code 1 and Code 2 bits form a code so that each combination corresponds to a character length of either 5, 6, 7 or 8 bits. The checking and generation of character parity adds one information bit to the character and therefore must be considered when selecting the unit code. Table 2-11 shows these code bits in relation to parity-inhibit bit.

SB = Stop-bit. A logic one in this bit causes the output logic to generate two stop bits (1.5 stop bits on 5 data bits) on output and a logic zero causes one stop bit to be generated.

ECHO = Echoplex mode. A logic one in this position causes the ACLA to return all data received from the modem on the receive-data line back to the modem on the send-data line while maintaining normal data processing in the input logic. A logic zero inhibits echoplex operation.

LIT = Loop-internal-test. A logic one in this position causes the ACLA to go into an echoplex mode. Data and modem control signals from the output section are routed (looped back) to the input section. Refer to Programming Notes for additional information on this mode of operation. A logic zero disables echoplex mode.

COMMAND WORD 4. This command word format is shown in table 2-12.

TABLE 2-11. CODE AND PARITY INHIBIT BITS

PI	CO1	CO2	Code Level (No Parity)	PI	CO1	CO2	Code Level (Including Parity)
1	0	0	5	0	0	0	6
1	1	0	6	0	1	0	7
1	0	1	7	0	0	1	8
1	1	1	8	0	1	1	9

TABLE 2-12. COMMAND WORD 4 FORMAT

11	10	9	8	7	6	5	4	3	2	1	0	Processor Word Bit Position
B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	Loop Cell Bit Position
X	OF1	OF2	OF3	IO1	IO2	IO3	IO4	IO5	IO6	IO7	IO8	LM to ACLA Interface
1	1	0	1	F/1	F/2	F/4	F/8	F/1	F/2	F/4	F/8	Bit Definition
	Format			Receive				Transmit				

Command word 4 is used in conjunction with bits IO5, IO6, IO7 and IO8 of command word 2 to set the baud rate for the input and output sections. The input and output baud rates may differ. To determine the proper code for the baud rate, select the range from command word 2 in which the desired rate is contained and the desired baud rate from command word 4 as shown in table 2-9.

BAUD RATE DETERMINATION. The baud rate can be determined from an algorithm using the appropriate bits (4 bits on input or 4 bits on output) from command word 4 in conjunction with the range selected from command word 2. Proceed as follows to find the baud rate:

1. Complement each bit of the two 4-bit fields of command word 4. Convert each result to a base ten number using IO4 as the most significant bit for the input field and IO8 as the most significant bit for the output field. Add one to each result.

2. Multiply the above result by 16_{10}.

3. Divide this product into the reference frequency selected from table 2-9.

EXAMPLE:

Given:

from command word 4 - IO4 IO3 IO2 IO1 = 0 1 0 1

from command word 2 - IO5 IO6 = 1 0 (19.2kHz)

Complement $0101_2 = 1010_2 = 10_{10}$

Add one to the result $= 1011_2 = 11_{10}$

Then, the baud rate $= \frac{19.2 \times 10^3}{11 \times 16} =$

109.1 baud

NOTE

The ACLA must be able to receive data that might have up to 40 percent distortion, that is, 40 percent of a bit per character. The result found above is valid for a 110 baud communications line, having .82 percent tolerance of the nominal value of 110 baud.

PROGRAMMING NOTES

The following notes provide additional information on the operation of the ACLA and are intended to assist the programmer. Typical input and output operations are presented.

ACLA INITIALIZATION

Before the ACLA is used following a power-up situation (either power up of a system in which the ACLA is already installed, or installation of the ACLA in an operating system), it should be cleared. Clearing the ACLA in this situation requires that the program:

SUPERVISION CELLS

The supervision cell on output gives information to the SCLA in the form of commands. On input this cell gives information to the processor in the form of status words from the SCLA.

STATUS WORDS

Most status changes, error conditions, or a status request command cause status to be reported, and both words are sent to the processor. The Status Word 1 and Status Word 2 formats are shown in tables 2-4 and 2-5, respectively. In both tables a logic one indicates that the associated modem signal or status condition is active (on), and a logic zero indicates not active (off).

TABLE 2-4. STATUS WORD 1 FORMAT

11	10	9	8	7	6	5	4	3	2	1	0	Processor Word Bit Position
B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	Loop Cell Bit Position
X	IF1	IF2	IF3	II1	II2	II3	II4	II5	II6	II7	II8	SCLA to LM Interface
1	1	0	1	CTS	DSR	DCD	RI	QM*	SQD*	ILE	OLE	Bit Definition

*2560-1 only. A logic 0 appears in these positions for 2560-2 and 2560-3.

Bit Definition

CTS = Clear-to-send. This status bit indicates the state of the modem signal clear-to-send. When active, it indicates the modem is ready to accept data from the SCLA. It must be active to enable data output from the CLA. If this signal changes from a logic one to a logic zero during character output, the current character is completed and the transmit-data line is set to marking. A change of state of this signal does not cause status to be reported.

DSR = Data-set-ready. This status bit indicates the state of the modem signal data-set-ready. When active, it indicates that power is applied to the modem and that it is connected to the communications line. Any change of state of this signal causes status to be reported.

DCD = Data-carrier-detect. This status bit indicates the state of the modem receive-line-signal-detector signal. When active, it indicates that a carrier signal is being received by the modem. Any change of state of this signal causes status to be reported. (For the 2560-2 SCLA connected to an AT&T 303 Wideband Data Station, this status is called AGC Lock.)

RI = Ring-indicator. This status bit is set and status reported each time the modem ring-indicator signal goes from an on state to the off state. This indicates that the modem is receiving an incoming call from a remote station. The status bit is reset when the status words are sent to the processor.

QM = Quality-monitor. This status bit indicates the state of the quality-monitor modem

signal. When off or zero, it indicates that the adaptive equalizer in the modem receiver is reset or retraining itself automatically due to poor error performance. Any data received has a "high" probability of error. When on or one, the automatic equalizer is in its normal trained mode and received data should have a "low" probability of error. Any change of state of this signal causes status to be reported.

SQD = Signal-quality-detector. This status bit indicates the state of the modem signal signal-quality-detector. It functions similarly to DCD but provides a fast responding indication of the presence (one) or absence (zero) of data carrier signal from the remote station. Any change of state of this signal causes status to be reported.

ILE = Input-loop-error. When set to a one, the bit indicates that the LM has detected a loop error while the SCLA was using the input loop. This status is reset when the status words are sent to the processor.

OLE = Output-loop-error. When set to a one, the bit indicates that the LM has detected a loop error while the SCLA was using the output loop. This status is reset when the status words are sent to the processor.

TABLE 2-5. STATUS WORD 2 FORMAT

11	10	9	8	7	6	5	4	3	2	1	0	Processor Word Bit Position
B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	Loop Cell Bit Position
X	IF1	IF2	IF3	II1	II2	II3	II4	II5	II6	II7	II8	SCLA to LM Interface
1	1	0	1	PES	DTO	0	NCNA	0	0	0	0	Bit Definition

D

Bit Definition

PES = Parity-error-status. This status is generated when the SCLA has been instructed to check for even or odd character parity and a character is received with incorrect character parity. The status always appears in the same line frame as the character. This status is reset when the status words are sent to the processor.

DTO = Data-transfer-overrun. This status is generated by the SCLA when it has a data character that is ready for transfer to the LM before the LM has picked up the previously assembled character. The previously assembled character is lost. This status is reset when the status words are sent to the processor.

NCNA = Next-character-not-available. This status indicates that while the SCLA was in the process of transmitting, the next character was not received from the LM in time to maintain continuous character output. Thus character frame synchronization was lost. The output section must be enabled (output-on active) for this status to occur. The status sets only once and does not repeat until the NCNA condition is alleviated (by

receiving a character to output) and then can recur. The status is reset when the status words are transferred to the processor.

COMMAND WORDS

The command cells are instruction commands from the processor in the form of command word 1, command word 2, and command word 3. Formats for command word 1 and command word 2 are shown in tables 2-6 and 2-7. A logic one, in the position indicated, activates the associated signal, while a logic zero deactivates the signal. The commands operate independent of each other.

TABLE 2-6. COMMAND WORD 1 FORMAT

11	10	9	8	7	6	5	4	3	2	1	0	Processor Word Bit Position
B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	Loop Cell Bit Position
X	OF1	OF2	OF3	IO1	IO2	IO3	IO4	IO5	IO6	IO7	IO8	LM to SCLA Interface
1	1	0	1	RTS	RSYN	0	NSYN LT*	DTR	0	ION	OON	Bit Definition

*2560-1 only. This bit position is used for LT for the 2560-2 and 2560-3.

BIT DEFINITIONS

RTS = Request-to-send. A logic one activates the request-to-send line to the modem while a logic zero in this position deactivates RTS.

RSYN = Resynchronize. When the input of the SCLA is activated, a logic one causes the SCLA to drop input synchronization and search for a new synchronization sequence. (This is a momentary, nonstored command. This bit position should normally be a logic zero.)

NSYN = New-sync. A logic one in this location tells the SCLA to notify the modem via the new-sync interface signal that another message is coming contiguous to the present message and the modem should drop and reestablish bit synchronization. This command should normally be on for one msec or more, depending on modem type (2560-1 only).

LT = Local-test. A logic one in this position causes the SCLA to notify the modem to route back transmit data to the SCLA input (2560-2 and 2560-3 only).

DTR = Data-terminal-ready. A logic one in this location causes the SCLA to notify the modem that the system is ready to communicate with the modem.

ION = Input-on. When this bit is a logic zero, the input section of the SCLA drops out of synchronization and does not receive data characters or transfer data to the LM. A logic one causes the normal acquisition of synchronization and transfer of data.

OON = Output-on. A logic one causes the output section of the SCLA to report output-data-demand initially when the command is received if clear-to-send is active, and enables the output to report output-data-demand whenever the output buffer is empty. A logic zero inhibits reporting of output-data-demand.

TABLE 2-7. COMMAND WORD 2 FORMAT

11	10	9	8	7	6	5	4	3	2	1	0	Processor Word Bit Position
B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	Loop Cell Bit Position
X	OF1	OF2	OF3	IO1	IO2	IO3	IO4	IO5	IO6	IO7	IO8	LM to SCLA Interface
1	1	0	1	0	ISR	ISON	LIT	PSET	PI	CO1	CO2	Bit Definition

BIT DEFINITIONS

ISR = Input-supervision-report. A logic one in this position causes the SCLA to report the status of the modem interface lines and any other SCLA status that may be active once each time the command is received. The ISR command is honored only when the SCLA has previously received a logic one in the ISON position. (If ISR = 1 and ISON = 1 appear in the same line frame, status is reported).

ISON = Input-supervision-on. When a logic one is placed in this bit position, the SCLA monitors the modem interface and reports status. A logic zero inhibits monitoring and reporting. Status is not reported automatically when this command bit is first received by the SCLA. The SCLA must receive either an ISR command or status change to report status.

LIT = Loop-internal-test. A logic one in this position causes the SCLA to go into an echo-test mode. Data and modem control signals from the output section are routed (looped back) to the input section. The SCLA also switches to an internal clock (2.4 kHz) supplied by the LM. Refer to Programming Notes for additional information on this mode of operation. A logic zero disables echo-test mode.

PSET = Parity-set. When this bit is a logic one, concurrent with PI set to a logic zero, the SCLA generates and checks for even parity. A logic zero concurrent with PI set to a logic zero causes the SCLA to generate and check for odd parity.

PI = Parity-inhibit. A logic zero in this position causes the SCLA to check character parity on input and generate character parity on output. A logic one causes SCLA to ignore parity.

CO1, CO2 Code 1 and Code 2 bits form a code so that each combination corresponds to a character length of either 5, 6, 7, or 8 bits. The checking and generation of character parity adds one information bit to the character and therefore must be considered when selecting the unit code level. Table 2-8 shows these code bits in relation to the parity-inhibit bit.

TABLE 2-8. CODE BITS

PI	CO1	CO2	Code Level (No Parity)	PI	CO1	CO2	Code Level (Including Parity)
1	0	0	5	0	0	0	6
1	1	0	6	0	1	0	7
1	0	1	7	0	0	1	8
1	1	1	8	0	1	1	9

Command Word 3 contains the synchronization character. This command is stored and is used for obtaining synchronization. When the SCLA is attempting to acquire character frame synchronization, a 5- to 8-bit sequence (depending on code length) is compared against this stored command at each bit time. The format for this synchronization character command is shown in table 2-9. Bit D1 is the first bit received from the modem.

TABLE 2-9. COMMAND WORD 3 FORMAT

11	10	9	8	7	6	5	4	3	2	1	0	Processor Word Bit Position
B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	Loop Cell Bit Position
X	OF1	OF2	OF3	IO1	IO2	IO3	IO4	IO5	IO6	IO7	IO8	Loop MUX to SCLA Interface
1	1	0	1	D8	D7	D6	D5	D4	D3	D2	D1	8- or 9-bit Character
1	1	0	1	0	D7	D6	D5	D4	D3	D2	D1	7-bit Character
1	1	0	1	0	0	D6	D5	D4	D3	D2	D1	6-bit Character
1	1	0	1	0	0	0	D5	D4	D3	D2	D1	5-bit Character

While the SCLA is attempting to synchronize, no characters are transferred to the LM. After synchronization is achieved, the SCLA transfers all characters to the LM until the SCLA is directed to resynchronize via command from the processor.

Each time a command is given to the SCLA, each bit in each command word must be set to the condition desired for the associated function. Failure to do this will result in the condition being cleared or set to a different condition. Each time a command is given to the SCLA, the SCLA reads each bit and takes the action dictated.

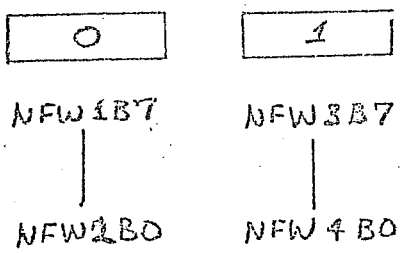
Whenever the SCLA receives command words, it assumes they are received in order. For example, to change command word 2, both words 1 and 2 must be sent to the SCLA in the same line frame. Changing command word 3 requires sending all three command words.

To bring the SCLA to an idle (cleared) state, set all bits in command words 1 and 2 to logic zero.

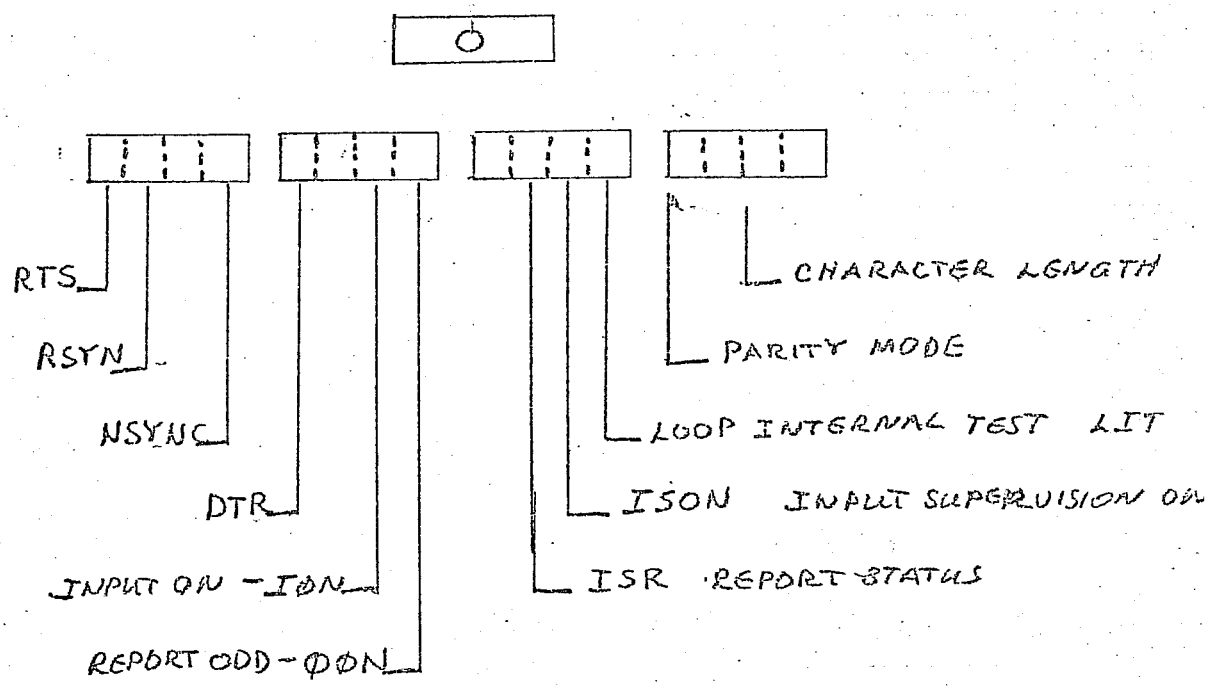
TABLE

(162) = FWA

NFCCSE = LAST COMMAND SENT TO CLA

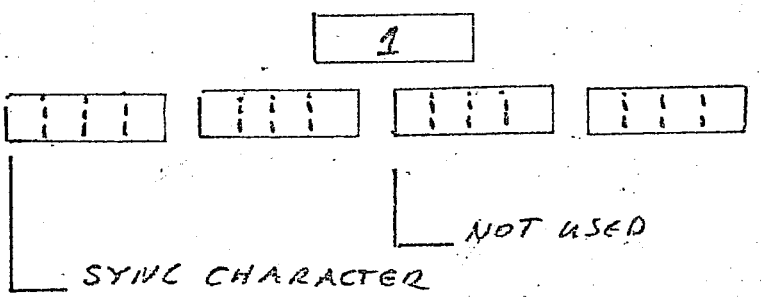


SYNCHRONOUS CLA

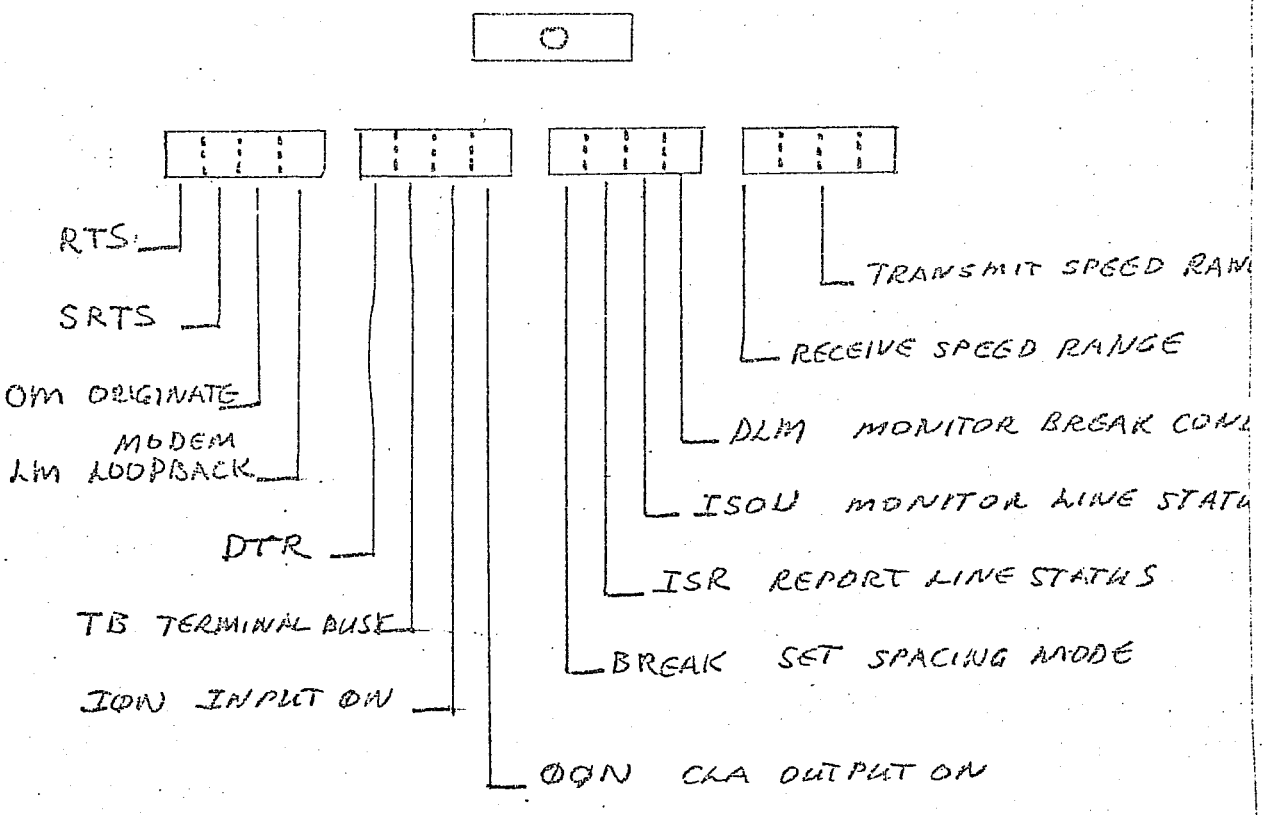


CHARACTER LENGTH 0=5 1=7 2=6 3=8

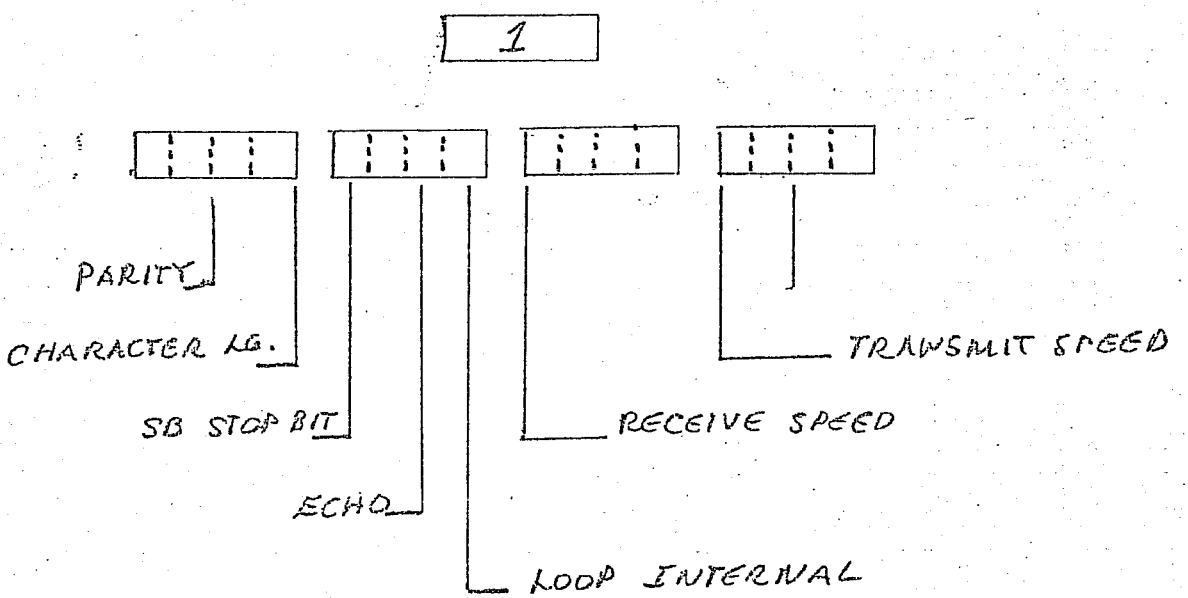
PARITY MODE 0 = ~~EVEN~~ PARITY ODD
1 = NO PARITY
2 = EVEN PARITY



ASYNCHRONOUS DATA



SPEED RANGE 0 = 45-100
1 = 600-9600
2 = 100-600

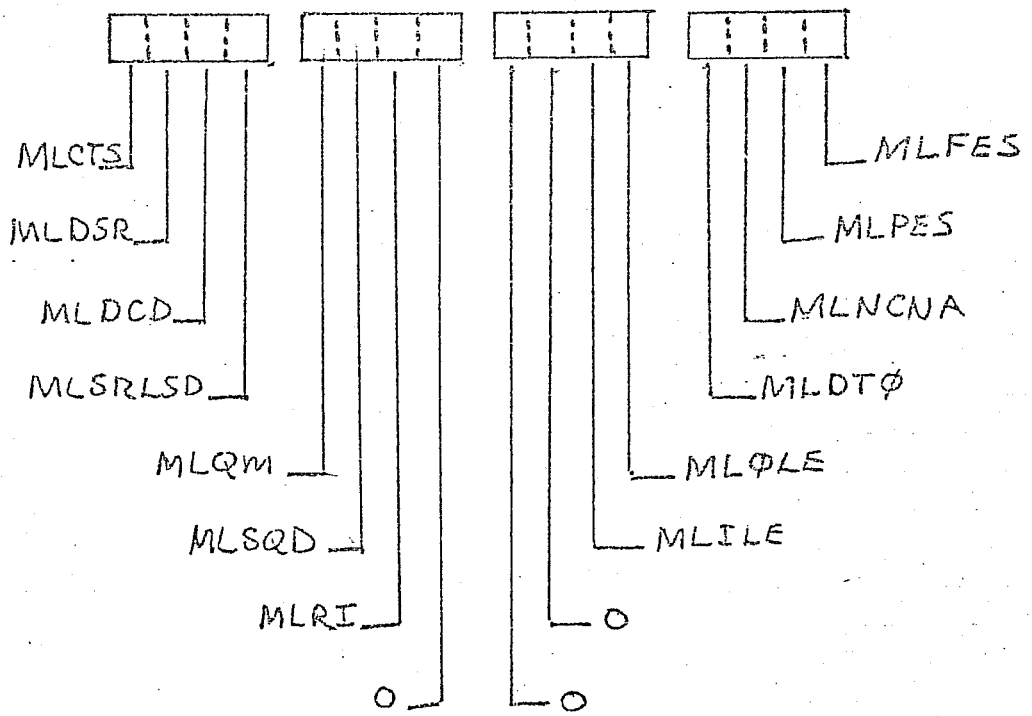


PARITY 0 = ODD PARITY
1 = NO PARITY
2 = EVEN PARITY

CHARACTER LENGTH 0 = 5 BITS
1 = 7 BITS
2 = 6 BITS 3 = 8 BITS

TABLE

MLASTAT (163) = FWA



MLCTS – CLEAR TO SEND

MLDSR – DATA SET READY

MLDCD – DATA CARRIER DETECTED

MLSRLSD – SECONDARY RECEIVE LINE SIG

MLQM – QUALITY MONITOR

MLSQD – SIGNAL QUALITY DETECT

MLRI – RING INDICATOR

MLILE – INPUT LOOP ERROR

MLOLE – OUTPUT LOOP ERROR

MLDTO – DATA TRANSFER OVERRUN

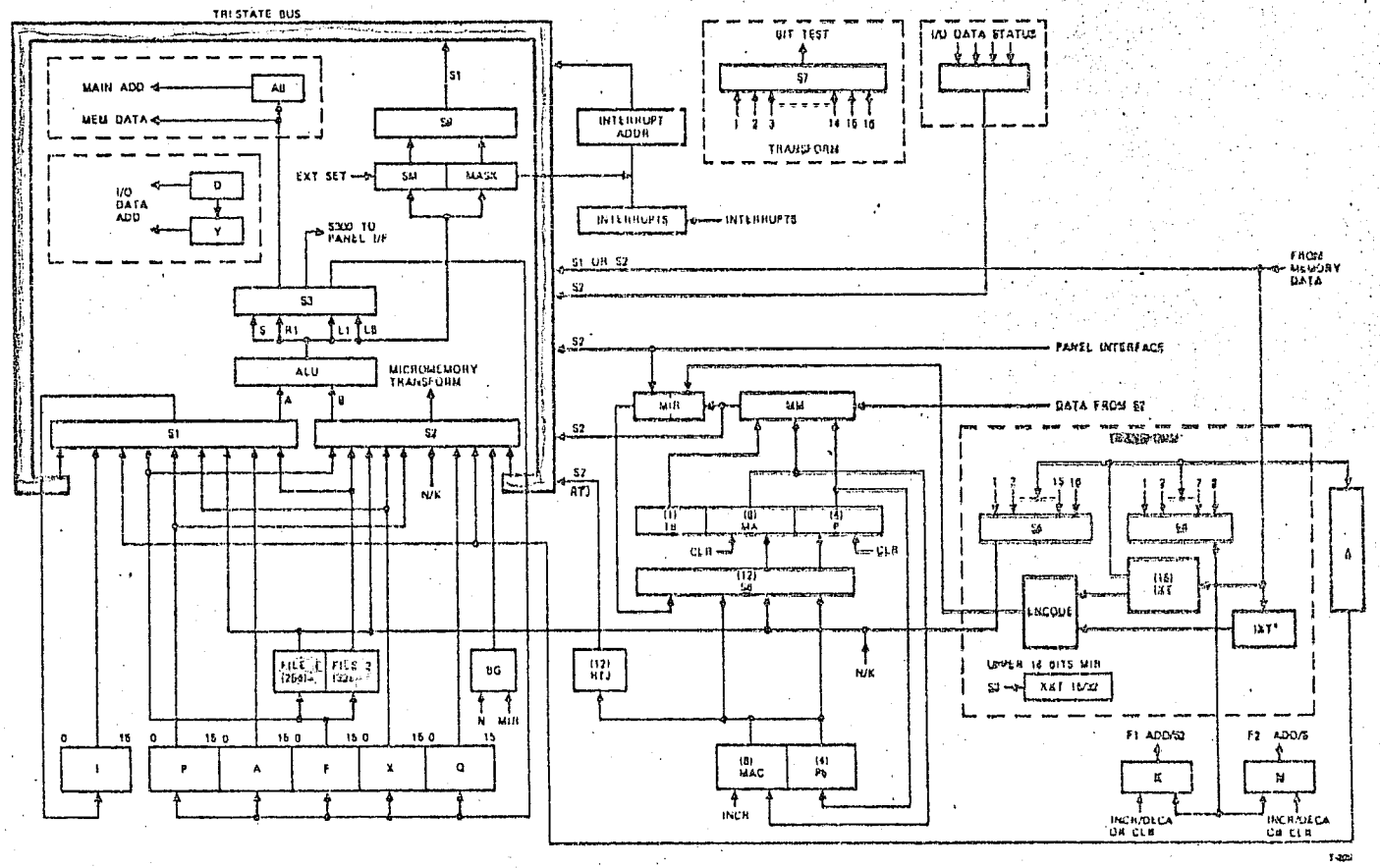
MLNCNA – NEXT CHARACTER NOT AVAILABLE

MLPES – PARITY ERROR

MLFES – FRAMING ERROR

TABLE 1-2. HCP FUNCTIONAL FEATURES

Main Memory
Capacity: 32,768 words (expandable to 256k words)
Word length: 18 bits (16 Data, 1 Parity, 1 Protect)
Read-access time: 330 or 280 nanoseconds
Memory addressing modes: 8
Memory word and region protection
Memory parity detection
Direct memory access (4 users)
External CPU Access
Automatic interlaced refresh
Micromemory
Capacity: 3,072 Instruction (2,048 read/write; 1,024 read only)
Instruction length: 32 bits - 2 per word
Read-access time: 168 nanoseconds
Interrupts
Macrointerrupts: 16
Microinterrupts: 16
Registers: 15
Files
File 1: 256 words } 16 Bits each
File 2: 32 words } 16 Bits each



Detailed Microprocessor Block Diagram

STATUS MODE BIT ASSIGNMENTS

Bit	Function
SM100	Not used (double precision)
SM101	One's Complement Arithmetic
SM102	Enable Bit Generator Input from N Register
SM103	Not used (split adder)
SM104	Macro Breakpoint Occurred (use with INT31/)
SM105	1700 Protect Fault Occurred
SM106	Enable Macro Interrupt System
SM107	Not used (enable decimal arithmetic)
SM108	Main Memory Parity Error Occurred
SM109	Enable Micromemory Halt
SM110	Overflow
SM111	Enable File 1 (Read/Write)
SM112	Not used (enable binary overflow)
SM113	Not used (enable R/W MM via transform)
SM114	Delay Enabling Macro Interrupts
SM115	MLIA Busy
SM200	Enable ADT Mode
SM201	Strobe or Read Data from MOS Device
SM202	Write Data to AQ Device
SM203	Terminate I/O Transfer
SM204	Enable Autoload
SM205	Enable MUX Subsystem Priority 3 Interrupt
SM206	Not used
SM207	Enable page selection with transform
SM208	Not used
SM209	Enable the unprotected instruction followed by a Protected Instruction Check
SM210	Write Data to Panel Device
SM211	Read Data to Panel Device
SM212	Write Data to FCR Register
SM213	Enable 1700 Enhanced Transform
SM214	Enable Console Control
SM215	Enable Macro Instruction Run

INTERRUPT BIT ASSIGNMENTS

T8010

Bit	Function
I100	MLIA
I101	Communications Console (TTY or CRT)
I102	MLIA
I103	(unassigned)
I104	2571/2570 Line Printer Controller
I105	(unassigned)
I106	(unassigned)
I107	Tape Cassette Controller
I108	Real Time Clock
I109	(unassigned)
I110	(unassigned)
I111	2571-2572 Card Reader Controller
I112	1700 Emulator
I113	1700 Emulator
I114	1700 Emulator
I115	1700 Emulator
I200	Power Fail/Memory Parity
I201	Communications Console (TTY or CRT)
I202	Multiplex Subsystem Priority 3 (source is SM205)
I204	2571/2570 Line Printer Controller
I205	CYBER Coupler 2 (2558-1)
I206	CYBER Coupler
I207	Tape Cassette Controller
I208	Real Time Clock
I209	1732/608-609 Mag Tape Controller (QSE)
I210	1740-1742 Line Printer Controller (QSE)
I211	2571-2572 Card Reader Controller
I212	MLIA
I213	MLIA
I214	(unassigned)
I215	Macro Breakpoint

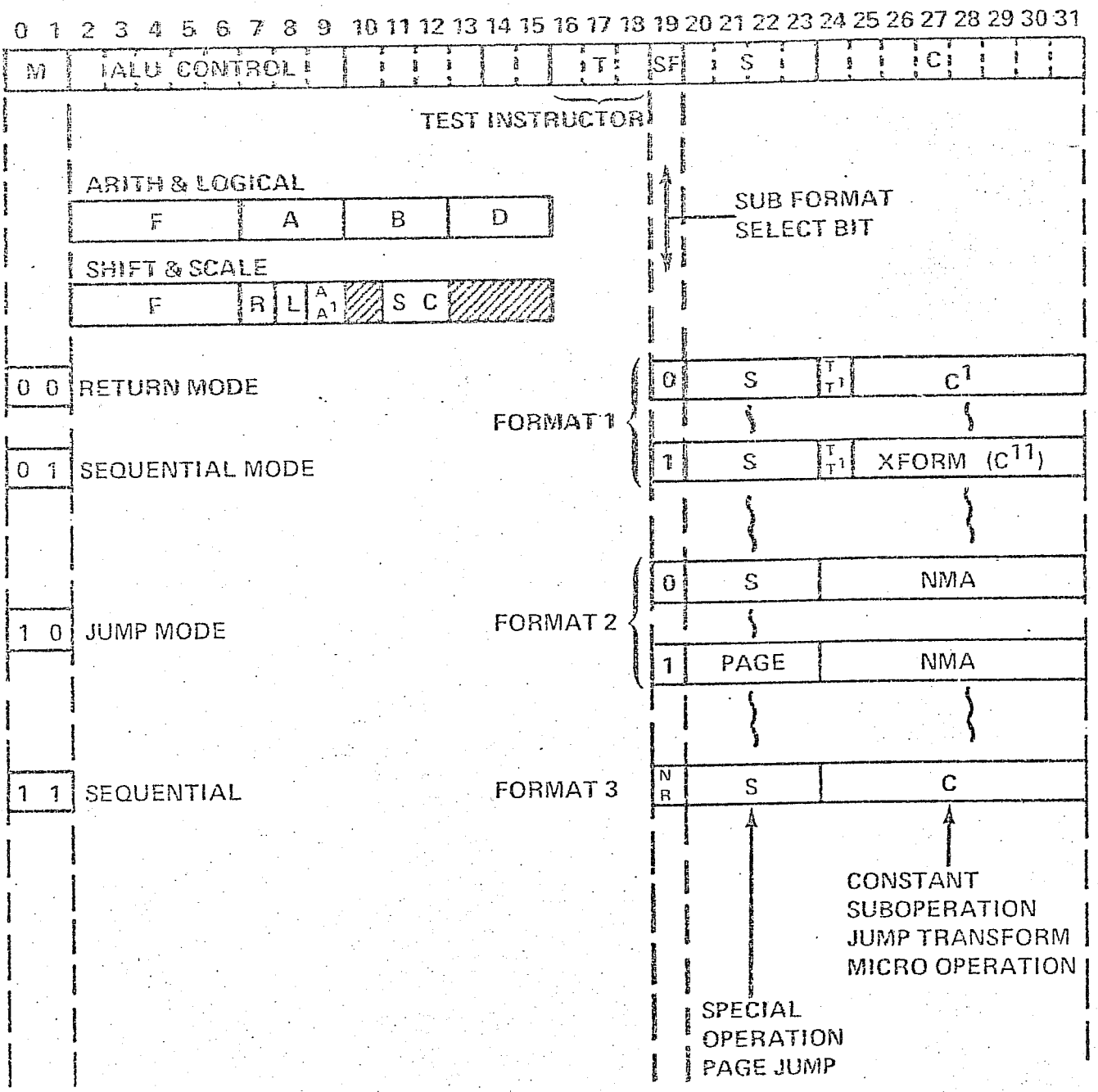
MICRO INTERRUPTS

MACRO INTERRUPTS

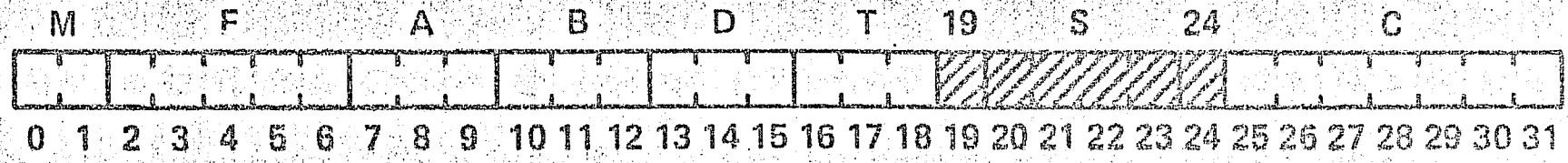
TR-30

I202 MLIA ERROR

MICRO INSTRUCTION FORMAT



MICRO INSTRUCTION FORMAT



S MODIFIES A-B-D
19 MODIFIES C
24 MODIFIES T

MICRO-CODE SUBROUTINES TO EMULATE LDA MACRO INSTRUCTION

T	P/MA	MICRO-MEM	LOCATION	F	A	B	D	S	C	MT	COMMENT
				DECODE NEXT INSTRUCTION							
0	058	6C79	0000 B	INI	SUB	P	MEM	P			INCR. TO NEXT INST., P = P + 1
1	058	D8D8	B307 G	-RNI				READ	N=F2IREG	INTU	READ NEXT INST.
0	059	4AF7	2830 B	+	-B		INTA	F	GATE IXT	U	PROCESS ACTIVE INTERRUPT
1	059	58D8	302C C	-XNI				PAGE0	GITMAKXT	U	TRANSFORM ON NEXT INST.
					o						
					o						
					o						
					o						
				LOAD A REGISTER F = C							
0	098	6C79	2300 G	+LDA	SUB	P	MEM	P	READ	U	READ (EA), P + 1 TO P
0	099	9FDD	4058	+	A	MEM		A	RNI	J	(EA) TO A, GO TO RNI

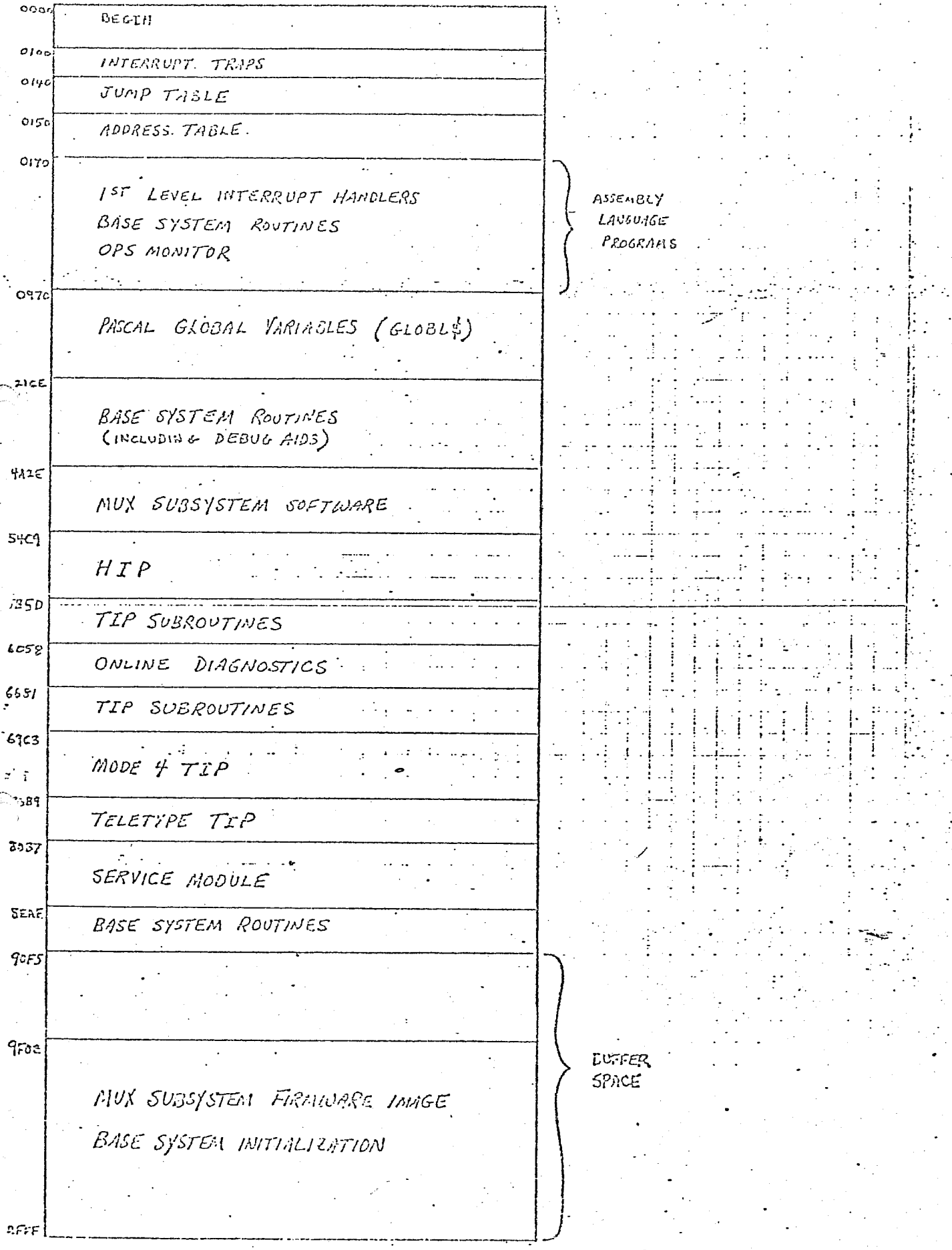

0000
BEGIN
0100
INTERRUPT TRAPS
0140
JUMP TABLE
0150
ADDRESS TABLE
0170
1ST LEVEL INTERRUPT HANDLERS
BASE SYSTEM ROUTINES
OPS MONITOR
ASSEMBLY LANGUAGE PROGRAMS
0970
PASCAL GLOBAL VARIABLES (GLOBL$)
21CE
BASE SYSTEM ROUTINES (INCLUDING DEBUG AIDS)
4A2E
MUX SUBSYSTEM SOFTWARE
54C9
HIP
735D
TIP SUBROUTINES
6058
ONLINE DIAGNOSTICS
6651
TIP SUBROUTINES
69C3
MODE 4 TIP
7589
TELETYPE TIP
8037
SERVICE MODULE
8EAE
BASE SYSTEM ROUTINES
90F5
9F02
MUX SUBSYSTEM FIRMWARE IMAGE
BASE SYSTEM INITIALIZATION
BFFF
BUFFER SPACE

TABLE 4-4. PIPRINT ADDRESS AREA FORMAT

$150	0	BYWLCB	WORKLIST CONTROL BLOCK	BASE
	1	JSWLADDR	WL ENTRY BY LEVELNO	
	2	B1TCB	INTERNAL PROCESSING TCB	
	3	B1BUFF	INTERNAL PROCESSING BLOCK	
	4	JKMASK	INTERRUPT MASKS	
	5	JKTMASK	PBAMASK SAVE AREA	
	[illegible]		[illegible]	
	6	CBTIMTBL	TIMAL TABLE	
	7	JACT	PD CONTROLLER TABLE	
	8	BECTLBK	BUFFER CONTROL BLOCK	
	9	BEBSA	BUFFER STAMP AREA	
	A	CLBFSPACE	BFR SPACE IN NO. SMALL BFRS	
	B	BKPIKT	POI TABLE	
	C	J1UERRSTAT	UPLINE ERR-STAT ROUTING	
	D	J1USVM	UPLINE SERVICE MSG ROUTING	
	E	J1DSVM	DOWNLINE SERVICE MSG ROUTING	
	F	0		
160	10	NAPORT	PORT TABLE	MUX SUBSYSTEM
	11	BQCIB	CIRCULAR INPUT BUFFER	
	12	NECCST	CLA CMD STATUS TABLE	
	13	MLSTABLE	CLA CURRENT STATUS TABLE	
	14	0		
	15	CGLCBS	LINE CONTROL BLOCKS	LINES - TIPS
	16	CHSUBLCB	SUB LCBS	
	17	BJTIPTYPT	TIP TYPE TABLE	
	18	NJTECT	TERMINAL CHARACTERISTICS TABLE	
	19	0		
	1A	JF1SNPTBLE	SNAPSHOT CORE TABLE	DEBUG AIDS
	1B	JFWRAP	WRAP-SNAP TABLE	
	1C	J1STRT	START ADDR FOR PBDUMP	
	1D	J1QUICKPTR	BFR PTR FOR PBQUICKIO	
	1E	0 NPINTAB	INIT SEQ COMPT AND ERROR TABLE	

MONITOR
OPS
PROGRAM SCAN SEQUENCE

1. PRIORITY PROGRAMS

A. PBINTPROC – INTERNAL PROCESSING
WORKLIST – BOINWL

B. PTSTART -- COUPLER HOST INTERFACE PROCESSING
WORKLIST – BOCOWL

C. PTTYP – TTY TERMINAL INTERFACE PROCESSING
WORKLIST – BOTYWL

D. PTMOD4T – MODE 4 TERMINAL INTERFACE PROCESSING
WORKLIST – BOM4WL

E. PBTIMAL – TIMING SERVICES
WORKLIST – BOTIWL

2. NON PRIORITY PROGRAMS

A. PBCONSOLE – CONSOLE INTERFACE PROCESSING
WORKLIST – BOCHWL

B. PBCONTINUE – RETURN CONTROL TO TIPS AT OPS LEVEL
WORKLIST – BOCOWQ

C. PBTPDBG – DEBUG ROUTINE
WORKLIST – BOTYWD

D. PBMLIAOPS – MUA PROCESSOR
WORKLIST – BOMLWA

E. PNSMWL – SERVICE MODULE PROCESSOR
WORKLIST – BOSMWL

LEVEL 8 PROCEDURE

PBTIMER

I28B REAL TIME CLOCK INTERRUPT INITIATED

CAUSE: REAL TIME CLOCK COUNT EQUALS CLOCK LIMIT EVERY 30MS OR LESS

CALLS: PMT200M EVERY 200MS TO CHECK FOR CHARACTER TIMEOUT

PBLSPUT EVERY 500MS TO ENTER WORKLIST ENTRY IN OPS PROGRAM PBTIMAL TO CHECK WAIT TIMER FIELDS FOR TIPS

LEVEL 6 PROCEDURE

PTINTPROC

I206 PRIMARY CYBER COUPLER INTERRUPT

LEVEL 5 PROCEDURE

PTINTPROC

I205 SECONDARY CYBER COUPLER INTERRUPT

CAUSE: NORMAL
1. CHAIN ADDRESS ZERO
2. CHANNEL ACTIVE 3 SECONDS
3. NPU STATUS ACCEPTED
4. ORDER WORD LOADED
5. TRANSFER TERMINATED BY PPU
6. TRANSFER COMPLETE

ERROR
1. PPU CHANNEL PARITY ERROR
2. MEMORY PROTECT ERROR
3. MEMORY PARITY ERROR

LEVEL 3 PROCEDURE

PMWOLP

I203 STATUS MODE S205 INTERRUPT

CAUSE: INITIATED BY MICRO CODE SEQUENCE N2P3INTA WHICH SETS SM205 IN STATUS MODE REGISTER. THE MICRO SEQUENCE IS CALLED BY OPS LEVEL PROGRAMS WISHING TO PASS CONTROL TO MUX DRIVER

CALLED BY: PBLCBTMSCN - TIMING SERVICE FOR TIPS TO RETU TO MUX LEVEL AFTER WAIT TIME H EXPIRED

PMCDRV - MUX COMMAND DRIVER TO TERMINAT INPUT OR OUTPUT

PMT200M - MUX SUBSYSTEM 200 MS TIMER TO RETURN TO MUX LEVEL AFTER WAIT CONDITION AND CHARACTER TIMEO

PMT1SEC - MUX SUBSYSTEM 1 SEC TIMER TO RETURN TO MUX LEVEL AFTER MOD TIMEOUT OR ODD TIMEOUT

PTWAIT - RETURN TO MUX LEVEL AFTER WAIT EVENT HAS OCCURRED

LEVEL 2 PROCEDURE

PMMLEH

I202 MLIA ERROR INTERRUPT

CAUSE:
1. ODD LIMIT REACHED
2. MLIA INPUT BUFFER FULL
3. OUTPUT LOOP ERROR
4. INPUT LOOP ERROR

LEVEL 1 PROCEDURE

PBTTYINT

I201 CONSOLE TTY INTERRUPT

CAUSE: 1. MANUAL INTERRUPT FROM CONSOLE
2. END OF OPERATION
3. DATA INPUT

LEVEL 0 PROCEDURE

PBLN00

I200 POWER FAIL / PARITY INTERRUPT

CAUSE: 1. MEMOAY PARITY ERROR
2. MEMORY PROTECT FAULT
3. POWER FAILURE

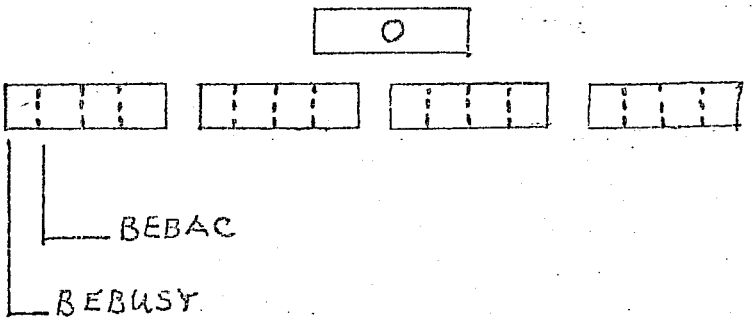
CALLS: PBBREA – BREAKPOINT PROCESSOR CALLED IF PROTECT FAULT TO CHECK FOR BREAKPOINT CONDITION

BCB BUFFER CONTROL BLOCKS (free buffer chain)

(158) = FWA

0	1	2	3	4	5	6	7
BEBUSY BEBAC	BENFB	BELFB	BEMSK	BELEN	BETRS1	BEWOR1	BEWOR2

8	9
BEDUM1 BEINDX BE DUM2	BECOMMSK



BEBUSY — INTERLOCK FLAG

BEBAC — BUFFERS AVAILABLE COUNT

1 POINTS TO LAST WORD OF BUFFER

BENFB — NEXT FREE BUFFER

2

BELFB — LAST FREE BUFFER IN CHAIN

3

BEMSK — MASK FOR CHAIN ERROR CHECKING

4

BELEN — LENGTH OF BUFFERS IN THIS POOL

5

BETRS1 — LOWER THRESHOLD FOR THIS BUFFER TYPE

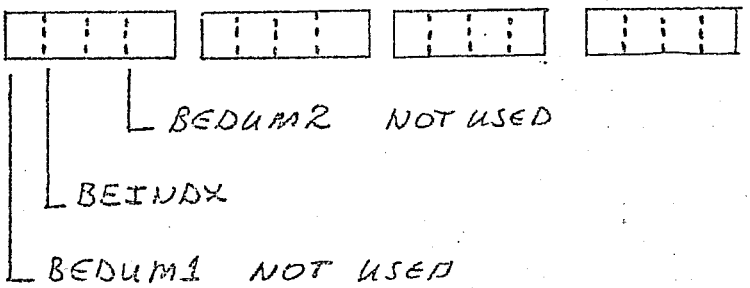
6

BEWDR1 — INITIAL VALUE OF FIRST WORD OF BUFFER
81LCD, 81FCD

7

BEWDR2 — INITIAL VALUE OF SECOND WORD OF BUFFER

8

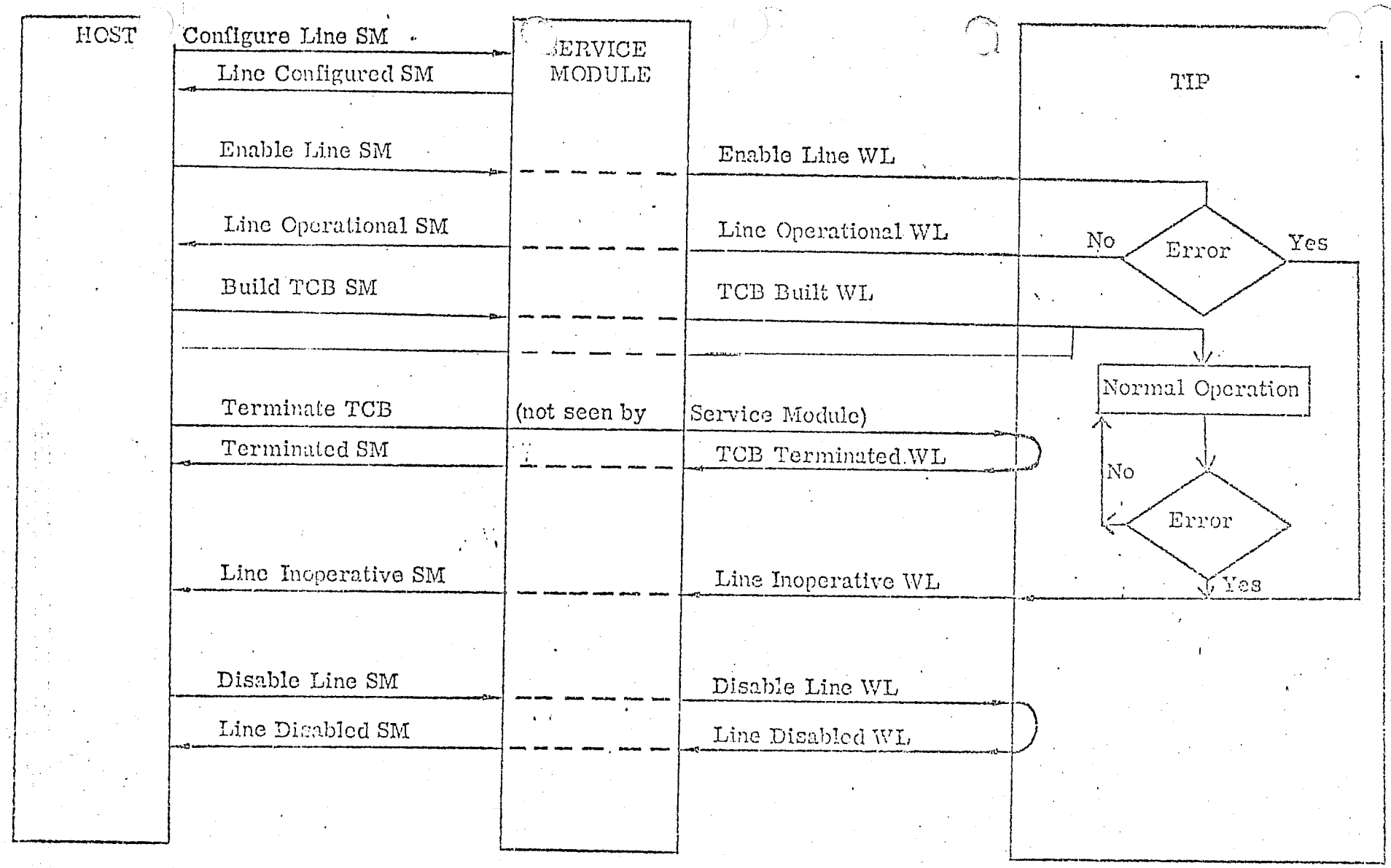


BEINDX — BUFFER SIZE

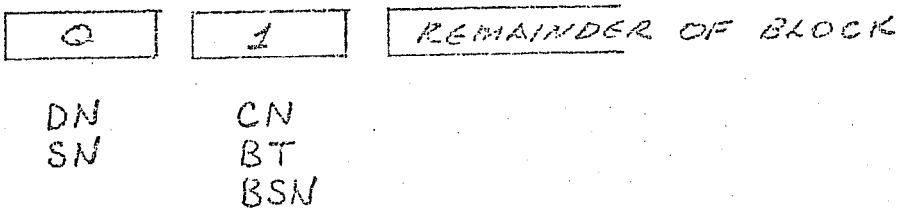
0 = BOS0	0D	CHARACTERS
1 = BOS1	1D	CHARACTERS
2 = BOS2	3D	CHARACTERS
3 = BOS3	7D	CHARACTERS

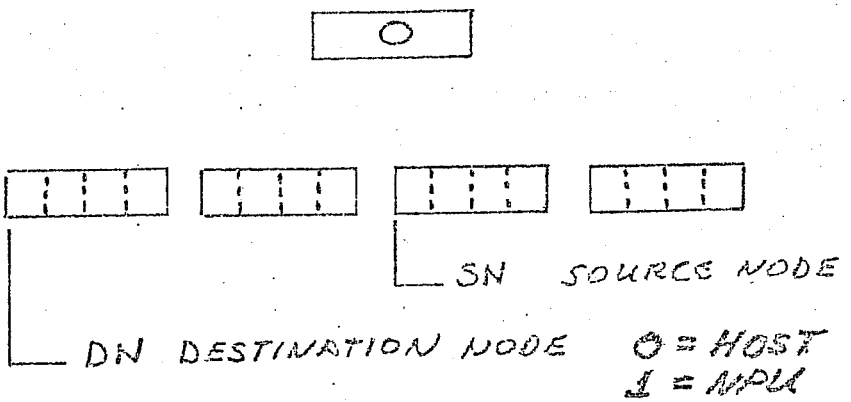
9

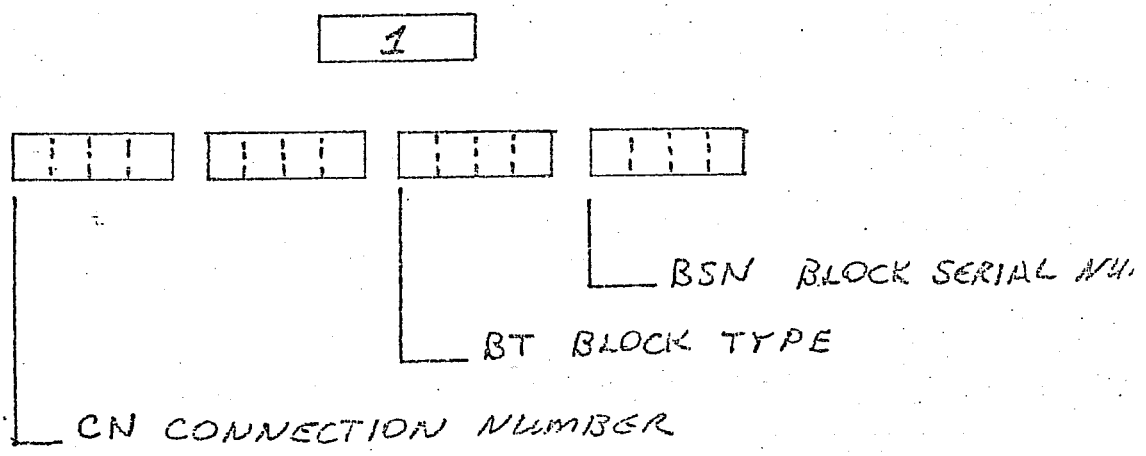
BECOMMSK — SETWORD



SM = Service Message
WL = Worklist







BT - BLOCK TYPE

1 = BACK BLOCK WITH SAME CONNECTION AND BSN HAS BEEN PROCESSED

2 = BLK BLOCK WITH PORTION BUT NOT END O

7 TERMINATE

3 = MSG BLOCK WITH END OF DATA

4 = CMD COMMAND

5 = BREAK DISCONTINUITY IN DATA OPP DIR

6 = RESET RESET DATA AFTER BREAK

DN
SN

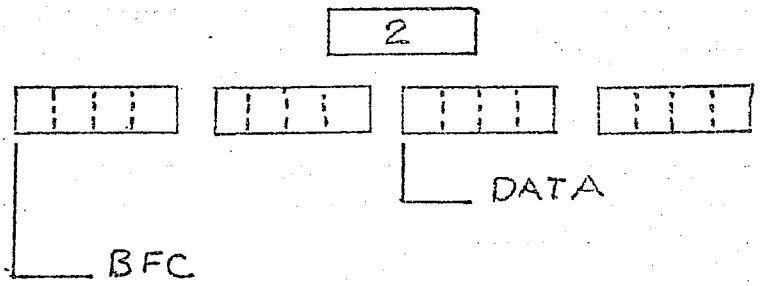
CN
BT=1
BSN

DATA BLOCK TYPE BLOCK OR MSG TYPE 2+

DN
SN

CN
BT=2+3
BSN

BFC
DATA

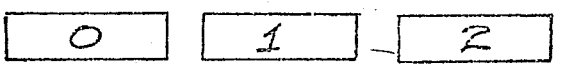


BFC – BLOCK FORMAT CODE EQUAL ZERO FOR ALL TYPES EXCEPT FOR MODE 4 TIP DOWNLINE

MODE-4 0 = CLEAR WRITE
1 = RESET WRITE
2 = WRITE

DATA – DATA BLOCK = 2048 BYTES OF DATA
MSG BLOCK = 2043 OR LESS BYTES OF DATA

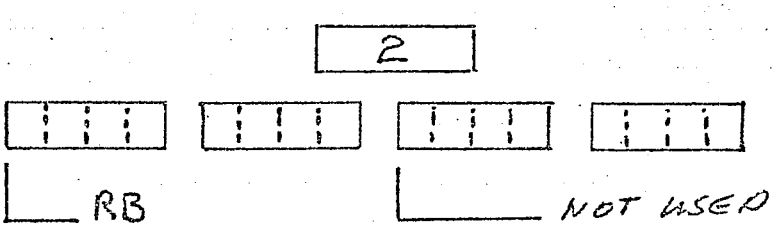
BREAK TYPE BLOCK -5-



DN
SN

CN
BT=5
BSN

RB



RB - REASON FOR BREAK. EQUAL ZERO FOR ALL EXCEPT MODE 4 TIP

MODE-4 BREAK RB

1 = VMNORESP	NO RESPONSE FROM TERMIN
2 = VMBDRESP	BAD RESPONSE FROM TERMIN.
3 = VMERRESP	ERROR RESPONSE FROM TERMINA
4 = VMPRINTFAIL	PRINTER FAILURE
5 = VMBATCHINT	BATCH-INTERRUPT

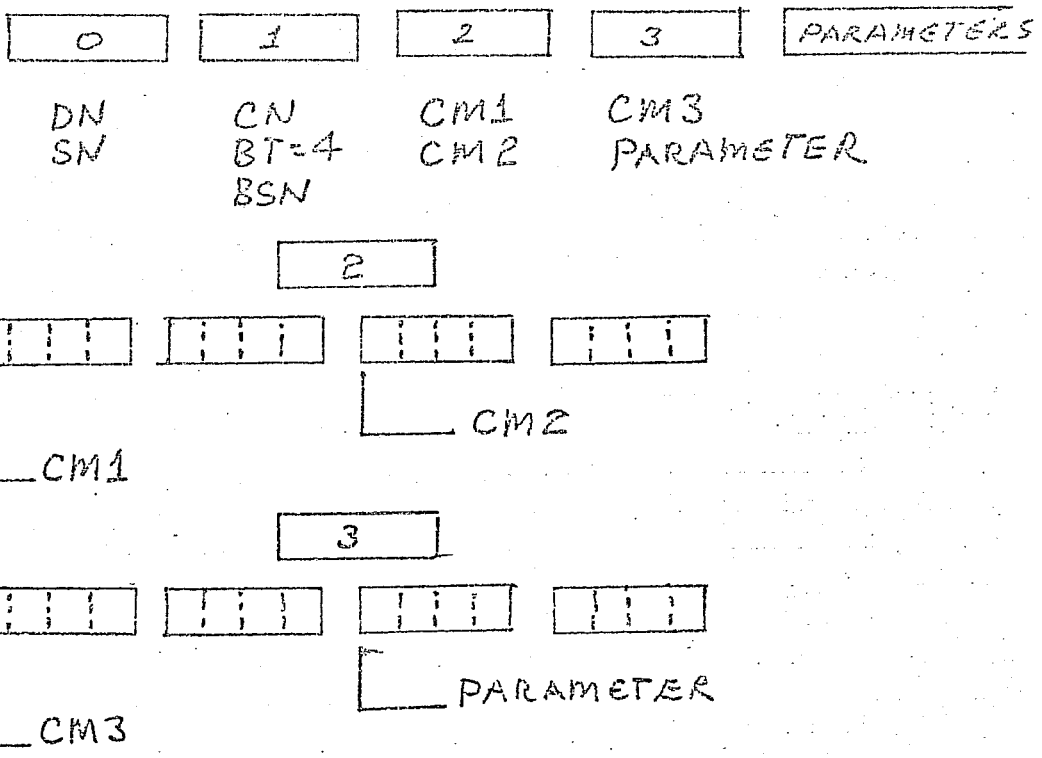
RESET TYPE BLOCK -6-

0	1
DN SN	CN BT = 6 BSN = 0

TERMINATE TYPE BLOCK -7-

0	1
DN SN	CN BT = 7 BSN = 0

COMMAND TYPE BLOCK - TT -

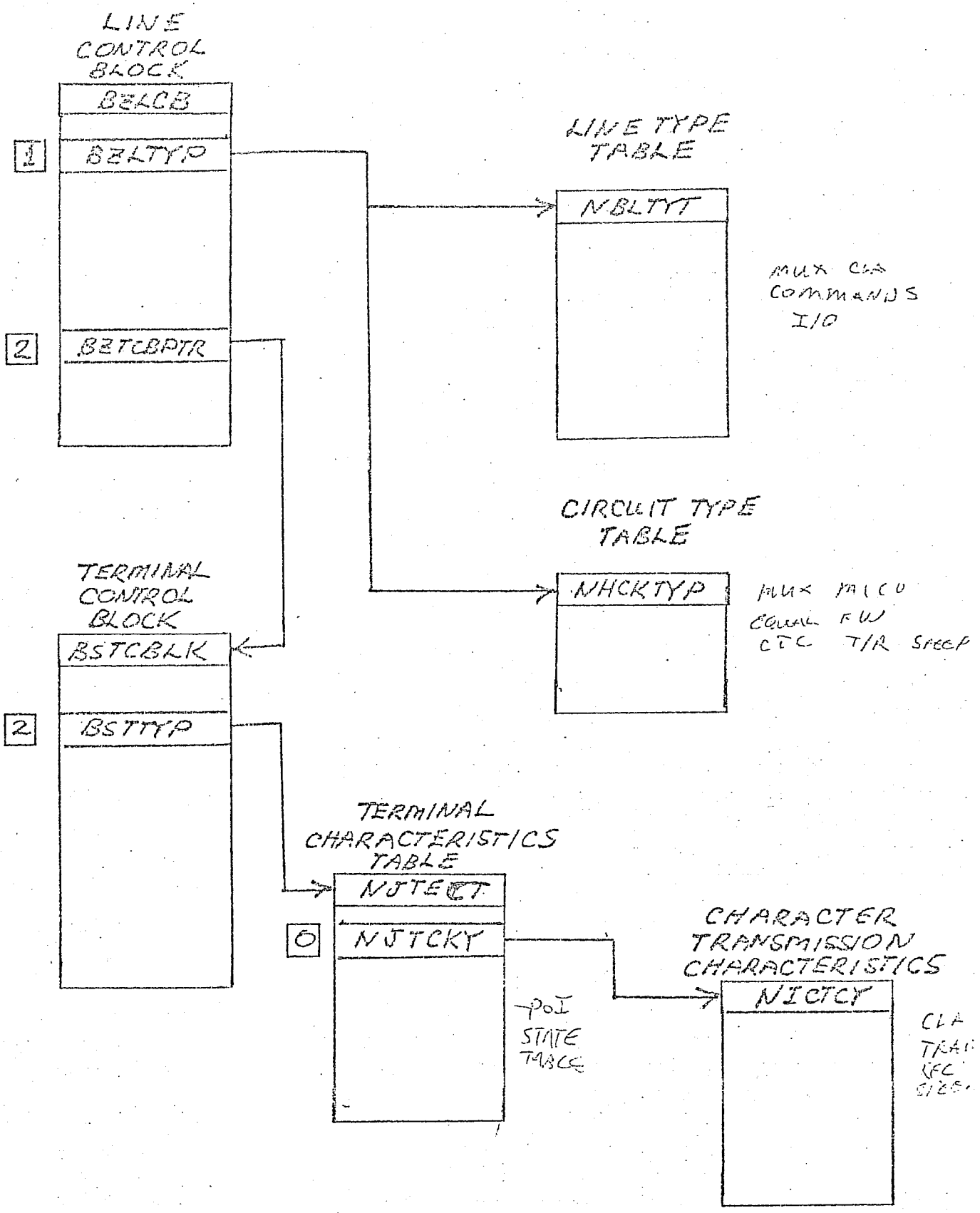


CM1	CM2	CM3	
1	0	0	CONFIGURE LINE MESSAGE
0	1	0	CONFIGURE LINE RESPONSE
1	0	1	CONFIGURE TERMINAL MESSAGE
0	1	1	CONFIGURE TERMINAL RESPONSE
2	0	0	ENABLE LINE MESSAGE
0	3	0	LINE OPERATIONAL
0	3	1	LINE INOPERATIVE
1	MD IB		MODE SET MD=0 INTERACTIVE TT =1 TAPE TT IB=0 INTERACTIVE MODE IB=1 BATCH MODE
3	SW		SET SCREEN WIDTH SW=WIDTH MODE
2			START POLL
0	4	0	REPORT TO CE ERROR FILE
0	4	1	NPU STATISTICS
0	4	2	LINE STATISTICS

COMMAND BLOCK CODES

CM1	CM2	CM3	
3	0	0	PLACE LINE OUT OF SERVICE
3	1	0	PLACE LINE IN SERVICE
3	2	0	START CSA INTERNAL LOOPBACK TEST
3	2	1	START MODEM LOOPBACK TEST
3	2	2	START EXTERNAL LOOPBACK
3	3	0	TERMINATE TEST

TERMINAL DEFINITION



TERMINAL LINE CONTROL BLOCK
—LCB—

0	1	2	3	4	5	6	7
BZLINO	BZTAPEX BZLTYP BZDUM2 BZTIPTYP BZLTIMER	BZSTATE BZTTYP BZCNFST BZTYPE	BZWSTAT	BZWAITMSK	BZERROR	BZRETADDR	BZLINSTA

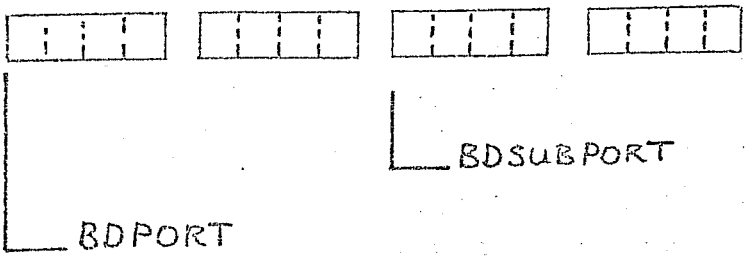
8	9	A	B	C	D	E	F
BZWTCOUNT BZPDMO	BZERACUM	BZRISAVE	BZR2SAVE	BZIFBA	BZITBA	BZOTBA	BZNPMORG BZLOSPEND BZLOS BZDIAG BZABORT BZNDME BZMDCL

10	11	12	13	14	15	16
BZBLKSENT BZBLKREC	BZCHASENT [illegible]	[illegible] BZCHARREC	[illegible] BZTCBPTR	BZLBTOMUX	MODE 4 BZCURTCB TTY BZTYRET	MODE 4 BZDISABLED BZENABLED BZAUTOREC BZMASK BZM4D1 BZDELAYLINE BZTDDISABLED BZMAXRETRY TTY BZTYCNT BZTYRSV

(165) = ENA LINE-0 LCB

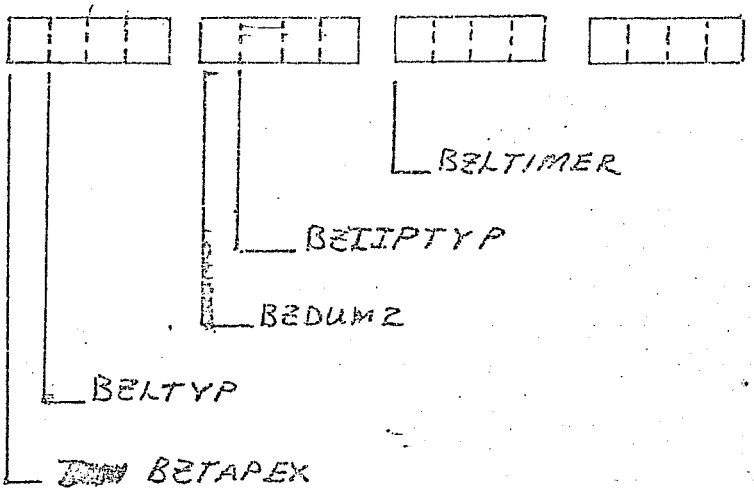
0

BZLINO



BDPORT = CLA ADDRESS PORT NUMBER LIVE NUMBER

BDSUBPORT = SUBPORT NUMBER



BETAPEX - TIMAL APPENDAGE EXISTS NO REF FOR SETTING

BELTYP - LINE TYPE

1 = NOL1	2560-1	DIAL UP	HALF DUPX	CON CARR	
2 = NOL2	2560	DEDICATED	FULL DUPX	CONT CARR	
3 = NOL3	2560	DEDICATED	FULL DUPX	CONST CARR	
4 = NOL4	2561	DIAL UP	FULL DUPX	CONST CARR	
5 = NOL5	2561	DEDICATED	FULL DUPX	CONST CARR	

6 = NOLDIAG FOR ON LINE DIAGNOSTIC

BEDUM2 - NOT USED

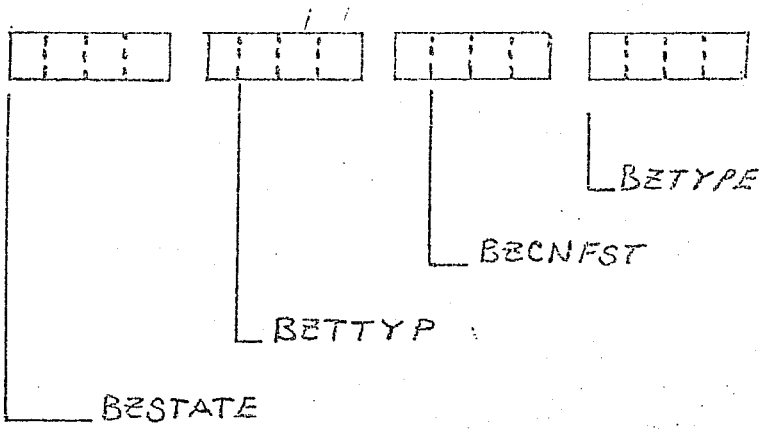
BETIPTYP - TIP TYPE

1 = NOTTY TTY TIP

2 = NOMODE4 MODE 4 TIP

6 = NOOLDIP ON LINE DIAGNOSTIC

BELTIMER - GENERAL PURPOSE TIMER USED IN PIWAIT AMOUNT OF ½ SEC INTERVALS TO WAIT FOR EVENT TO OCCUR



BESTATE – CONFIGURED TERMINAL TYPE FROM MDM AFTER AUTO RECOGNITI

1 = NOT1TTY TTY 110 BAUD

2 = NOT2TTY 150 BAUD

3 = NOT3TTY 300 BAUD

4 = NOT4TTY TTY AUTO RECOGNITION

5 = NOT5MD4 MODE 4A BCD 200 ULT

6 = NOT6MD4 MODE 4A ASCII 734

7 = NOT7MD4 MODE 4C 711-714

8 = NOT8MD4 MODE4 AUTO RECOGNITION

9 = NOT9TTY TTY 600 BAUD

A = NOTATTY 1200 BAUD

F = NOT DIAG DIAG

BETTYP –

SET FROM CONFIGURE LINE COMMAND FROM HOST
SAME FORM AS BESTATE

BECNFST – CURRENT STATE OF CONFIGURATION

0 = C7VOID CONTROL BLOCK VOID

1 = C7DISABLED LINE CONFIGURED BUT NOT ENABLED

2 = C7ENREQ ENABLE REQUESTED TO TIP

3 = C7OPER LINE OPERATIONAL BUT NO TCB

4 = C7TERCNF TERMINAL CONFIGURED

5 = C7ENABLED

6 = C7INOP LINE INOPERATIVE AND NO TIP

7 = C7DCREQ DISCONNECT REQUESTED BY TIP

BETYPE — TYPE OF ACB MODE 4 OR TTY OR DIAGNOSTIC

0 = BESTD TTY OR MODE 4 OR DIAGNOSTIC

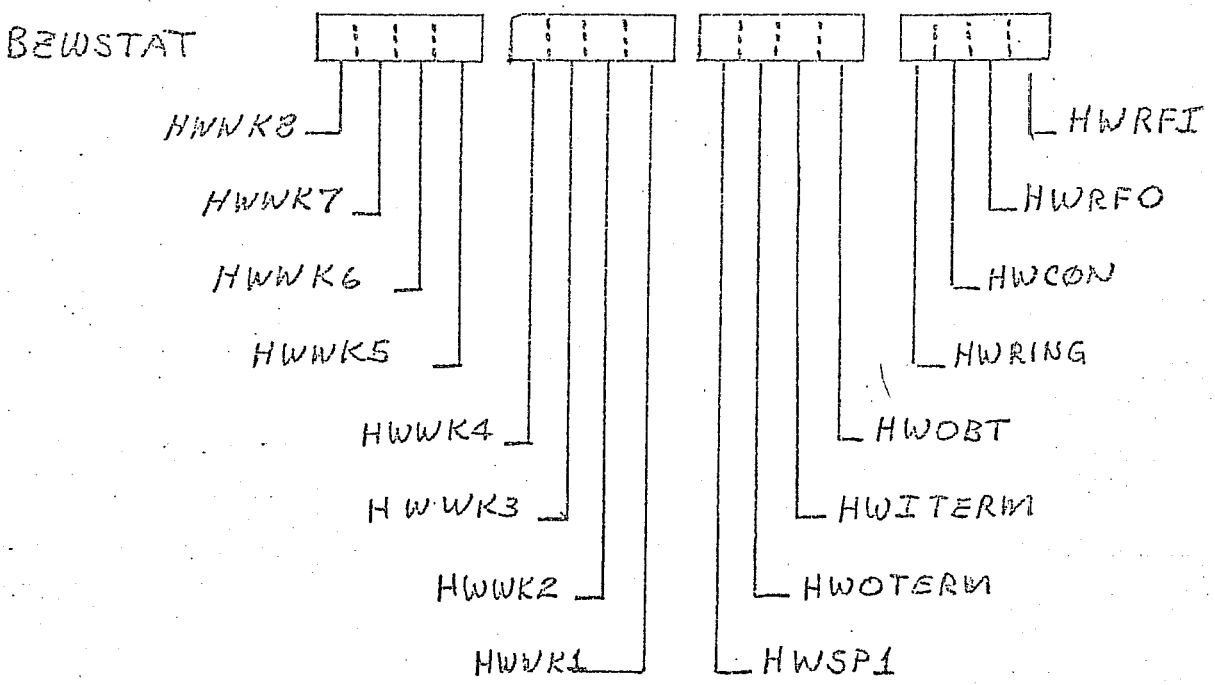
6 = MLIA

7 = CONSOLE

8 = COUPLER

TABLE 3-3. TERMINAL TYPE CODES

Term Type Code	TIP Type	Char. Transmission Characteristics Key	General Description	Specific Terminals Supported
1	TTY	1	10 cps, 110 baud	Teletype M33, M35, and M38 CDC 713-10
2	TTY	2	15 cps, 150 baud	Teletype M37 CDC 713-10
3	TTY	3	30 cps, 300 baud	CDC 713-10
4	TTY	1-3	Automatic recognition of above line speeds	Any teletype compatible terminals specified above
5	MD4	4	Mode 4A BCD	214, 217 (200 UT), 731-12, 732-12, 734-1
6	MD4	4	Mode 4A ASCII	217, 731-12, 732-12, 734-1
7	MD4	4	Mode 4C	711-10, 714-10/20
8	MD4	4	Automatic recognition of above by repeated poll of Controllers 70, 71, and 72	Any Mode 4 terminals specified above, after address strapping which implies terminal type from controller address



BEWSTAT - LINES CARRENT EVENT WAIT STATUS

HWWK8	MODE4-	
	TTY - GHBREAK	ESC/CNTLE ENCOUNTERED
HWWK7	MODE4-VM4EOF	WAIT FOR EOF IN READ
	TTY - GHDLMLF	LINE FEED ENCOUNTERED
HWWK6	MODE4-VM4ERR	WAIT FOR LINE ERROR
	TTY - GHDLMCR	CARRIAGE RETURN ENCOUNTE
HWWK5	MODE4-VM4REJ	WAIT FOR REJ+ACK+ERR
	TTY - GHBLKSZ	DATA BLOCK SIZE EXHAUSTED
HWWK4	MODE4-VM4RE3	WAIT FOR READ E3 CODE
	TTY - GHRCFRS	FIRST INPUT CHARACTER RECEIVED
HWWK3	MODE4-VM4RE2	WAIT FOR READ E2 CODE
	TTY - GHA030	AUTO BAUD AT 30 CPS
HWWK2	MODE4-VM4RE1	WAIT FOR READ E1 CODE
	TTY - GHA015	AUTO BAUD AT 15 CPS
	DIAGNOSTIC - HWSTAT	CAA STATUS RETURNED

HWWK1 TTY - GHADIO AUTO BAUD AT 10 CPS
MODE4 - VM4SOH WAIT FOR SOH

HWSP1

HWOTERM - OUTPUT TERMINATED

HWITERM - INPUT TERMINATED

HWOBT - OUTPUT BUFFER TRANSMITTED

HWRING - RING INDICATOR

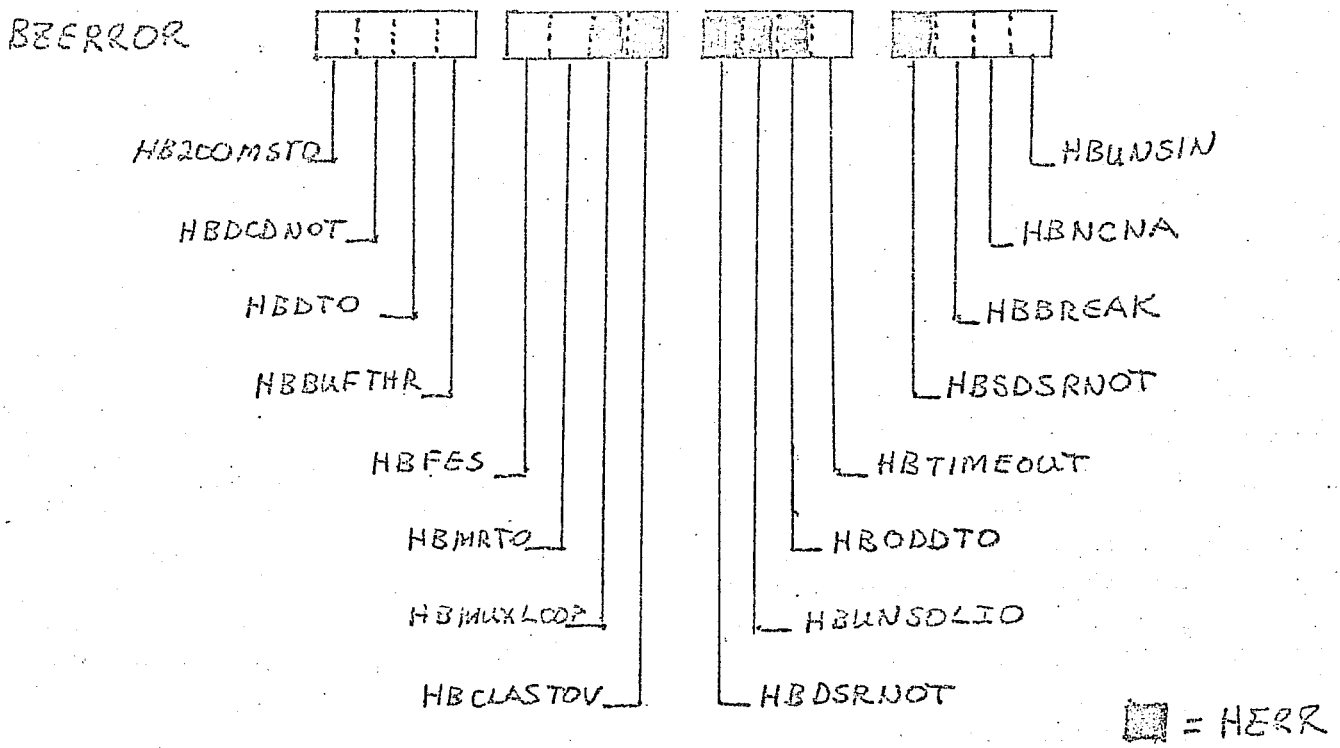
HWCON - CARRIER ON

HWRFO - READY FOR OUTPUT

HWRFI - READY FOR INPUT

BEWAITMASK – EVENT TIP IS WAITING TO OCCURR, SET BEFORE CALLING PTWAIT. SAME FORMAT AS BEUSTAT-WORD 3 LCB.
WHEN EVENT HAPPENS CONTROL IS RETURNED BACK TO TIP

Line Type Code	CLA Type	Modem Type	Circuit Type	Transmission Facility	Controlled Carrier
1	2560-1	RS232-201A Comp.	Switched	Half Duplex	Yes
2	2560-1	RS232-201B Comp.	Dedicated	Full Duplex	Yes
3	2560-1	RS232-201B Comp.	Dedicated	Full Duplex	No
4	2561-1	RS232-103E/113 Comp.	Switched	Full Duplex	No
5	2561-1	RS232-103 Comp.	Dedicated	Full Duplex	No



BBERROR – LINE LOGICAL STATUS DETECTED BY MUX DRIVER DURRING PTWAIT TIME

HB200MSTO – 200 MS TIMEOUT

HBDCDNOT – DATA CARRIER DROPPED

HBDTO – DATA TRANSFER OVERRUN OUTPUT CHAR SENT BEFORE LAST CHAR TRANSMITTED

HBBUFTHR – BUFFER THRESHOLD REACHED

HBFES – FRAMING ERROR STATUS TTY CHAR WITHOUT STOP BIT

HBMRTO – MODEM RESPONSE TIMEOUT

HBMUXLOOP – MUX LOOP ERROR

HBCLASTOV – CLA STATUS OVERFLOW, EXCESS OF 128 HARDERR OCCURRED IN .5 SEC.
HARDERROR = HBMUXLOOP + HBDSRNOT + HBUNSOLIO + HBODDTO + HBSDSRNOT

HBDSRNOT – DSR + CTS NOT READY DEDICATED LINE

HBUNSOLIO – UNSOLICITED I/O (ODD + NCNA)

HBODDTO - ODD TIMEOUT

HBTIMEOUT - WAIT TIMEOUT

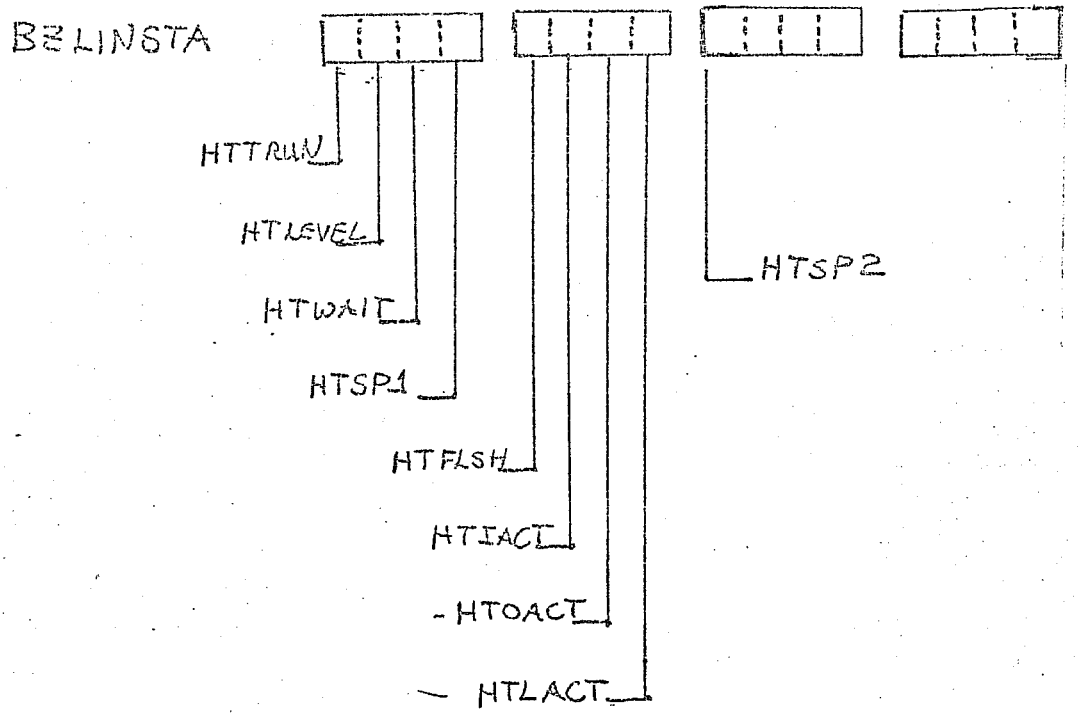
HBDSRNOT - DSR NOT READY DIAL UP LINE

HBBREAK - BREAK CHARACTER RECEIVED

HBNCNA - NEXT CHARACTER NOT AVAILABLE

HBUNSIN - UNSOLICITED INPUT DATA IN CIB

BERETADDR - RETURN ADDRESS IN TIP TO RETURN TO WHEN EVENT SET IN BEWAITMASK OCCURES IN BEWSTAT



HTTRUN – WAIT TIMER ACTIVE WAITING ON EVENT

HTLEVEL – LEVEL 1 = MUX 0 = ØPS WHICH CALLED PTWAIT

HTWAIT – WAIT OUTSTANDING

HTSP1 – NOT USED

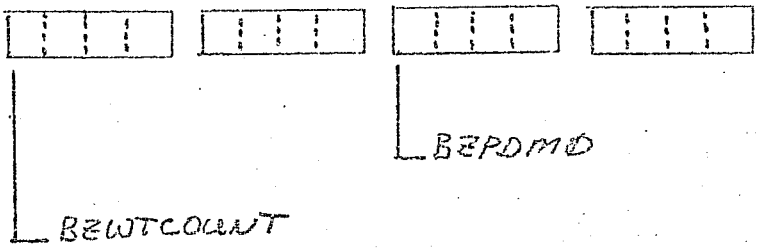
HTFLSH – FLUSH

HTIACT – INPUT ACTIVE

HTOACT – OUTPUT ACTIVE

HTLACT – LINE ACTIVE

HTSP2 – NOT USED



BZWTCOUNT – WAIT COUNT FOR CONTINUE W/L ENTRY

BZPDMD – CLA HARDERR STATUS COUNT, USED TO DETERMINE BZERROR.HBCMASTOV. 128 ERRORS IN .5 SEC DURING PTWAIT SEQUENCE.

SET EQUAL TO ZERO EVERY .5 SEC WHEN ABLCBTMSG IS CALLED

BZERACUM – ACCUMULATED BZERROR DURRING PTWAIT SEQUENCE. SAME FORMAT AS BZERROR LCB WORD-5

A

BZR1SAVE — SAVE R1 REGISTER AT TIME OF PTWAIT CALL

B

BZR2SAVE — SAVE R2 REGISTER AT TIME OF PTWAIT CALL

C

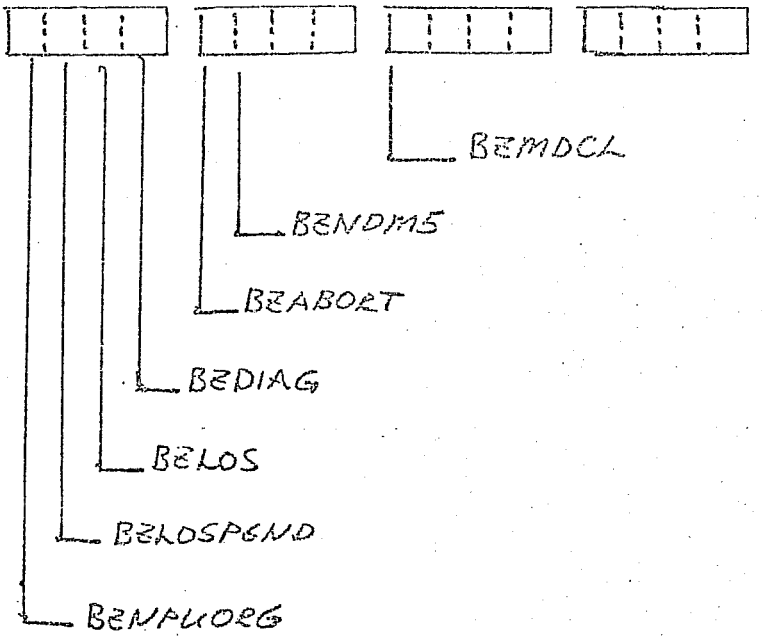
BZIFBA - BUFFER ADDRESS RECEIVED WHEN INPUT BEGAN

D

BZITBA — BUFFER ADDRESS RECEIVED WHEN INPUT ENDED

E

BZOTBA — BUFFER ADDRESS RECEIVED WHEN OUTPUT ENDED



BENPUORG — NPU REQUESTED DISABLE LINE

BELOSPEND — DISABLE LINE BY HOST PENDING

BELOS — LINE OUT OF SERVICE

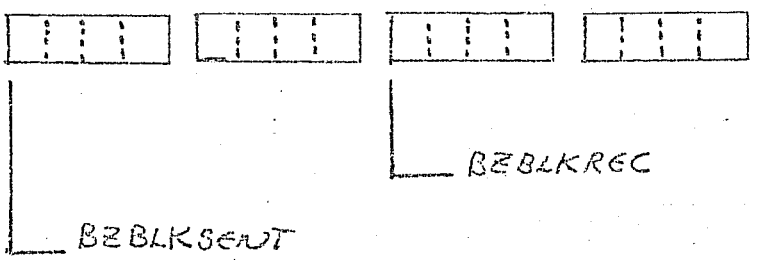
BEDIAG — DIAGNOSTIC IN PROGRESS

BEABORT — TERMINATE DIAGNOSTIC

BENDM5 — NOT USED

BEMDCL — MODEM CLASS USED WITH DIAGNOSTICS

10



BZBLKSENT — BLOCKS SENT

BZBLKREC — BLOCKS RECEIVED

11

BZCHRSENT — CHARACTERS SENT

12

BZCHRREC — CHARACTERS RECEIVED

13

BZTCBPTR — FIRST TCB TERMINAL CONTROL BLOCK ATTACHED TO THIS LCB

14

BZLBTØMUX — LAST BUFFER GIVEN TO MUX DRIVER TO OUTPUT

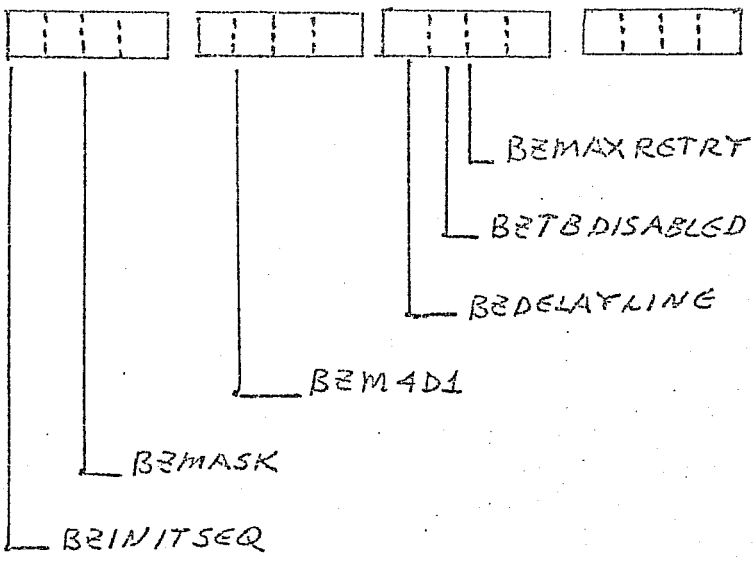
15	MODE4 LINE

BECURTCB — CURRENT TCB TERMINAL CONTROL BUFFER BEING SERVICED

15	TTY LINE

BETYRET — RETURN ADDRESS FROM SUB CALL

16 MODE4 LINE



BEINITSEQ — LINE INITIALIZATION SEQUENCE

0 = BEDISABLED LINE DISABLED

1 = BEENABLED ENABLE LINE

2 = BEAUTOREC AUTO RECONITION

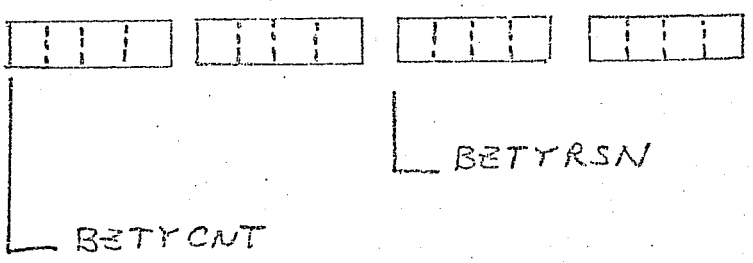
BEMASK— AUTO RECOGNITION MASK

BEM4D1 -

BEDELAYLINE — DELAY LINE SERVICE NEGATIVE EVENT TASK SENT BY TIP. SERVICE NEXT LINE

BETBDISABLED — DISABLE LINE RECEIVED

BEMAXRETRT — MAX RETRY FOR TIP TASK



BETYCNT — MAX NUMBER OF RETRIES

BETYRSN — REASON TO BRING LINE DOWN

0 = INOP

1 = TCB

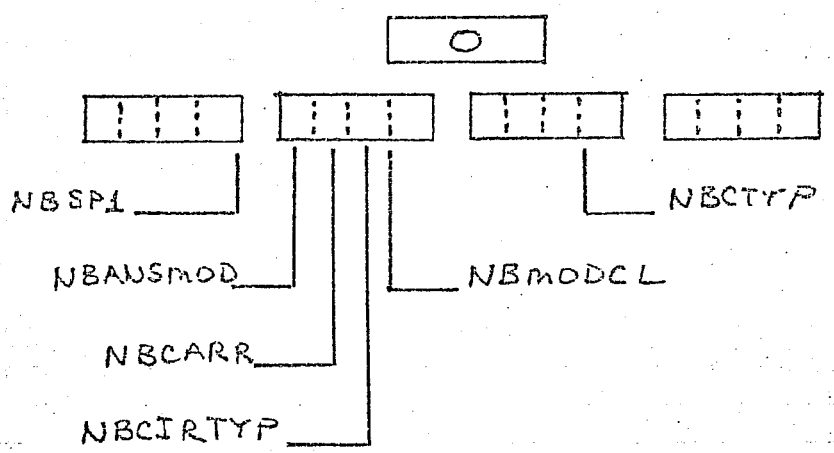
2 = DIS

LINE TYPE TABLE NBLTYT

0	1	2	3	4	5	6	7
NBSP1 NBANSMOD NBCARR NBCIRTY NBMODCL NBCTYP	NBSP2	NBSP3	NBSP4	NB3TIME NB3SEND NB3SEQT	NB4TIME NB4SEND NB4SEQT	NB5TIME NB5SEND NB5SEQT	NB6TIME NB6SEND NB6SEQT

BELTYP 1
1 - 6

8	9	A	B
NB8TIME NB8SEND NB8SEQT	NBATIME NBASEND NBASEQT	NBBTIME NBBSEND NBBSEQT	NBCTIME NBCSEND NBCSEQT



NBSP1 - - NOT USED

NBANSMOD - ANSWER MODE 0 = AUTO 1 = DEDICATED

NBCARR - CARRIER 0 = CONSTANT 1 = CONTROLLED

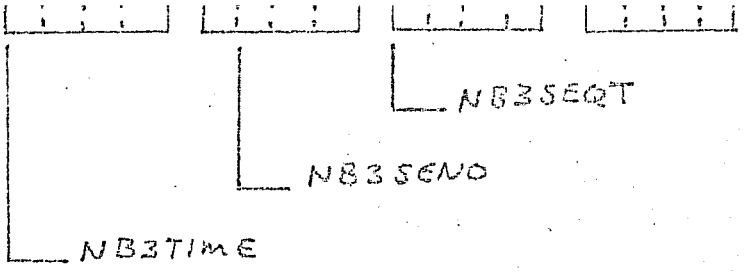
NBCIRTYP - CIRCUIT TYPE 0 = 2 WIRE 1 = 4 WIRE

NBMODCL - MODEM CLASS

NBCTYP - CLA TYPE 0 = 2560 (SYNC) 1 = 2561 (ASYNC)

1	-	3

NBSP2
NBSP3 - NOT USED
NBSP4



NB3TIME — TIMER FOR NCTIME IN MUX LCB IN MUX DRIVER NKTURN

NB3SEND — NUMBER OF CLA OUTPUT CELLS NEEDED FOR NKTURN

NB3SEQT — MUX MODEM COMMAND SEQUENCE KEY FOR NKTURN

| 5 | — | 8 |

SEE pg 60

SAME FORMAT AS WORD 4

| 5 |

NB4TIME — NKINIL TIMER FOR NCTIME IN MUX LCB

NB4SEND — NKINIL NUMBER OF CLA OUTPUT CELL NEEDE

NB4SEQT — NKINIL MUX MODEM SEQ COMMAND KEY

| 6 |

NB5TIME — NKENBL TIMER

NB5SEND — NKENBL CLA OUTPUT CELL COUNT

NB5SEQT — NKENBL MUX MODEM SEQ COMMAND KEY

| 7 |

NB6TIME — NKINPT TIMER

NB6SEND — NKINPT CLA OUTPUT CELL COUNT

NB6SEQT — NKINPT MUX COMMAND SEQ KEY

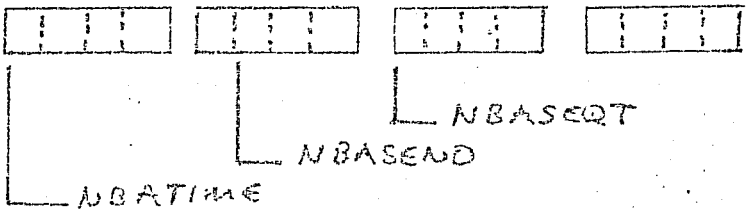
| 8 |

NB8TIME — NKOUTPUT - NKQUOUT TIMER

NB8SEND — NKOUTPUT - NKQUOUT CLA OUTPUT CELL COUNT

NB8SEQT — NKOUTPUT - NKQUOUT MUX COMMAND SEQ KEY

9



NBATIME – NKENDIN TIMER FOR NCTIME IN MUX LCB

NBASEND – NKGNDIN CLA OUTPUT CELL COUNT

NBASEQT – NKENDIN MUX COMMAND SEQ KEY

A

NBBTIME – NKENDOUT TIMER FOR NCTIME IN MUX LCB

NBBSEND – NKGNDOUT CLA OUTPUT CELL COUNT

NBBSEQT – NKENDOUT MUX COMMAND SEQ KEY

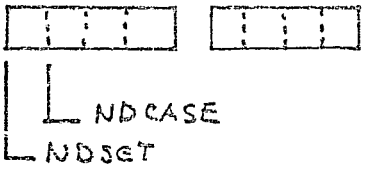
B

NBCTIME – NKDISL TIMER FOR NCTIME IN MUX LCB

NBBSEND – NKDISL CLA OUTPUT CELL COUNT

NBBSEQT – NKDISL MUX COMMAND SEQ KEY

MODEM COMMAND SEQUENCE

NB3SEQT 

NDSET — SET CONDITION SPECIFIED BY NOCASE IF SET
CLEAR CONDITION SPECIFIED BY NOCASE IF ZERO

NDCASE — CLA COMMAND BIT TO CHANGE IN PROCEDURE PMPRAS

1 = RTS REQUEST TO SEND

2 = SRTS SECONDARY REQUEST TO SEND

3 = ØM ORIGINATE MODE

4 = LM MODEM LOOPBACK

5 = DTR DATA TERMINAL READY

6 = TB TERMINAL BUSY

7 = ION INPUT ON

8 = OON OUTPUT ON

9 = BREAK BREAK MODE

A = ISR INPUT SUPERVISION REPORT

B = ISON INPUT SUPERVISION ON

C = DLM DATA LINE MONITOR CHECK END OF BREAK

D = PARITY SET ODD PARITY WHEN ON

E = PARITY ENABLE

F = ECHO SET ECHO MODE

10 = LBT INTERNAL LOOP BACK

11 = RSYN RESYNCHRONIZE INPUT

12 = HALF DUPLEX IF INPUT AFTER OUTPUT CLEAR ION

MODEM COMMAND SEQUENCE TABLE
1434

INDEX		
	02	$\overline{RTS}$
	03	RSYN
	04	IF ION AFTER OON THEN $\overline{ION}$
	05	$\overline{OON}$
	08	$\overline{DTR}$
	09	$\overline{RTS}$
	0C	ISR — INITIALIZE LINE
	0D	ISON
	10	RTS
	11	ISR
	12	DTR
	16	RTS
	17	OON
	18	$\overline{ION}$
	1C	DTR
	1D	RTS
	20	$\overline{OON}$
	21	$\overline{RTS}$
	24	RSYN
	25	IF ION AFTER OON THEN ION
	28	OON
	2C	$\overline{DTR}$
	30	RTS
	31	ISR
	34	DTR
	35	ISON

38 $\overline{ISON}$ CWA MOD

DATA FROM LINE TYPE TABLE AND MODEM COMMAND SEQUENCE TABLE

LINE-TYPE	2	4	5
INITIALIZE LINE CMD	ISR ISØN	ISØN	ISR ISON
EXIT INITIALIZE CMD STATUS TBL	ISØN $\overline{\text{ISR}}$ $\overline{\text{RSYN}}$	ISØN	ISØN $\overline{\text{ISR}}$ $\overline{\text{DLM}}$
ENABLE LINE CMD	DTR ISR ISØN	RTS DTR ISØN	RTS DTR ISR ISØN
EXIT ENABLE CMD STATUS TBL	DTR ISØN $\overline{\text{ISR}}$ $\overline{\text{RSYN}}$	RTS DTR ISØN $\overline{\text{ISR}}$ $\overline{\text{DLM}}$	RTS DTR ISØN $\overline{\text{ISR}}$ $\overline{\text{DLM}}$
TERMINATE INPUT CMD	$\overline{\text{IØN}}$	$\overline{\text{IØN}}$	$\overline{\text{IØN}}$
TERMINATE OUTPUT CMD	$\overline{\text{ØØN}}$	$\overline{\text{ØØN}}$	$\overline{\text{ØØN}}$
DISABLE LINE CMD	ISØN $\overline{\text{DTR}}$ $\overline{\text{RTS}}$	ISØN $\overline{\text{DTR}}$ $\overline{\text{RTS}}$	ISON $\overline{\text{DTR}}$ $\overline{\text{RTS}}$
CLEAR LINE CMD	$\overline{\text{ISON}}$ $\overline{\text{DTR}}$ $\overline{\text{RTS}}$	$\overline{\text{ISON}}$ $\overline{\text{DTR}}$ $\overline{\text{RTS}}$	$\overline{\text{ISON}}$ $\overline{\text{DTR}}$ $\overline{\text{RTS}}$

TCB

0	1	2	3	4	5	6	7
BSCHAIN	BSLCBP	BSTPN BSTTYP BSBATCH BSHDQUED	BSOBLKL BSSPI	BSQPTR	BSLDEKY BSXMOD BSOPMOD BSPARY BSTCBSIZ BSQPTYP	BSRQPTR	BSQQPTR

8	9	A	B	C	D	E	F
BSBTPROC BSBSNDWU BSBSNUP BST8TERM BSACPOUT BSACPIN BSRQOVF	BS DN BS CN	MODE4 BSBLKSEUT TTY BSTYOFRS	MODE4 BSBLK REC TTY BSTYRTA	MODE4 BSRETRANS TTY BSTYRGL BSTYOP BSTYXON BSTYLF BSTYIST	MODE4 BSRECERR	MODE4 BSBREAK	MODE4 BSCLUSADD BSTERMAD

10	11	12	13	14	15	16	17
MODE4 BSM4D4 BSERRCNT BSEXITSK	MODE4 BSRDECODE BSERR2 BSTOGREC BSEXT2	MODE4 BSWRECODE BSAUTOREC BSM4D5 BSEXT3	MODE4 BSBLANK BSSCREEN	MODE4 BSBLNOTBK BSPFECODE BSTASK BSWRE3Q BSWAITCRD BSTOGKNOW BSTOGSTAT BSTOGWAS BSTOGLOST BSTOGDET BSTOGCORR	MODE4 BSCLSPTR	MODE4 BSM4POLL POLL	MODE4 POLL

18	19	1A	1B	1C	1D
MODE4	MODE4	MODE4	MODE4	MODE4	MODE4

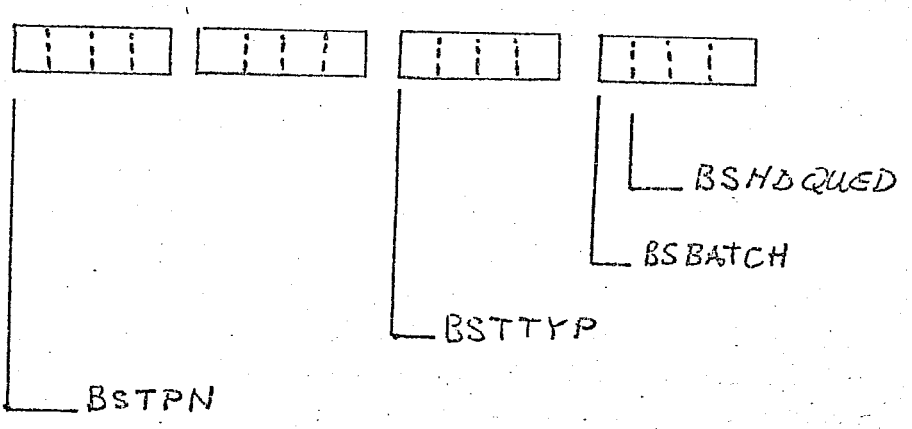
0

BSCHAIN – CHAIN POINTER TO NEXT TCB ON LINE OR ZERO IF END OF CHAIN

1

BSLCBP – POINTER TO FWA LCB LINE CONTROL BUFFER WHICH IS ASSOCIATED WITH THIS TCB

2



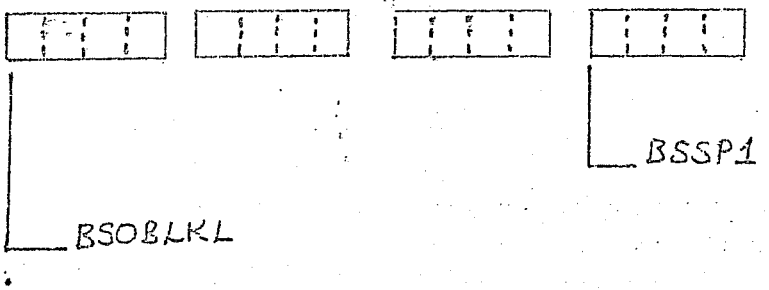
BSTPN – TERMINAL POSITION NUMBER NOT USED

BSTTY – TERMINAL TYPE AS DETERMINED ON CONFIGURE TERMINAL COMMAND. SAME FORMAT AS BZTTYP IN LINE CONTROL BUFFER LCB

BSBATCH – TERMINAL IN BATCH MODE

BSHDQUED – HOST DOWN MESSAGE QUEUED FOR THIS TERMINAL

3

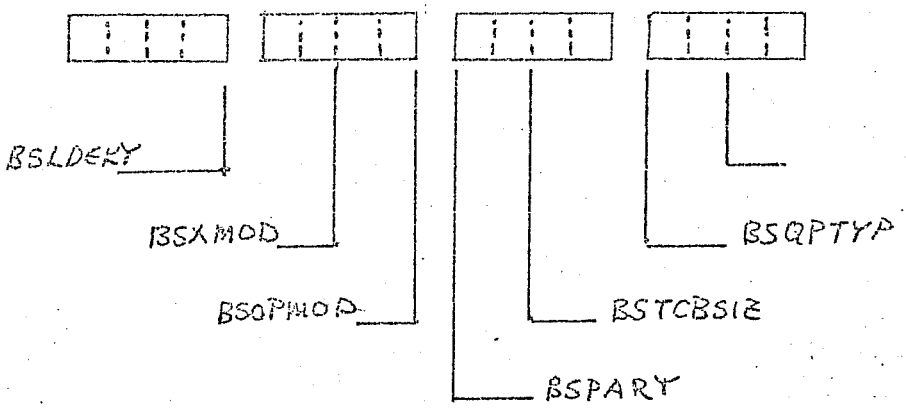


BSOBLKL - OUTPUT BLOCK SIZE IN CHARACTERS

BSSP1 -

4

BSQPTR - OUTPUT QUEUE POINTER



BSLDEKY — LINE DELAY NOT USED

BSXMOD — TRANSMISSION MODE

0 - NOFDX FULL DUPLEX

1 - NO HDX HALF DUPLEX

2 - NOEPX

3 - NOSPX

BSOPMOD — TELETYPE OPERATING MODE

0 - NOKEY

1 - NOTAPE

2 - NOTTT

3 - NOEEE

BSPARY — PARITY TYPE 00 - ODD PARITY
01 - NO PARITY
10 - EVEN PARITY

BSTCBSIZ — BUFFER SIZE OF THIS TCB 1 = 1D TTY - BOS1
2 = 3D MODE4 - BOS2

BSQPTYP - TYPE OF QUEUE POINTER IN THIS TCB

0 = BOQCBLOCK QUEUE CONTROL BLOCK

1 = BOQNEXTSEG NEXT SEGMENT TO GET

2 = BOQDATLIST DATA LIST QUEUE

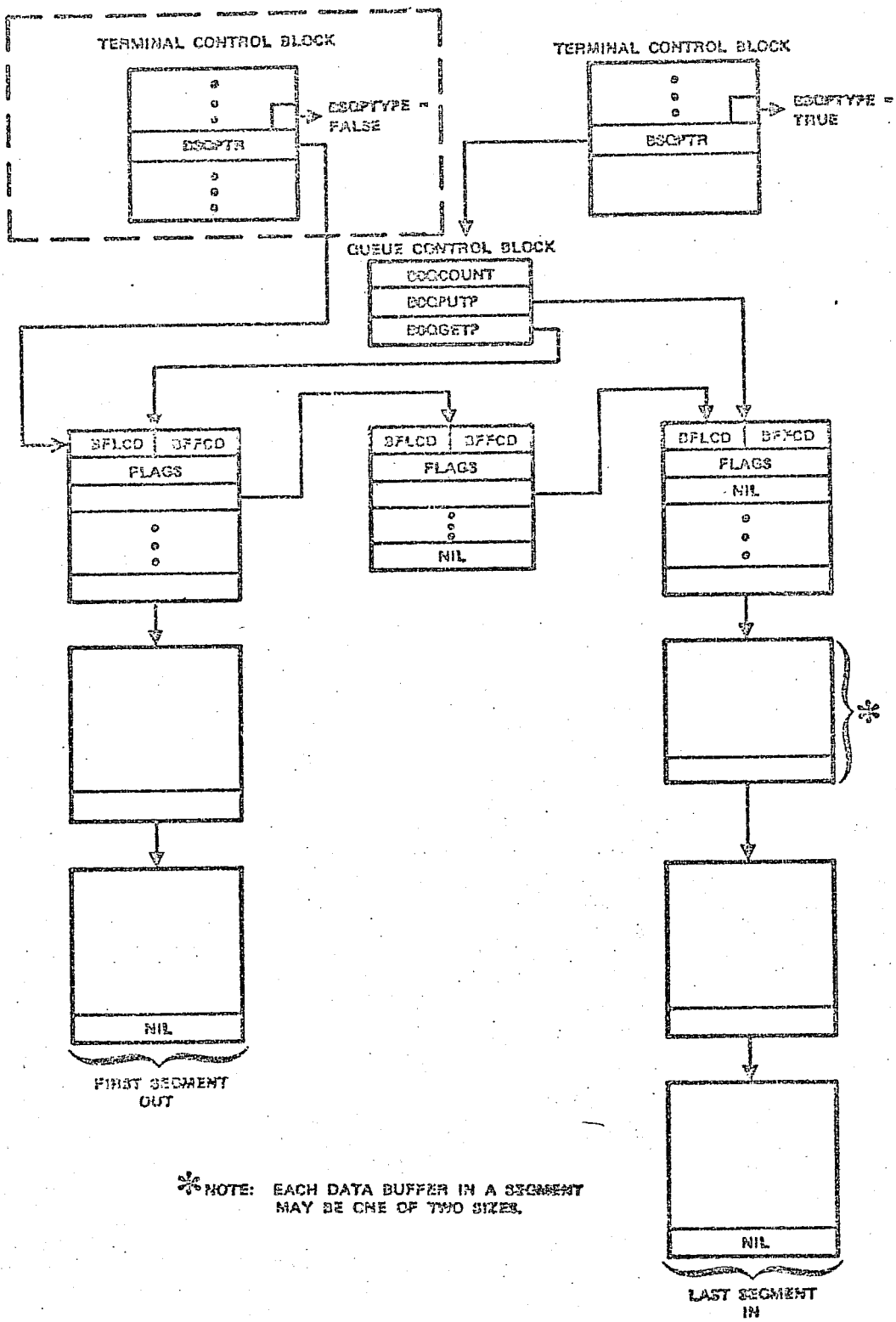


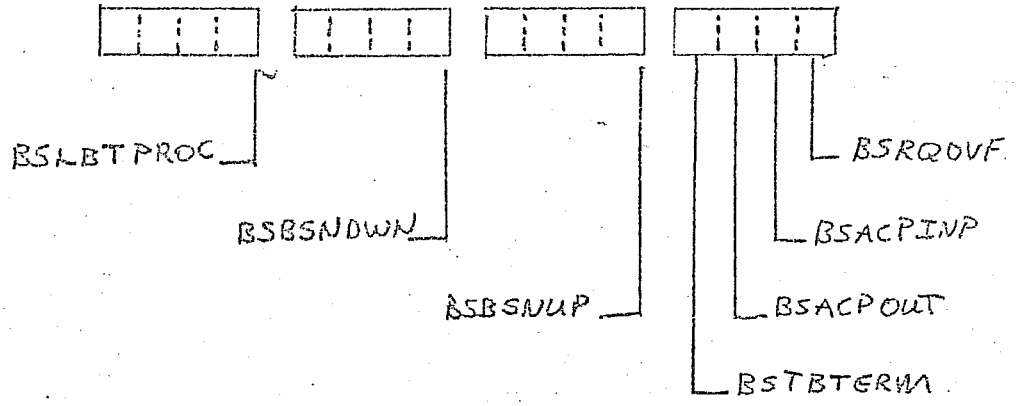
Figure 4-7. Structure of a Queue

6

BSRQPTR – SOURCE RETENTION QUEUE PTR
INPUT DATA FOR HOST FROM TERMINAL

7

BSRQOPTR – SOURCE RETENTION OVERFLOW POINTER



BSLBTPROC – LAST BLOCK TYPE PROCESSED BY TIP

0 = HTSVM INTERNAL SERVICE MESSAGE

1 = HTBACK BACK ACKNOWLEDGEMENT

2 = HTBLK DATA BLOCK FIRST BLOCK UP TO LAST

3 = HTMSG SHORT DATA BLOCK OR END OF HTBLK

4 = HTCMD COMMAND TYPE BLOCK

5 = HTBREAK BREAK

6 = [redacted] HTRESET RESET BREAK

BSBSNDWN – LAST BLOCK SERIAL NUMBER PROCESSED BY TIP

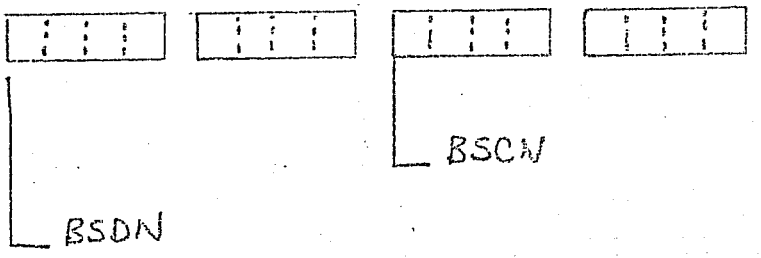
BSBSNUP – NEXT BLOCK SERIAL NUMBER TO SEND TO HOST

BSTBTERM – LOGICAL CONNECTION TO BE TERMINATED

BSACPOUT – ACCEPT OUTPUT

BSACPIN – ACCEPT INPUT

BSRQOVF – SOURCE RETENTION QUEUE CURRENTLY OVERFLOWED



BSDN — BLOCK DESTINATION NODE

BSCN — BLOCK CONNECTION NUMBER

A MODE4 TIP

BSBLKSENT — BLOCKS SENT

A TTY TIP

BSTYØFRS — FIRST OUTPUT BUFFER

B MODE4 TIP

BSBLKREC — BLOCKS RECEIVED

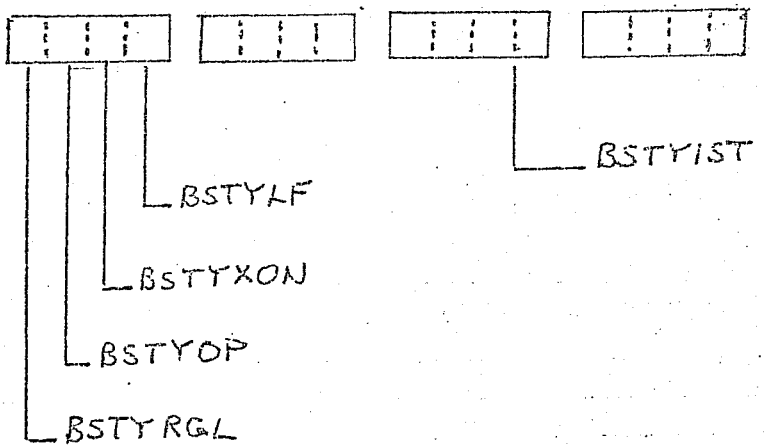
B TTY TIP

BSTYRTA — RETURN ADDRESS SAVE AREA

C MODE4 TIP

BSRETRANS — BLOCKS RETRANSMITTED

C TTY TIP



BSTYRGL — REGULATION IS ON

BSTYOP — DELIVER OUTPUT

BSTYXON — SEND XON TO TERMINAL

BSTYLF — SUPPRESS 1 LFI ON OTPT

BSTYIST — INPUT STATE

2 = KEYBOARD

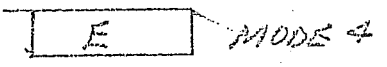
4 = CLEARSTATE

6 = PAPERTAPE

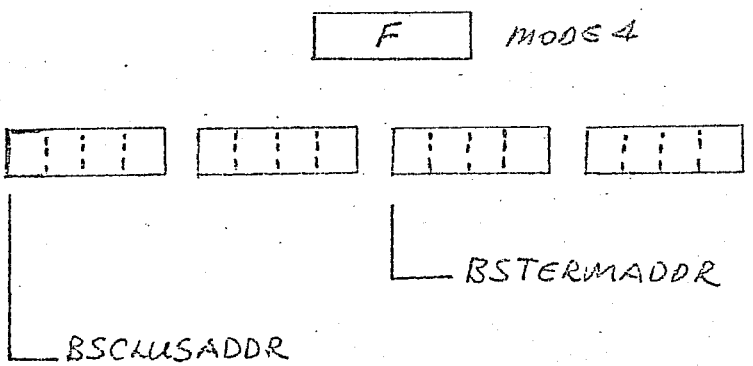
8 = BRKSTAT

12 = QIFCD

BSRECERR — BLOCKS NOT ACCEPTED DUE TO ERRORS

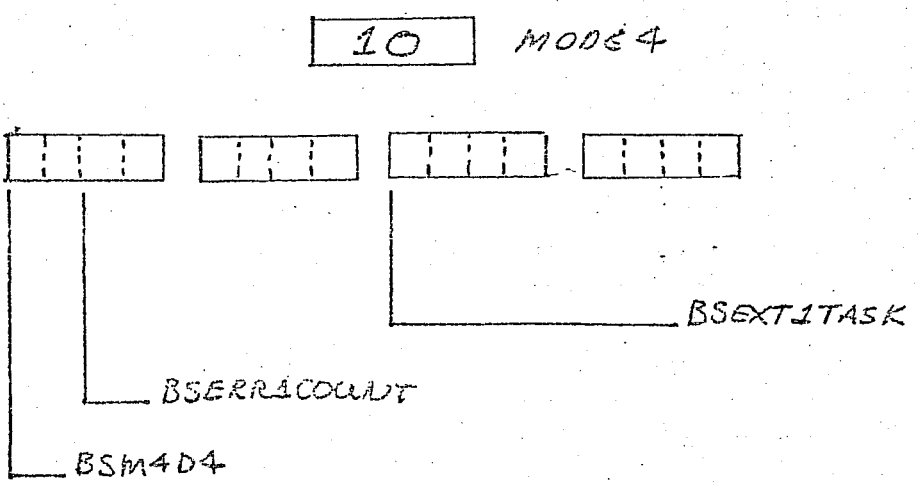


BSBREAK — UPLINE BREAKS DUE TO ERROR OF INCORRECT RESPONSE FROM TERMINAL



BSCLUSADDR — CLUSTER ADDRESS STATION

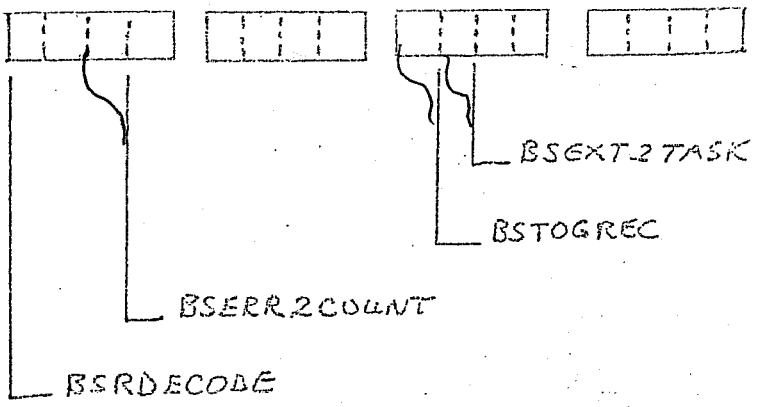
BSTERMADDR — TERMINAL ADDRESS DEVICE



BSM4D4 —

BSERRICOUNT — NO RESPONSE COUNT FROM TERMINAL FOR TASK IN BSTASK WORD 14 TCB

BSEXTITAST — NEXT TASK TO PREFORM WHEN CORRECT RESPONSE RECEIVED FROM TERMINAL, SAME FORMAT BSTASK WD-



BSRDECODE — LAST E-CODE RECEIVED FROM TERMINAL

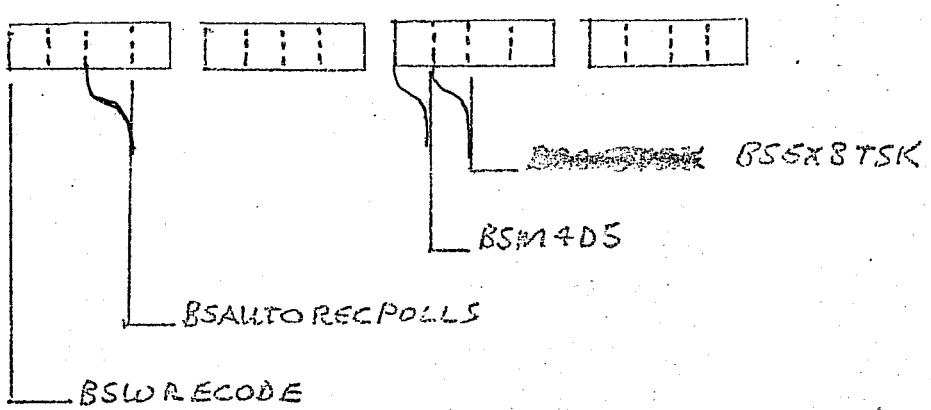
0 = E1 1 = E2 2 = E3 3 = E4

BSERR2COUNT — BAD ERROR-REJECT RESPONSE FROM TERMINAL COUNT. FOR TASK IN BSTASK WD-14

BSTOGREC — LAST ADDRESS TOGGLE BIT RECEIVED

BSEXT2TASK — NEXT TASK TO PREFORM AFTER REJECT RECEIVED FOR RESPONSE FROM TERMINAL FOR TASK IN BSTASK WORD 14. SAME FORMAT AS BSTASK MAY BE NEGATIVE NUMBER

12 MODE4



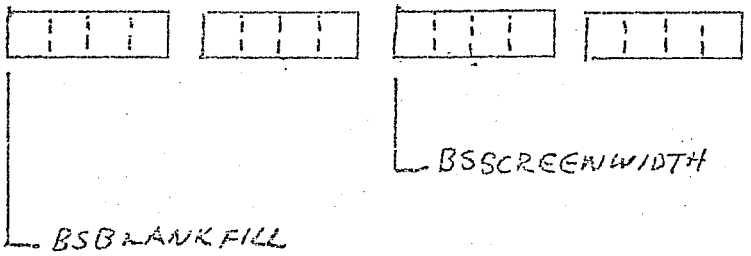
BSWRECODE — LAST E-CODE TRANSMITTED

BSAUTORECPOLLS — AUTO RECOGNITION POLL COUNT

BSM4D5 —

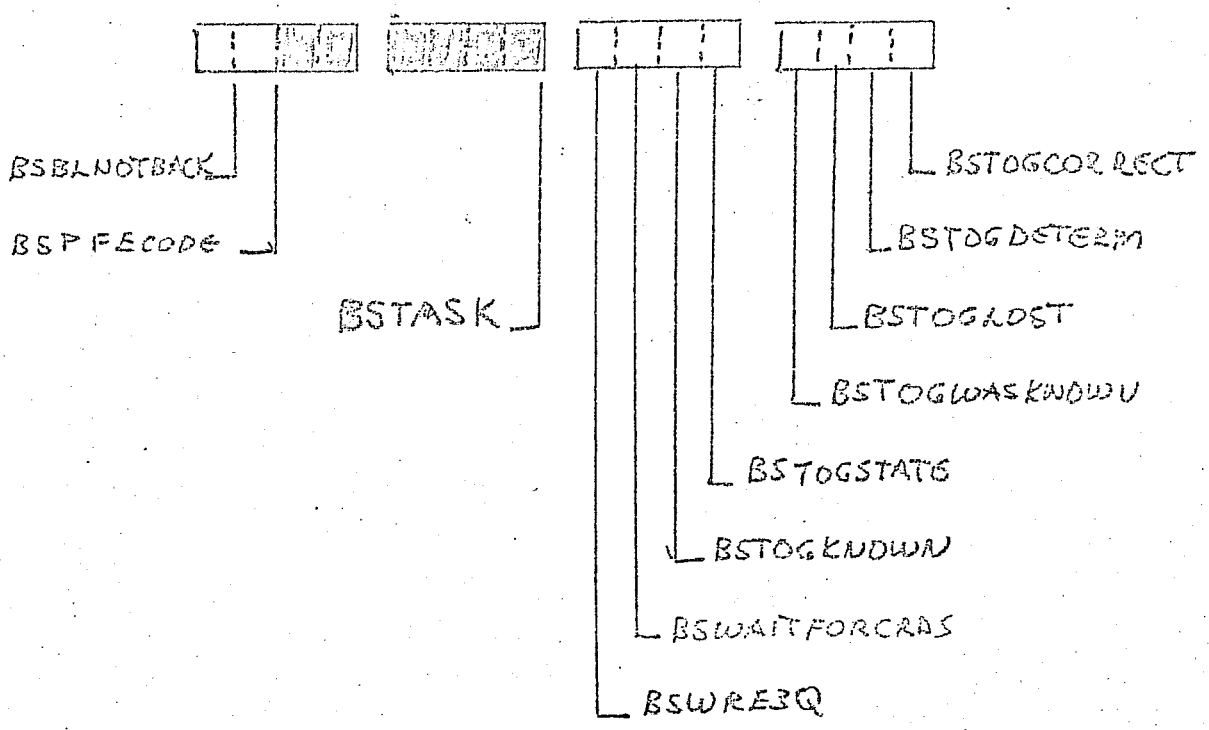
BSEXT3TASK — NEXT TASK TO PREFORM AFTER NO RESPONSE FROM TERMINAL. SAME FORMAT BSTASK WITH TCB. MAY BE NEG

13 MODE4



BSBLANKFILL — NUMBER OF BLANKFILL CHARACTERS +1

BSSCREENWIDTH — SCREENWIDTH IN CHARACTERS



BSBLNOTBACK — BLOCKS NOT BACKED

BSPFECODE - ESCAPE TO POLL E-CODE

BSTASK — CURRENT TASK TO PREFORM FOR THIS TERMINAL

0 = VTZERO FIRST TASK NEW TCB

1 = VTRSER RESET ERROR COUNTERS

2 = VTCHOQ CHECK OUTPUT QUEUE

3 = VTPSPR ISSUE POLL FOR PRINTER STATUS

4 = VTPSNO NO RESPONSE FROM PRINTER

5 = VTPSRE READ RESPONSE FROM PRINTER POLL

6 = VTPSE3 READ E3 RESPONSE FROM PRINTER POL

7 = VTWRPL WRITE OR POLL CARDS

8 = VTWRIT ISSUE WRITE MSG

9 = VTWRBD BAD RESPONSE ON WRITE

A = VTWRER ERROR RECEIVED ON WRITE

B = VTWRNO TOGGLE CANNOT BE OBTAINED

C = VTWRAK ACK RECEIVED ON WRITE

D = VTPCRE — ISSUE POLL FOR CARDS

E = VTPCRE — READ RECEIVED ON POLL FOR CARDS

F = VTPDSP — ISSUE POLL FOR DISPLAY

10 = VTPDRE — READ RECEIVED ON POLL DISPLAY

11 = VTPDRJ — REJECT RECEIVED ON POLL DISPLAY

12 = VTWNOR — NO RESPONSE ON WRITE

13 = VTPECD — POLL FOR E-CODE

14 = VTPERE — READ RECEIVED ON POLL FOR E CO

15 = VTPEND — NO RESPONSE ON POLL FOR E CODE

16 = VTPEBD — BAD RESPONSE ON POLL FOR E CODE

17 = VTPEER — ERROR RESPONSE ON POLL E CODE

18 = VTPERJ — REJECT RESPONSE ON POLL E CODE

19 = VTPTOG — POLL FOR TOGGLE

1A = VTPTNO — NO RESPONSE ON POLL FOR TOGGLE

1B = VTPBD — BAD RESPONSE ON POLL FOR TOGGLE

1C = VTPTDT — DATA RESPONSE ON POLL FOR TOG

1D = VTARPL — AUTO RECOGNITION POLL SENT

1E = VTAREC — AUTO RECOGNITION PREFORMED

1F = VTARNO — AUTO RECOGNITION FAILED

BSWRE3Q — WRITE E3 QUEUED

BSWAITFORCARDS — WAITTING FOR CARDS

BSTOGKNOWN — ADDRESS TOGGLE IS KNOWN

BSTOGSTATE — EXPECTED ADDRESS TOGGLE BIT

BSTOGWASKNOWN — TOGGLE BIT WAS KNOWN

BSTOGLOST — LAST TOGGLE WAS INCORRECT

BSTOGDETERM — DETERMINE TOGGLE

BSTOGCORRECT — LAST TOGGLE CORRECT

15	MODE 4

BSCLSPTR — CLUSTER TCB. POINTER TO FIRST TCB IN MULTISTATION TCB LINE

16	——	1D	MODE 4

BSM4POLL — OUTPUT BUFFER WITH POLL FOR THIS TERMINAL

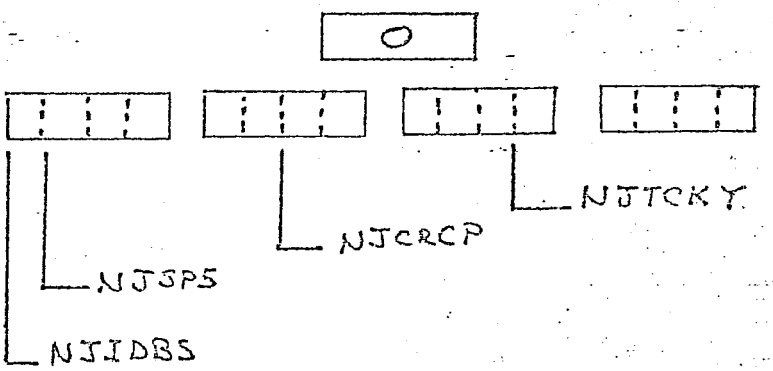
16	LCD | FCD
17	EØF
18	SYNC | SYNC
19	SYNC | SYNC
1A	SOH 01 | STATION ADDRESS
1B	TERMINAL ADDRESS | [redacted] POLL 05
1C	[redacted] ETX / MPC 03
1D	[redacted]

TERMINAL CHARACTERISTICS TABLE

0	1	2	3	4	5	6	7
NJIDB NJSP5 NJCRCP NJTCKY	NJISPTP	NJCNT NJIBF	NJSP1 NJBLKK	NJCXL	NJPOIKEY	NJPOIKEY	NJSP3 NJXMOD NJTIPTY NJOPMOD NJCODKY

8	9	A	B
NJLDEKY NJTCBSI NJBREAKWL NJTERMWL NJFOQWL IOQWL NJSP6	NJTCBBSUB	NJWDID	NJWDII

See p 7



NJIDBS – INPUT BUFFER SIZE 0 = SMALL 1 = LARGE

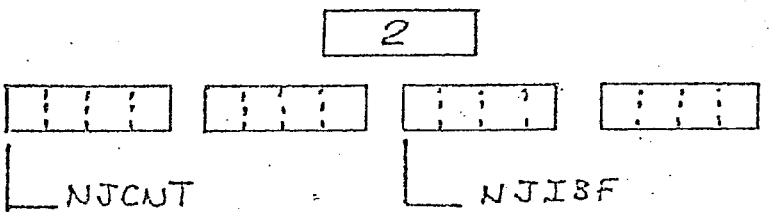
NJSP5 – NOT USED

NJCRCP – CRC POLYNOMIAL KEY

NJTCKY – CHARACTER TRANSMISSION CHARACTERISTICS KEY *

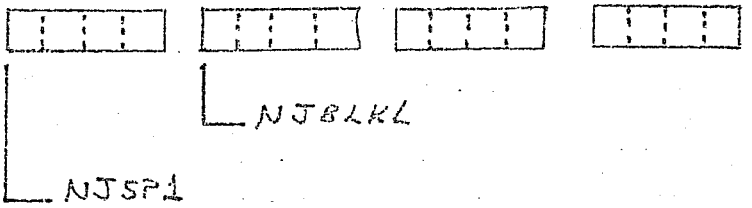
1

NJISPTP – INPUT STATES TABLE POINTER



NJCNT – INPUT STATE COUNTER 1 DCC1 – INTCC1
NJIBF – FCD OF FIRST BUFFER OF A SEGMENT

3



NJSP1 — NOT USED

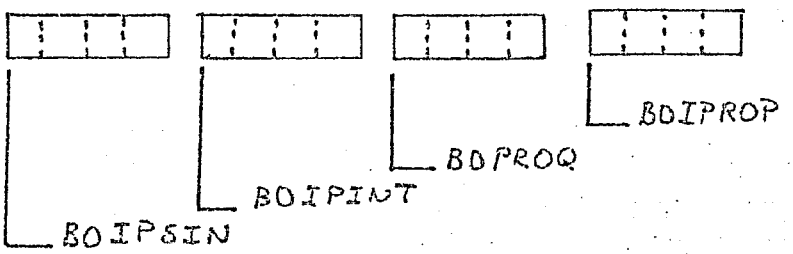
NJBLKL — INPUT STATE COUNTER 2 DCC2 - INTCC2

4

NJCXL — CODE XLATE TABLE ADDRESS

5

NJPOIKEY



BOIPSIN — POI POST INPUT KEY

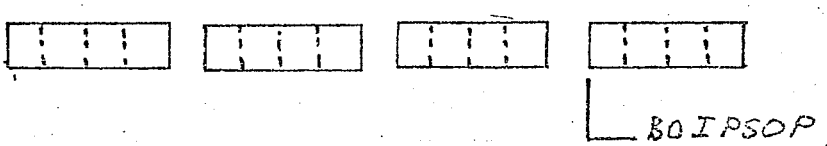
BOIPINT — POI INTERNAL INPUT KEY

BOPROQ — POI INTERNAL OUTPUT KEY

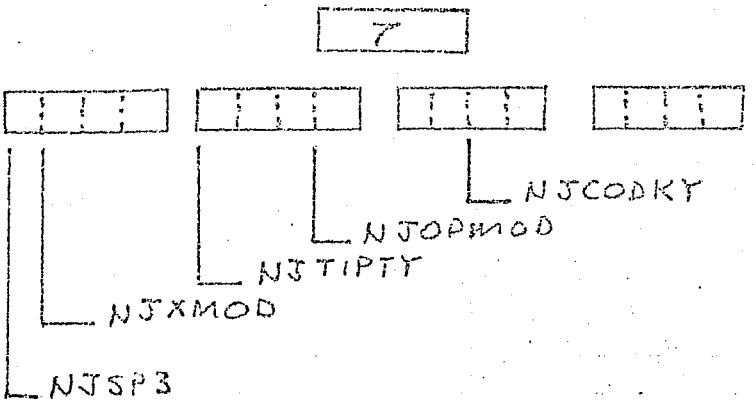
BOIPROP — POI PRE OUTPUT KEY

6

NJPOIKEY



BOIPSOP — POI POST OUTPUT KEY



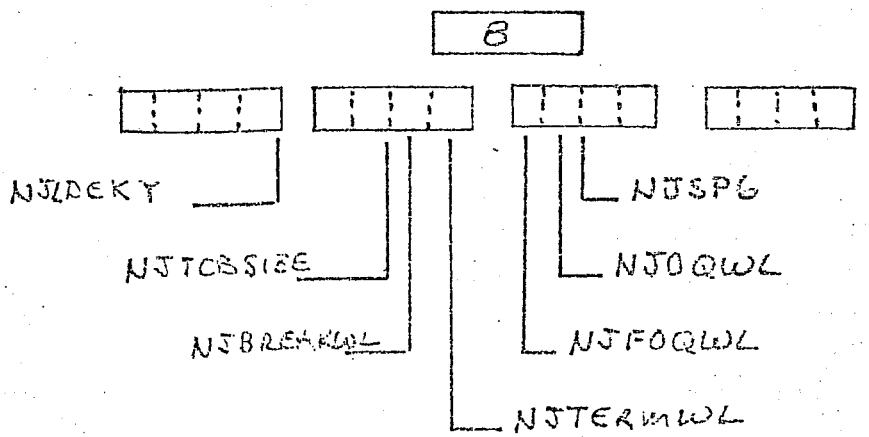
NJSP3 – NOT USED

NJXMOD – TRANSMISSION MODE 0 = NOFDX
1 = NO HDX
2 = NOEPX
3 = NO SPX

NJTIPTY – TIP TYPE 2 = TTY 4 = MODE 4

NJOPMOD – OPERATING MODE 0 = NOKEY
1 = NOTAPE
2 = NOTTT
3 = NO222

NJCODKY – CODE SET KEY



NJLDEKY – LINE DELAY KEY

NJTCBSIZE – TCB BUFFER SIZE

NJBREAKWL – BUILD W/L FOR TIP FOR BREAK COMMAND

NJTERMWL – BUILD W/L FOR TIP FOR TERMINATE COMMAND

NJFOQWL – BUILD W/L FOR TIP FOR FIRST OUTPUT QUEUE

NJOQWL – BUILD W/L FOR TIP FOR ANY OUTPUT QUEUED

NJSP6 – NOT USED

9

NJTCBBSUB — ADDRESS OF TCB BUILD SUBROUTINE

A	-	B

NJWD10 - NOT USED

NJWD11 - NOT USED

POI KEY

B1POI ↓ / B1KEY →	1	2	3	4	5
B1IPSIN →	PT4PSINP	PTYPINT			
B1IPINT →	PB21POI	PB22POI	PB23POI	PB24POI	PB25PO
B1PROQ →	PB31POI	PB32POI	PB33POI		
B1PROP →	PT4PROUT	PTYPREOUT			
B1IPSOP →	PT4PSOUT	PTYPOUT			

B1KEY FROM TERMINAL CHAR TABLE

BSTTYP TERMINAL TYPE ↓ / BOPOI →	BOIPSIN	BOIPINT	BOPROQ	BOIPROP	BOIPSOP
CONSOLE →		3	1		
COUPLER →		1	1		
BMD4–BMD4 → MODE4	1	2	3	1	1
ITTY–ATTY → TTY	2	2	3	2	2

B1IPSIN - POST INPUT

B1IPINT - INTERNAL INPUT

B1PROQ — INTERNAL OUTPUT

B1PROP - PRE OUTPUT

B1IPSOP - POST OUTPUT

TTY

PTYPINT - Post Input

Does statistics on input data block and then forewords it to the host

PTYPREOUT - Pre Output

Output from host is removed from the Q

PTYPOUT - Post Output

Statistics, send back, release buffers

MD4

PT4PSINP - Post input

Statistics, upline block handler

PT4PROUT - Pre output

Does nothing

PT4PSOUT - Post output

Statistics

Both + S.M

PB22P0I - Internal in

Set EOT flag in last non empty buffer change TCB

PB33P0I - Internal out

Down line block handler

HIP

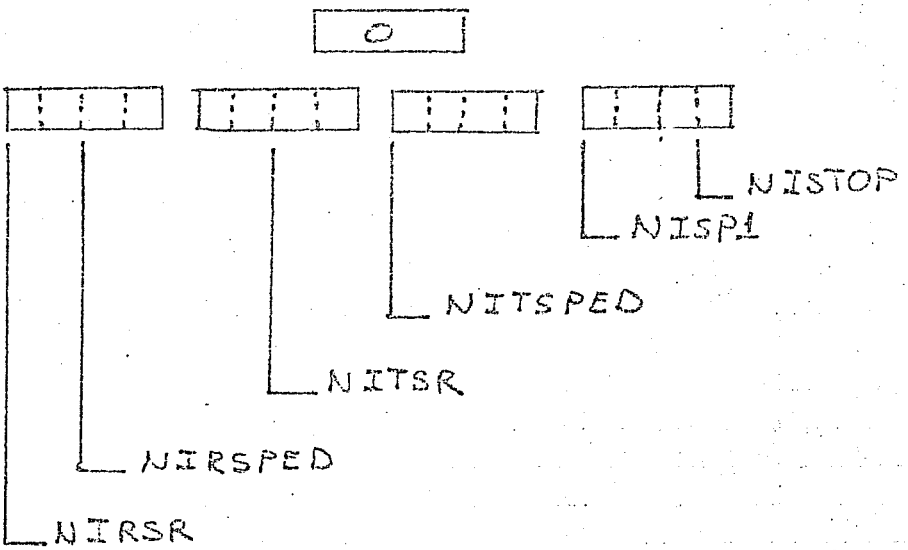
PB21P0I - Internal in

PNROUTE, generate WL if serv MSG, change TCB

PB31P0I - Internal out

Queue back to output TCB, generate WL if necessary

CHARACTER TRANSMISSION CHARACTERISTICS NICTCT



NIRSR – CLA RECEIVE SPEED RANGE

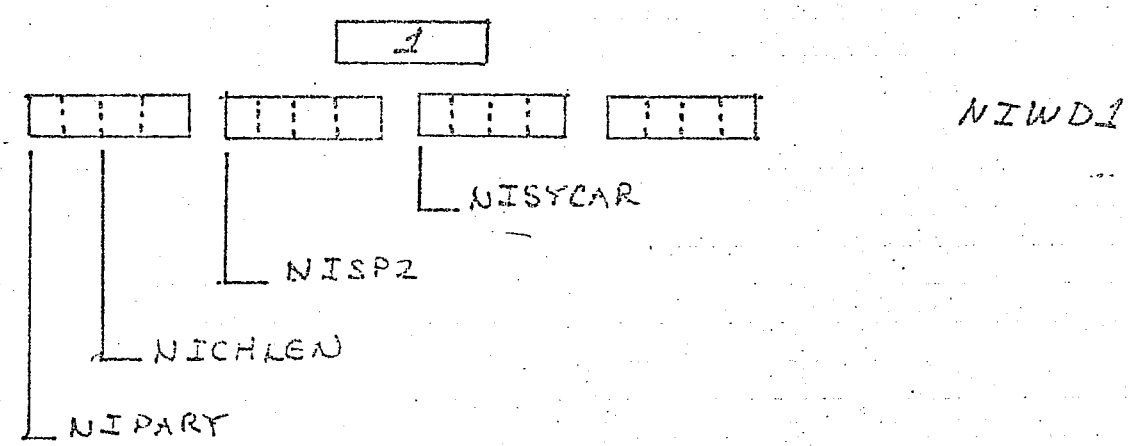
NIRSPED – CLA RECEIVE SPEED

NITSR – CLA TRANSMIT SPEED RANGE

NITSPED – CLA TRANSMIT SPEED

NISP1 – NOT USED

NISTOP – CLA STOP BITS 0 = 1-STOP BIT 1 = 2-STOP BITS

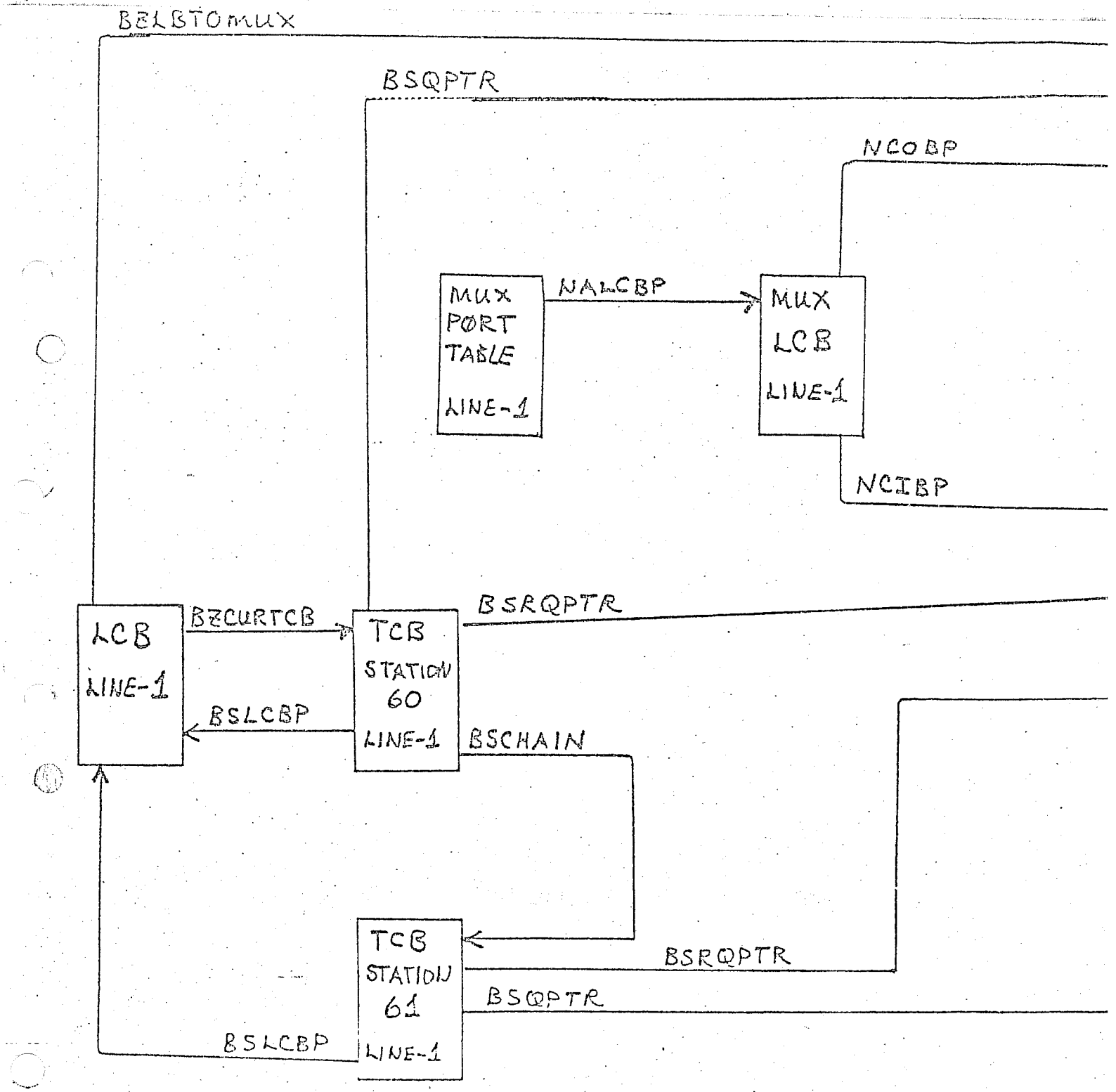


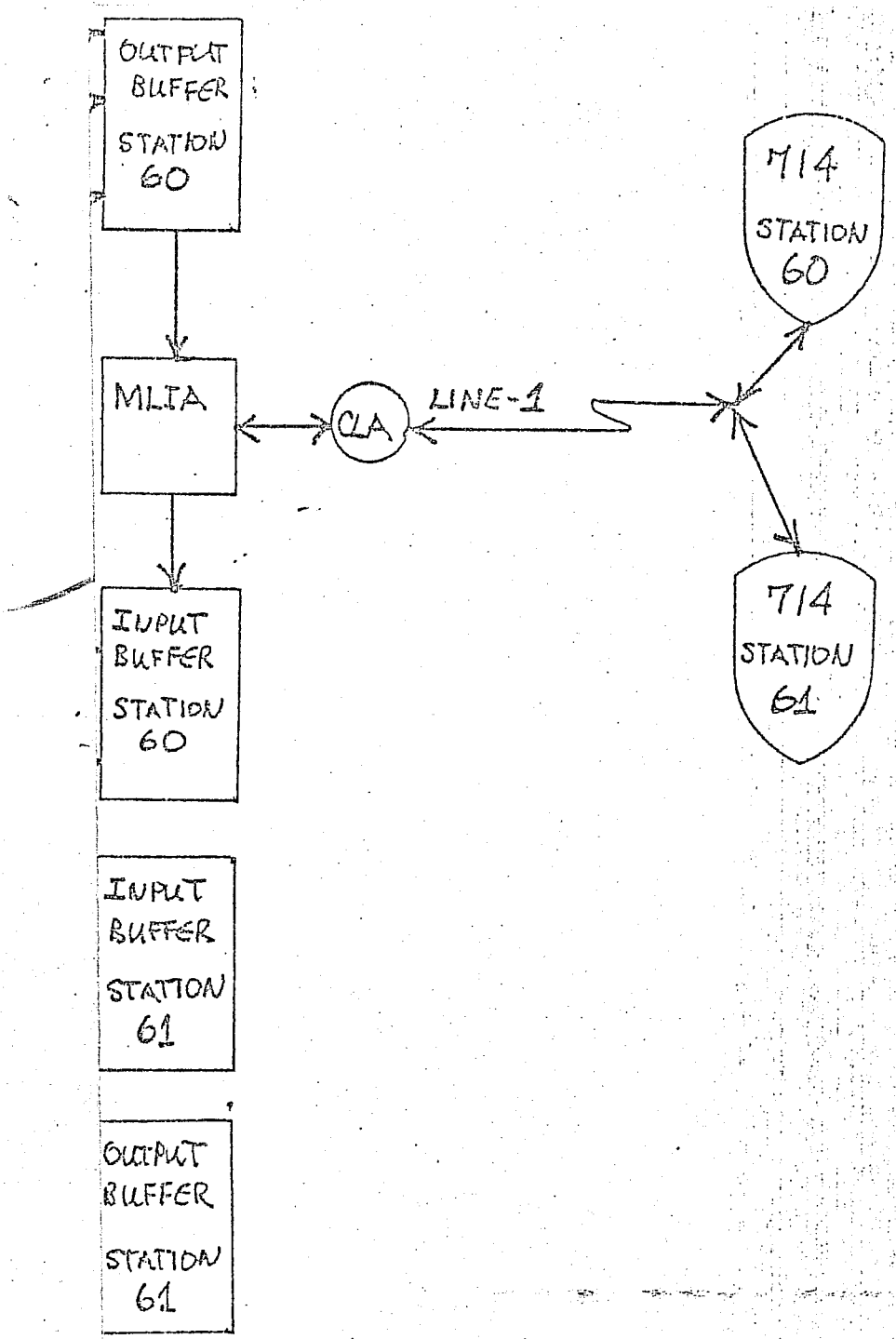
NIPARY – CLA PARITY MODE

NICHLEN – CLA CHARACTER LENGTH

NISP2 – NOT USED

NISYCAR – CLA SYNC CHARRACTER


BELBTOmux
BSQPTR
NCOBP
MUX PORT TABLE LINE-1
NALCBP
MUX LCB LINE-1
NCIBP
LCB LINE-1
BZCURTCB
TCB STATION 60 LINE-1
BSRQPTR
BSLCBP
BSCHAIN
TCB STATION 61 LINE-1
BSRQPTR
BSQPTR
BSLCBP

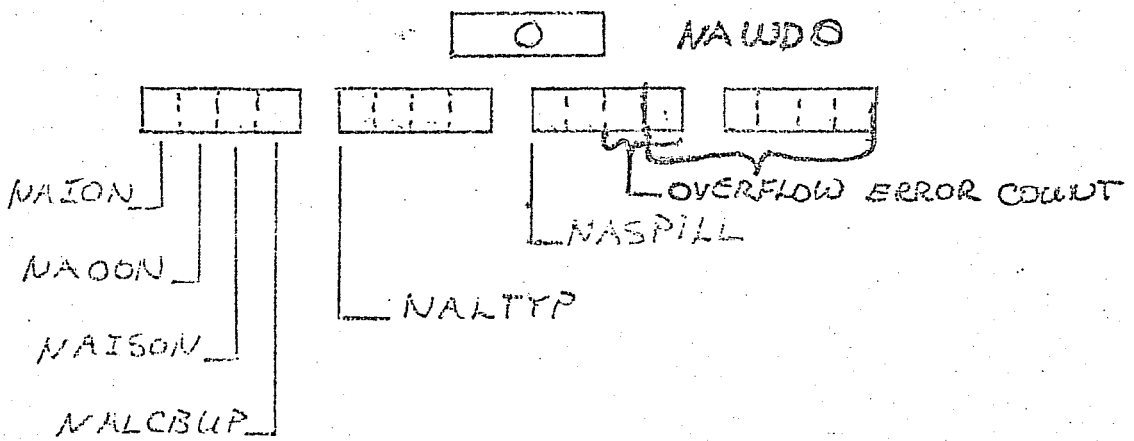

OUTPUT BUFFER STATION 60
MLIA
CLA
LINE-1
714 STATION 60
714 STATION 61
INPUT BUFFER STATION 60
INPUT BUFFER STATION 61
OUTPUT BUFFER STATION 61

MUX PORT TABLE

FWA = (160)

196D

0	1
NAION	NALCBP
NAOON	
NAISON	
NALCBUP	
NALTYP	
NASPILL	



NAION - INPUT ON

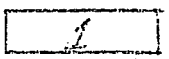
NAOON - OUTPUT ON

NAISON - INPUT STATUS SUPERVISION ON

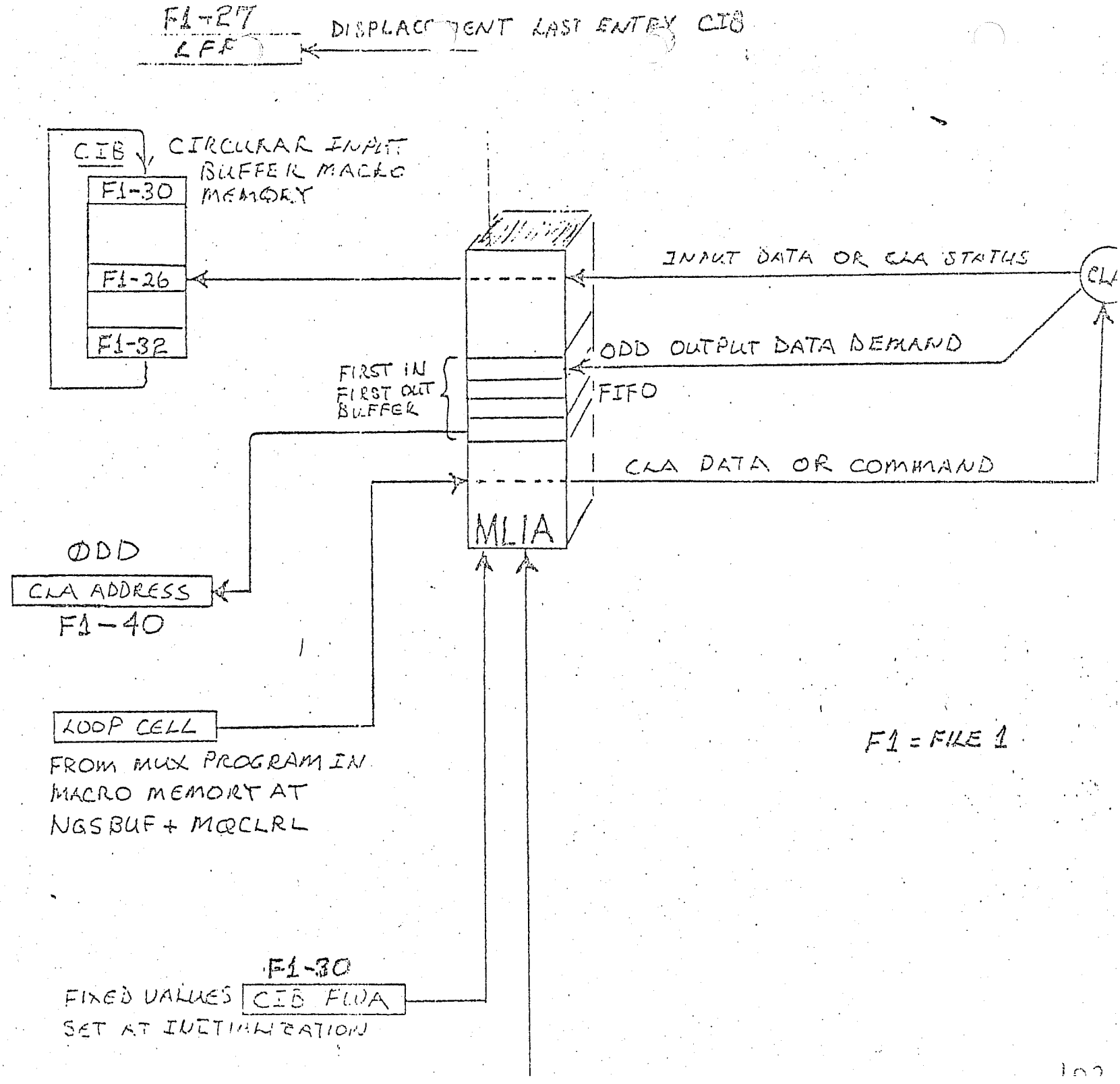
NALCBUP - LCB ASSIGNED

NALTYP - LINE TYPE SEE LCB B2LTYP

NASPILL - CAA STATUS ERROR COUNTER USED TO SET OVERFLOW STATUS



NALCBP - POINTER (MUX LCB)


F1-27
LFE
DISPLACEMENT LAST ENTRY CIB
CIB
CIRCULAR INPUT BUFFER MACRO MEMORY
F1-30
F1-26
F1-32
INPUT DATA OR CLA STATUS
ODD OUTPUT DATA DEMAND
FIRST IN FIRST OUT BUFFER
FIFO
CLA DATA OR COMMAND
MLIA
ODD
CLA ADDRESS
F1-40
LOOP CELL
FROM MUX PROGRAM IN MACRO MEMORY AT NGSBUF + MQCLRL
F1 = FILE 1
F1-30
FIXED VALUES
CIB FLWA
SET AT INITIALIZATION

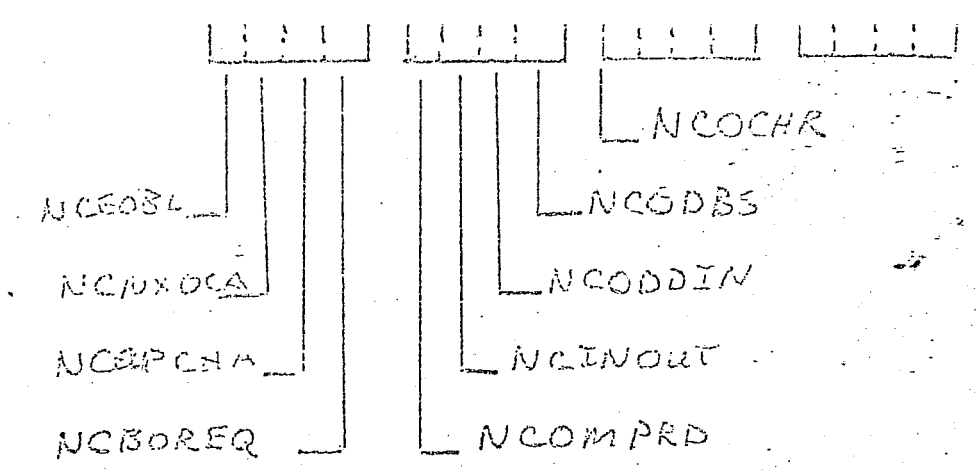
MUX

NCLCB — LINE CONTROL BLOCKS

0	1	2	3	4	5	6	7
NCEOBL NCAXOCA NCUPCHA NCBOREQ NCOMPRO NCINOUT NCODDIN NCODBS NCOCHR	NCENDX NCCARR NCSPL NCTIME NCOBKCD	NCOBP	NCRINCH NCCAREC NCNOXL NCRPRT NCIDBS NCSP2 NCINPRO NCSP4 NCSP5 NCISTA	NCCNTL NCCNT1	NCISPTA	NCIBP	NCDM NCTTY NCSP6 NCCRC

8	9	A	B	C	D	E	F
NCSCHR NCIBFCD	NCCRCS	NCZER1 NCCNT2	NCZER2 NCBLKL	NCCXLTA	NCSCBA		

MUX PORT TABLE ENTRY NALCBP HAS POINTER TO NCLCB FOR EACH LINE THAT IS ENABLED



NCEOBL — END OF BLOCK FLAG

NCNXOCA — NEXT OUTPUT CHARACTER AVAILABLE

NCEUPCHAIN — SUPPRESS BUFFER CHAINING

NCBOREQ — BUFFER CHAINING REQUIRED

NCOMPRD — OUTPUT MSG IN PROGRESS

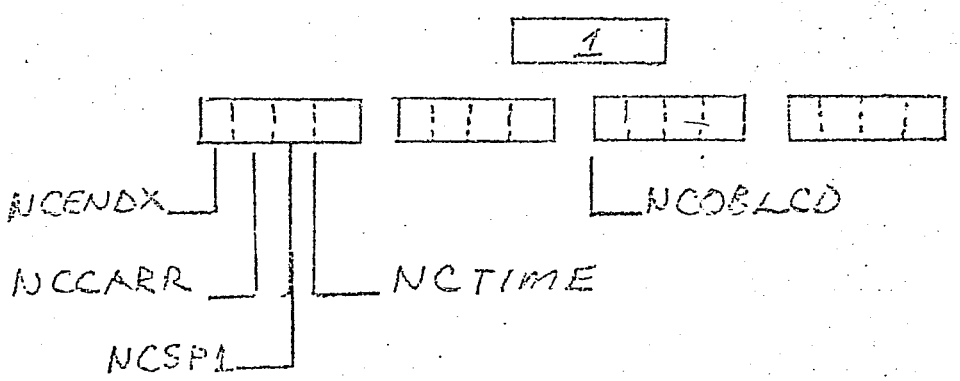
NCINOUT — INPUT AFTER OUTPUT FLAG

NCODDIN — ODD RECEIVED FLAG

NCODBS — OUTPUT DATA BUFFER SIZE

0 = SMALL 1 = LARGE

NCOCHR — NEXT OUTPUT CHARACTER



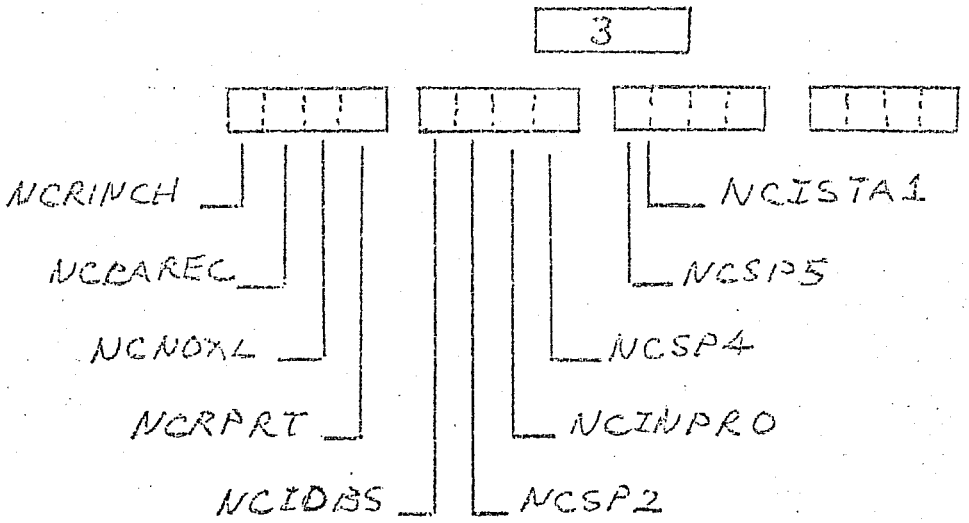
NCENDX — END OF TRANSMISSION NCSPI — SPARE

NCCARR — CARRIER

NCTIME — TIMER

NCOBLCD — LCD OF OUTPUT BUFFER

NCOBP — OUTPUT BUFFER POINTER



NCRINCH — INPUT CHARACTER IN RIGHT BYTE 8 BITS

NCCAREC — CHARACTER RECEIVED FLAG

NCNOXL — CODE TRANSLATE ACTIVE

NCRPRT — REMOVE PARITY BIT

NCIOBS — INPUT DATA BUFFER SIZE

0 = SMALL 1 = LARGE

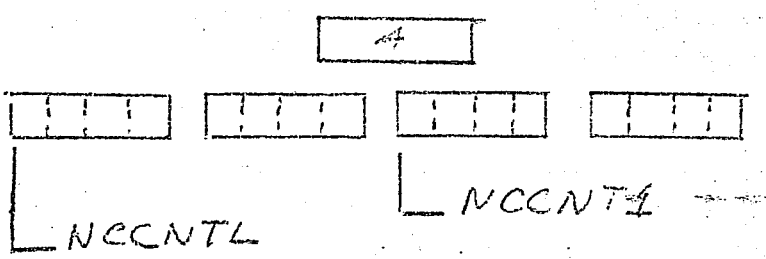
NCSP2 — NOT USED

NCINPRO — INPUT MSG IN PROGRESS

NCSP4 — NOT USED

NCSP5 — NOT USED

NCISTA1 — INDEX TO INPUT STATES POINTERS



NCCNTL — CHARACTER COUNT LIMIT SET FROM TERMINAL CHARACTERISTICS TABLE VALUE NJCNT1

NCCNT1 — CURRENT CHARACTER 1 COUNT FROM INTCC1 OR DCC1 COMMAND IN STATES TABLE

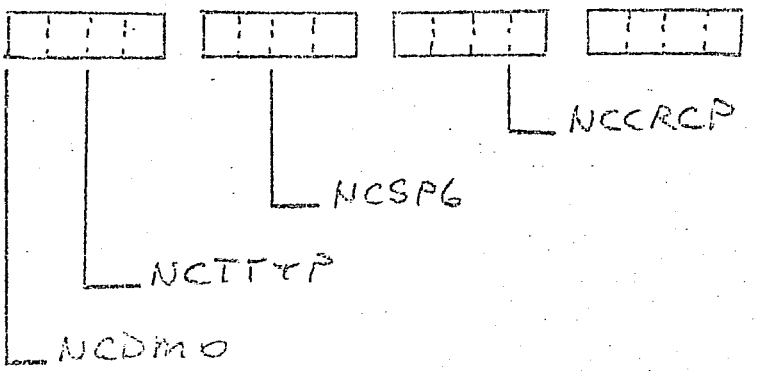
5

NCISPTA — ADDRESS OF INPUT STATES POINTER TABLE

6

NCIBP — INPUT DATA BUFFER POINTER

7



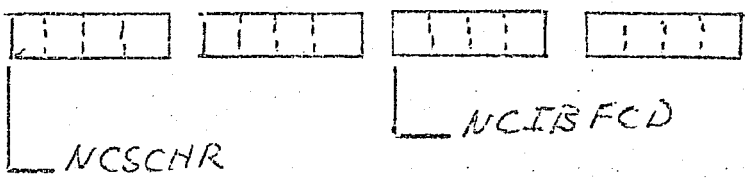
NCDMO — NOT USED

NCTTYP — TERMINAL TYPE SAME AS BESTATE OF LCB BETTYP

NCSP6 — NOT USED

NCCRCP — CRC POLYNOMIAL

8



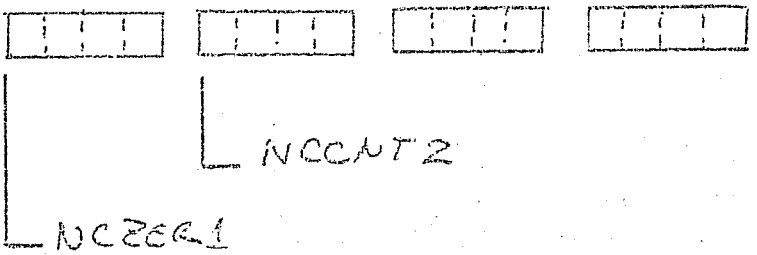
NCSCHR — SPECIAL CHARACTER

NCIBFCD — FCD OF INPUT BUFFER

9

NCCRCS — CRC ADDRESS

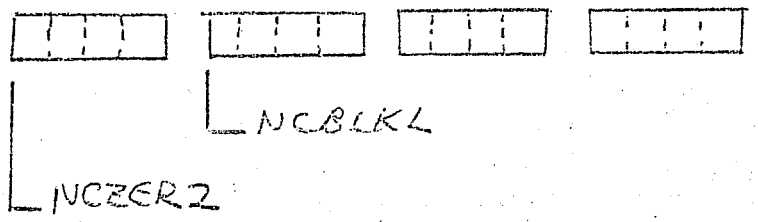
A



NCZER1 – NOT USED ZERO

NCCNT2 – CHARACTER COUNTER-2 SET FROM INITCC2 OR PCC2 COMMAND IN STATES TABLE

B



NCZER2 – NOT USED ZERO

NCBLKL – MAX BLOCK LENGTH SET FROM LINK CHARACTER TABLE NJBLKL USED TO INITIALIZE NCCNT1 WORD A

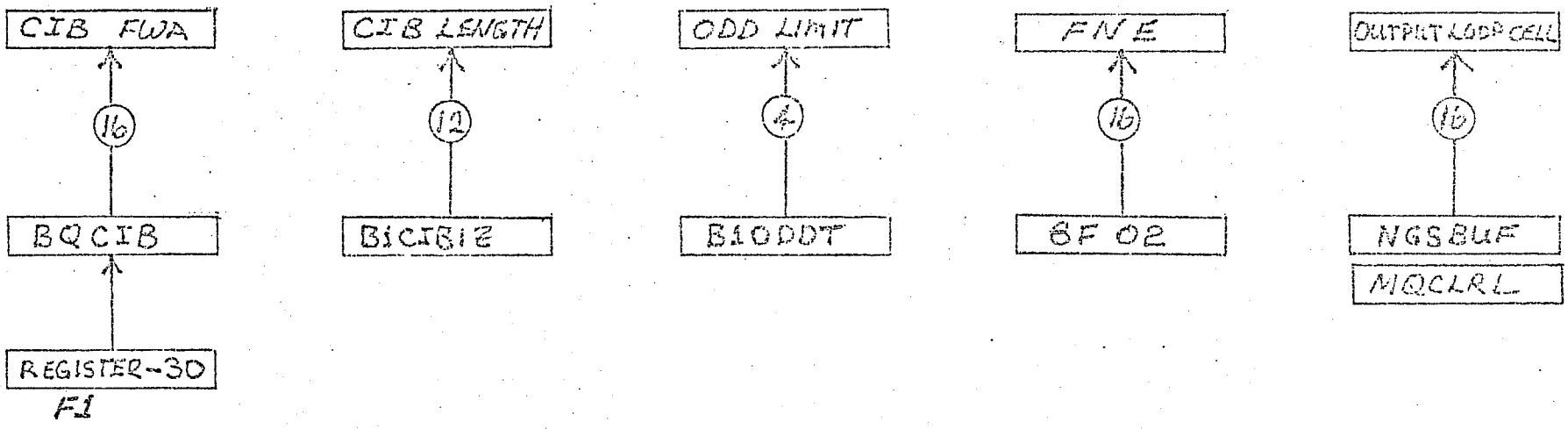
C

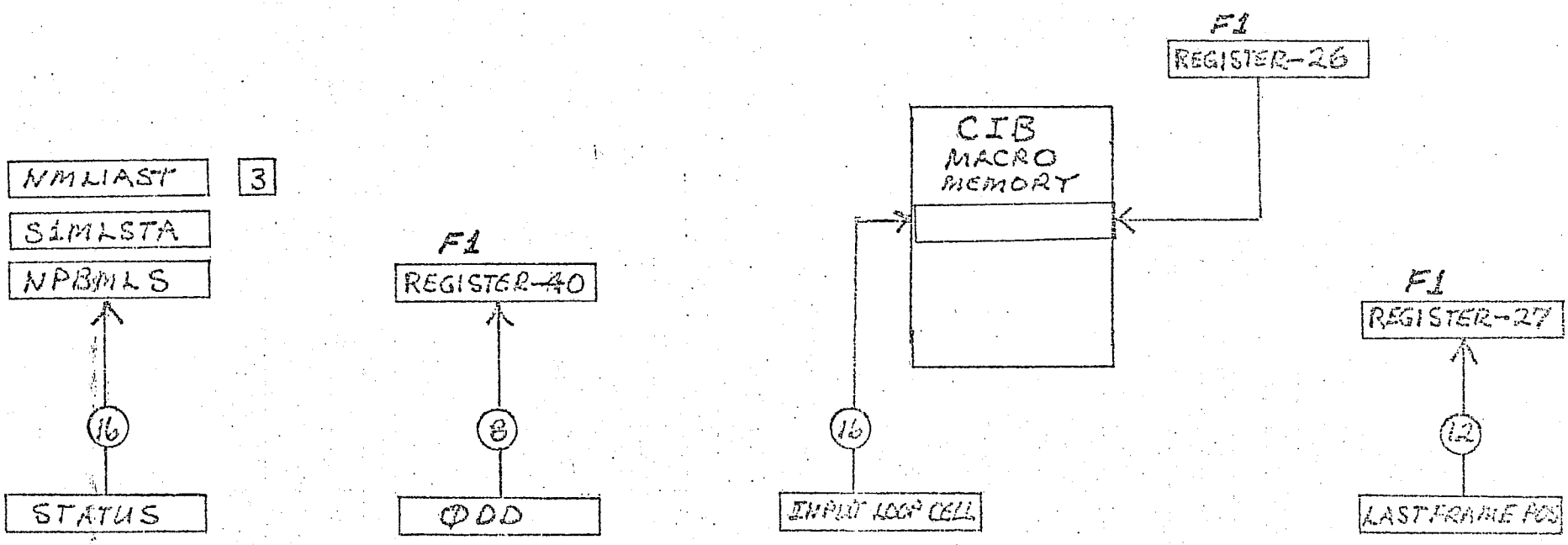
NCCXLTA – CODE TRANLATE TABLE ADDRESS

D

NCSCBA – FIRST BUFFER OF BLOCK ADDRESS

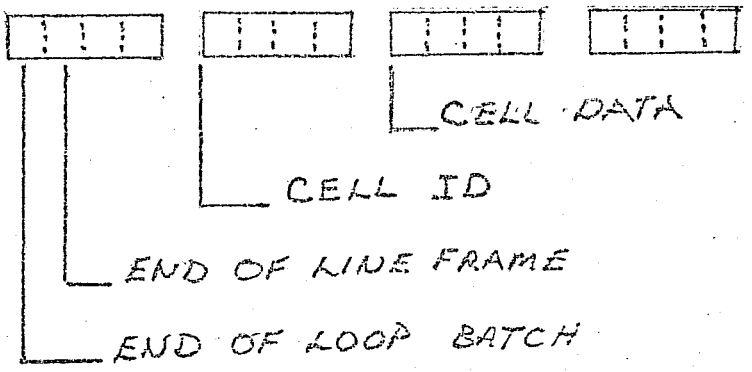
TO MLIA





FROM MLIA

INPUT LOOP CELL



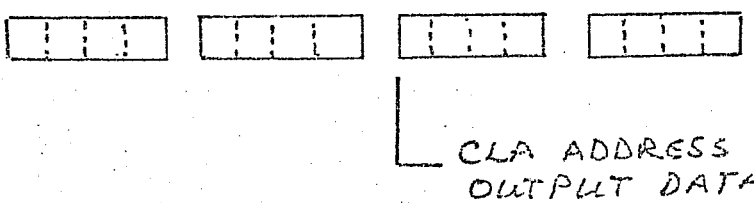
CELL ID

C - INPUT DATA RECEIVED FROM TERMINAL

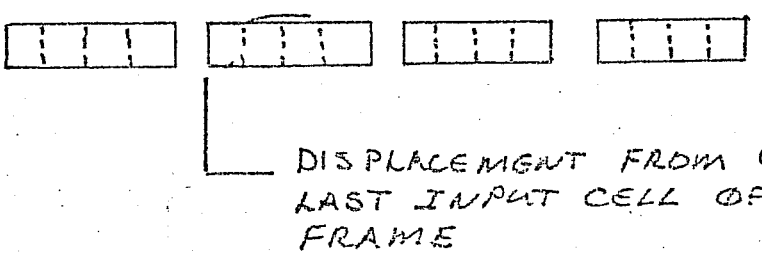
D - CLA STATUS WORD

E - CLA ADDRESS

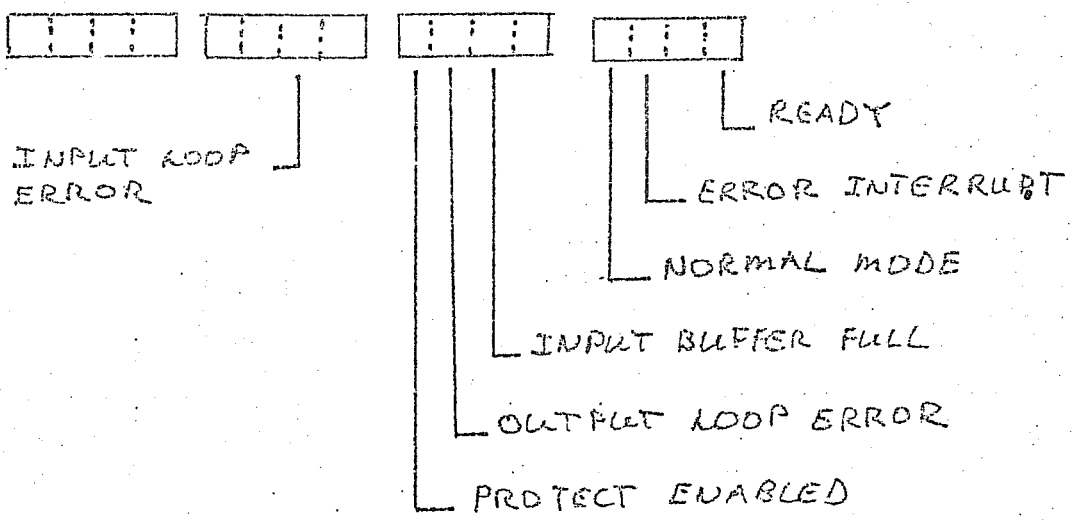
ØDD INPUT



LFP INPUT



MLIA STATUS



READY — MLIA CARD PLUGGED IN AND HAS POWER APPLIED INPUT AND OUTPUT LOOPS ACTIVE

ERROR INTERRUPT — INPUT BUFFER FULL + OUTPUT LOOP ERR + INPUT LOOP ERROR

PROTECT ENABLED — MLIA ALLOWED TO WRITE OR READ PROTECTED AREA IN MEMORY

INPUT BUFFER FULL — INPUT FIFO OR ODD FIFO ARE FULL

NORMAL MODE — NOT IN DIAGNOSTIC MODE

STATE PROCESS INSTRUCTIONS

OPCODE (HEX)		DESCRIPTION
00	/NOP	NOP
01	/SETINP /RSTINP	Set/Reset input message in process flag
02	/RPLC	Replace character
03	/BLDWL	Build event worklist
04	/TIBWL	Terminate input buffer
05	/CRSEQ	Skip if CRC equal
06	/DCC1, /DCC2	Decrement character count
07	/INTCC1, /INTCC2	Initialize character count
08	/STATE	Set input state
09	/STBLC	Store block length character
0A	/CHARLS	Skip if character is less than operand
0B	/STATLS	Skip if state is less than operand
0C	/CHARNE	Skip if character not equal
0D	/SPCHEQ	Skip if special character equal
0E	/RESYNC	Resync
0F	/SETRAN, /RSTRAN	Set/reset translate mode
10	/INTCRC	Reset cyclic checksum storage

Bits 15 8 7 5 4 0

F2	Operand	F1	Op Code

Entry Point	F1 Value	Entry Point Function
/EXIT	1	001	Discard character
/STOREXIT	2	010	Store character
/CRCSTOREXIT	3	011	Accumulate CRC and store character*
/CRCEXIT	4	100	Accumulate CRC and discard character
	5	101	Undefined
	6	110	Undefined
	7	111	Undefined

Exit when F1 not equal to 0 except for Skip instruction

OP CODE	F2 = 0	F2 = 1
01	Reset msg in process flag	Set msg in process
04	--	Repoint input buffer, Store LCD
05	Compare 8 bits	Compare 7 bits
06	Decrement counter 1	Decrement counter 2
07	Initialize counter 1	Initialize counter 2
08	Execute F1	Ignore F1
0F	Set Xlate mode	Resets Xlate mode
10	Reset checksum to 0	Reset checksum to 1

HOST COUPLER

TCB

0	1	2	3	4	5	6	7

WORD 0-9 SEE MODE4 TERMINAL TCB P78
BSLCB = SUB LINE CONTROL BLOCK NUMBER-3

8	9	A	B	C	D	E	F
		BSCBSENT	BSCBRECV	BSCHSENT	BSCHRECV	BSCBUFRQ	BSCP

10	11	12	13	14	15	16	1
BSCPINPUT	BSCPAVPTR	BSCPCONN	BSCPOUTPUT	BSBUFOT	BSCPSTAT	BSCPDATA	BSCP

18	19
BSCPBUFAV BSCPBSTA	BSCPIDLT

A

BSCBSENT — BLOCKS TRANSMITTED TO HOST

B

BSCBRECV — BLOCKS RECEIVED FROM HOST

C

BSCHSENT — CHARACTERS SENT TO HOST

D

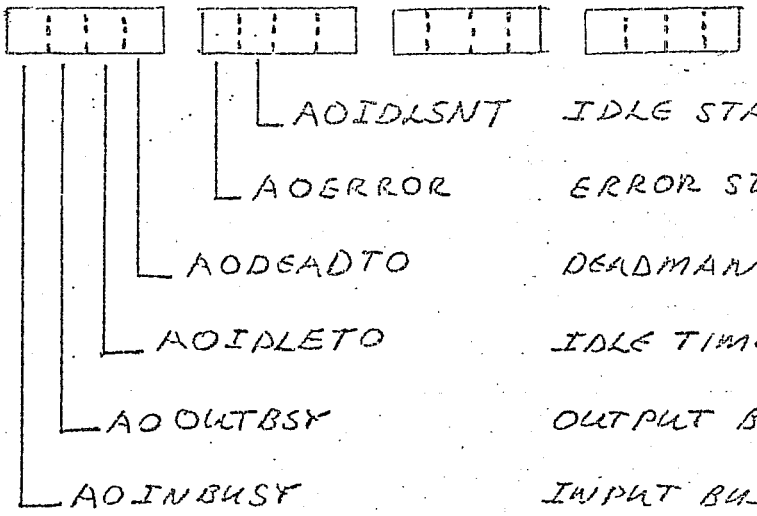
BSCHRECV — CHARACTERS RECEIVED FROM HOST

E

BSCBUFRQ — REQUEST FOR ADDITIONAL BUFFERS FOR MSG FROM HOST

F

BSCPFLAGS – COUPLER FLAGWORD



10

BSCPINPUT – INPUT BUFFER TO HOST WAITING TO BE SENT

11

BSCPAVPTR – NEXT FREE BUFFER

12

BSCPCONN – COUPLER CONNECT CODE

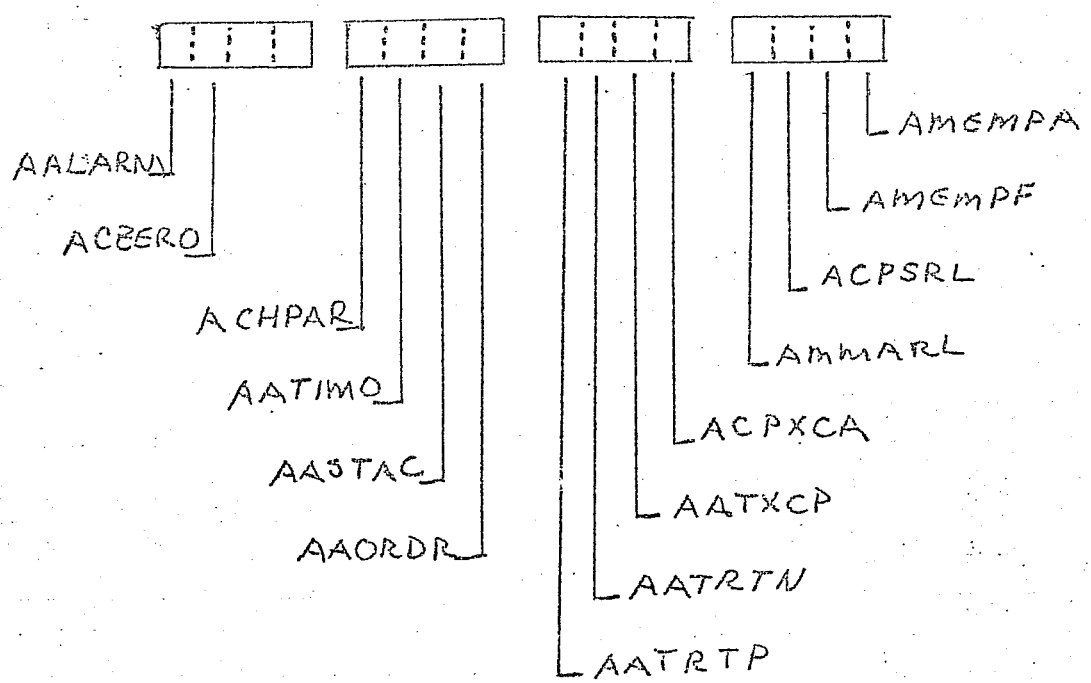
13

BSCPOUTPUT – LAST BUFFER RECEIVED FROM HOST

14

BSBUFOT – MAR BUFFER ADDRESS STORAGE. FIRST BUFFER ADDRESS SENT TO COUPLER COULD BE INPUT OR OUTPUT

BSCPSTATUS – COUPLER STATUS



AALARM – POSITIVE TRANSITION OF ANY OTHER BIT OCCURED

ACZERO – CHAIN ADDRESS ZERO

ACHPAR – CYBER 170 PARITY ERROR

AATIMO – HARDWARE TIMEOUT CHANNEL-ACTIVE 3 SEC

AASTAC – HOST READ NPU STATUS

AAORDR – HOST HAS LOADED ORDER WORD

AATRTP – HOST SET CHANNEL INACTIVE DURRING TRANSFER

AATRTN – NPU TERMINATED TRANSFER

AATXCP – HOST COMPLETED TRANSFER

ACPXCA – POWER FAILURE

AMMARL – MAR MEMORY ADDRESS JUST LOADED

ACPSRL – NPU STATUS REGISTER LOADED

AMEMPF – MEMORY PROTECT FAULT

AMEMPA – MEMORY PARITY ERROR

BSCPDATA - STORAGE FOR ORDER WORD / INPUT DATA
SET BY HOST TO INDICATE IT HAS OUTPUT FOR
NPU 1 = BACK, 2 = BLK, 3 = MSG, 4 = CMD, 5 = BREAK, 6 = RESET, 7 = TER

17

BSCPCMD
BSCMND - LAST COUPLER COMMAND

10 = ACPIMA INPUT MEMORY ADDRESS

30 = ACPIFC INPUT FCD / PCD

40 = ACPINS INPUT NPU STATUS

* 50 = ACPICS INPUT COUPLER STATUS

* 60 = ACPIOW INPUT ORDER WORD

70 = ACPIIO INPUT I/O

04 = ACPILW INPUT LAST WORD FROM DATA CHANNEL

14 = ACPIFD INPUT FDMAR0 / FDMAR1

24 = ACPILC INPUT LAST CHARACTER FROM DMA UPPER

34 = ACPIFL INPUT FDMAI / FLAGREGISTER

* 54 = ACPISS INPUT SWITCH STATUS

74 = ACPICH INPUT CHARACTER

08 = ACPOMZ OUTPUT MEMORY ADDRESS ZERO

38 = ACPOFP OUTPUT FCD / PCD / LCD

* 48 = ACPONS OUTPUT NPU STATUS

* 58 = ACPOBL OUTPUT BUFFER LENGTH

68 = ACPOOW OUTPUT ORDERWORD

* 0C = ACPCLR CLEAR COUPLER

1C = ACPTRM TERMINATE TRANSFER

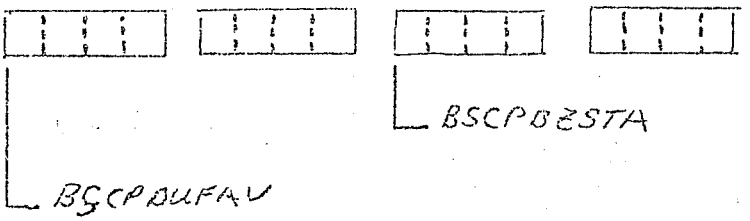
4C = ACPOTS OUTPUT TEST

5C = ACPITS INPUT TEST

* 6C = ACPOMA OUTPUT MEMORY ADDRESS

7C = ACPOCH OUTPUT CHARACTER

* = FUNCTIONS USED BY CCP



BSCPBUFAV — NUMBER OF ALOCATED BUFFERS

BSCPBESTA - TEMP STORAGE FOR BESTATE

1 = AOPT1 WAIT FOR ACCERO + AATIMO + AATRTP + AATXC
THEN GO TO STATE AOPT2 ON INPUT FROM HO
1. CHAIN ADDRESS ZERO
2. HARDWARE TIMEOUT
3. TRANSFER TERMINATED BY PPU
4. TRANSFER COMPLETE

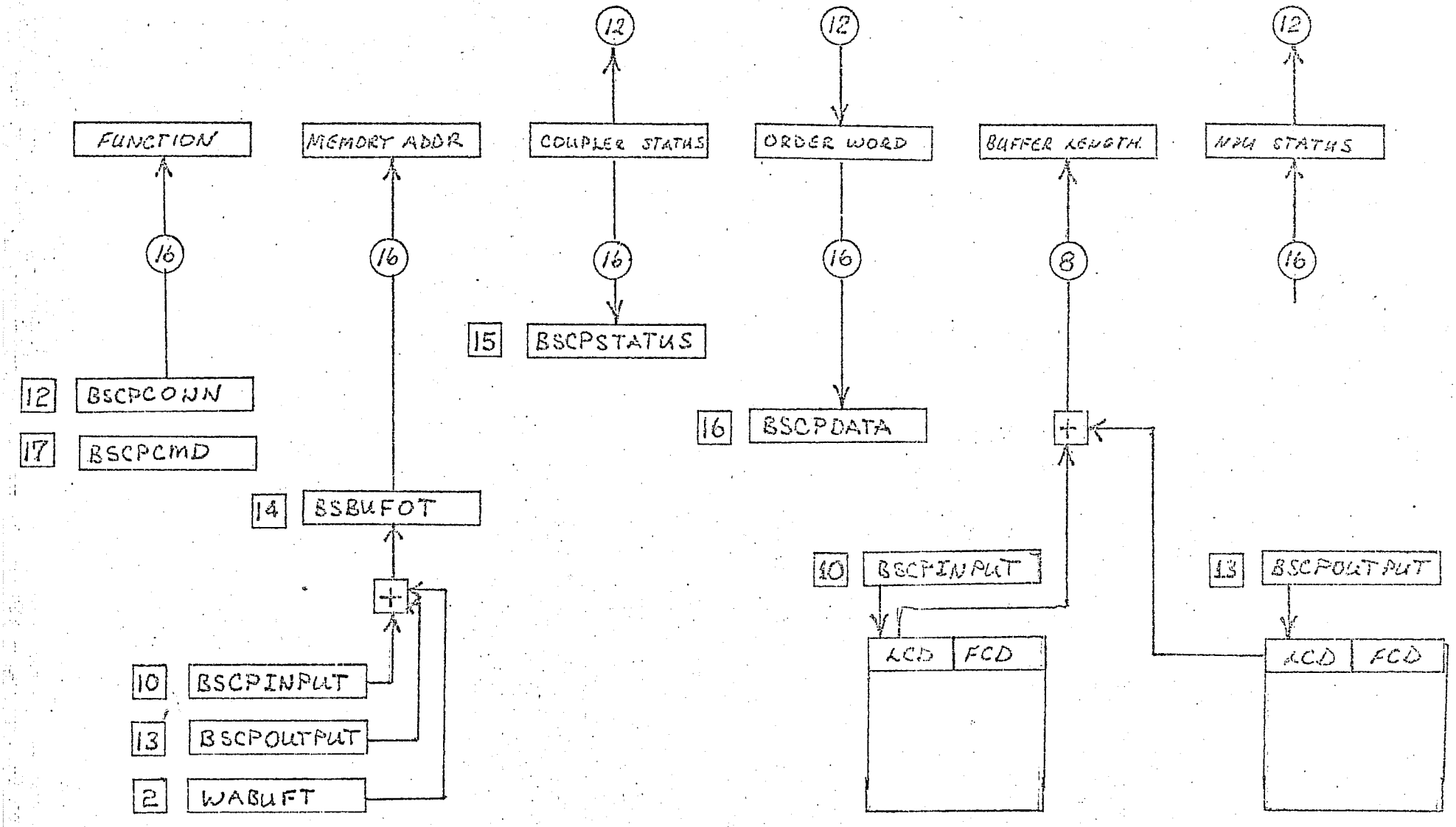
2 = AOPT2 COMPLETE INPUT TRANSMISSION FROM HOS

3 = AOPT3 INITIALIZE OUTPUT TRANSACTION

4 = AOPT4 WAIT FOR ACCERO + AATIMO + AATRTP + AATXCP
THEN GO TO STATE AOPT5 ON OUTPUT TO HOST

5 = AOPT5 COMPLETE OUTPUT TO HOST TRANSMISSION

BSCPIDLT — IDLE TIME COUNTER


FUNCTION
MEMORY ADDR
COUPLER STATUS
ORDER WORD
BUFFER LENGTH
MPH STATUS
12
16
8
12 BSCPCONN
17 BSCPCMD
14 BSBUFOT
15 BSCPSTATUS
16 BSCPDATA
10 BSCPINPUT
13 BSCPOUTPUT
2 WABUFT
LCD
FCD

NPU STATUS

AIDLE	1	IDLE STATUS
AAREADY	4	READY TO ACCEPT OUTPUT FROM HOST
AANREADY	7	NOT READY TO ACCEPT OUTPUT FROM HOST
AINPAV	11	INPUT AVAILABLE TO HOST NOT BLK OR MSG
AINSSB	13	INPUT AVAILABLE TO HOST SMALL BLK OR MSG
AINPLB	14	INPUT AVAILABLE TO HOST LARGE BLK

NPU FUNCTIONS USED BY CCP 1.0

ACPCLR	0C	CLEAR COUPLER
ACPONS	48	OUTPUT NPU STATUS
ACPICS	50	INPUT COUPLER STATUS
ACPOBL	58	OUTPUT BUFFER LENGTH
ACPOOW	60	INPUT ORDER WORD
ACPOMA	6C	OUTPUT MEMORY ADDRESS

HOST ORDER WORD

HTBACK	1	BACK BLOCK ACKNOWLEDGEMENT
HTBLK	2	DATA BLOCK
HTMSG	3	MSG OR SMALL DATA
HTCMD	4	COMMAND
HTBREAK	5	BREAK
HTRESET	6	RESET BREAK
HTTERM	7	TERMINATE

TABLE 3-5. COUPLER STATUS REGISTER

Bit Number	Flag Name	Set Condition	Interrupt Alarm*	Reset** Condition
0	Memory Parity	NPU memory parity error	A	1
1	Memory Protect Fault	NPU memory protect fault	A	1
2	NPU Status Word Loaded	NPU writes status word	-	2
3	Memory Address Register Loaded	Host or NPU writes memory address one	-	3
4	External Cab. Alarm	Power failure	I	4
5	Transmission Complete	Host completes any input or output operation	I	4
6	Transfer Terminated by NPU	NPU terminates transfer (not used)	I	4
7	Transfer Terminated by Host	Host sets channel inactive during data I/O	I	4
8	Orderword Register Loaded	Host writes orderword	I	5
9	NPU Status Read	Host reads NPU status word	-	4
10	Timeout	Coupler selected and active 3+ seconds in host data I/O operation	I	1
11	CYBER 170 Channel	Enable parity switch on and data channel 12-bit word (plus parity) not odd	A	6
12-13	Not Used			
14	Chain Address Zero	Coupler finds zero in last word of NPU buffer	-	4
15	Alarm	Positive transition of any flag causing alarm	A	4

NOTES:

*Raising associated flag causes alarm (A), interrupt (I) or neither alarm or interrupt (-).

**All flags are reset (cleared) by Master Clear. All except bit 2 are reset when NPU or host clears the coupler. Other reset conditions are as follows:

1. Reset when coupler status register is cleared.
2. Reset when host reads NPU status word.
3. Reset on first direct memory access (DMA).
4. Reset when NPU reads coupler status register.
5. Reset when NPU reads orderword.
6. Reset when NPU reads coupler status register or by enable parity switch positive transition.

WORK LIST CONTROL BLOCK

0	1	2	3	4	5	6	7
BYCNT	BYPUT	BYGET	BYFEINC BYINC	BYEXPRI	BYOVLY	BYNPPT	BYPRADD

0

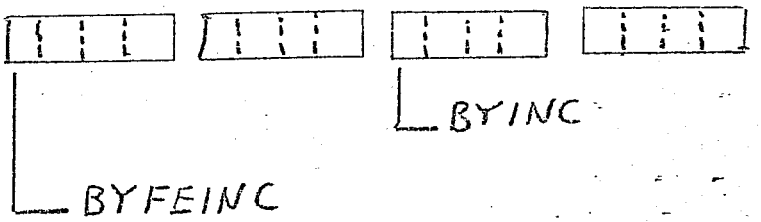
BYCNT — ENTRY COUNT

1

BYPUT — PUT LOCATION FOR NEXT ENTRY

2

BYGET — GET LOCATION FOR NEXT ENTRY



BYFEINC — FIRST ENTRY DISPLACEMENT IN BUFFER

BYINC — LENGTH OF ENTRY

4

BYEXPRI — MONITOR CONTROL 0 = SCAN WORK LIST

1 = INHIBIT SCAN

2 = RETURN TO PROG

5

BYOVLY — OVERLAY INDEX

6

BYNPPT — LINK TO NEXT WORK LIST CONTROL BLOCK TO SCAN

7

BYPRADDR — ENTRY POINT FOR ROUTINE

PROGRAMS WITH WORK LIST IN THE ORDER THEY APPEAR IN THE WLCB

1. PBCONS	– CONSOLE	9EA
2. PNSMWL	– SERVICE MODULE	9F2
3. PBMLIAOPS	– MLIA HANDLER	9FA
4. PBTIMA	– TIMING SERVICES	A02
5. PB2IPO	– INTERNAL PROCESSOR	A0A
6. PTTYP	– TTY TIP	A12
7. PTMOD4T	– MODE 4 TIP	A1A
8. PBTIPD	– TIP DEBUG	A22
9. PBCONTINUE	– CONTINUE WORKLIST	A2A
10. PTSTAR	– COUPLER HIP	A32
11	– TTY CONSOLE DRVR PRIORITY	A3A
12.	– TTY CONSOLE DRVR NON PRIORITY	A42
13. PMWOLP	– MUX EVENT	A4A
14.	– HIP DATA LIST QUEUE	A52
15. PMT200M	– CRITICAL 200MS TIMING	A5A
16.	– DUMMY	A62

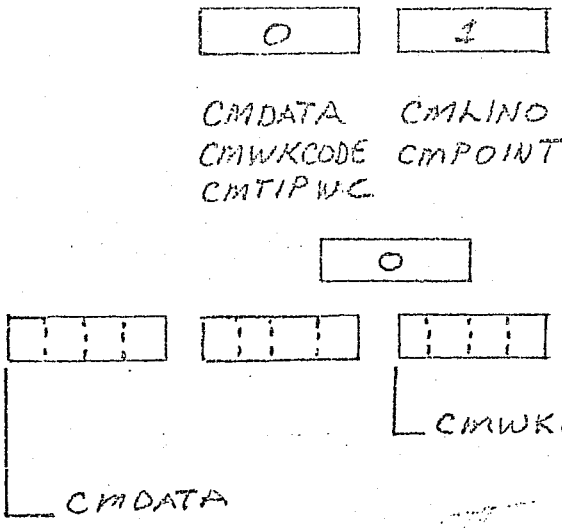
TIP – HIP – LIP

0 = AOCONTINUE	CONTINUE WORK CODE
20 = AOTIMEOUT	TIMER EXPIRED
21 = AOQUEOUT	OUTPUT IN QUEUE
22 = AOSMEN	ENABLE LINE FROM SERVICE MODULE
23 = AOSMDA	DISABLE LINE FROM SERVICE MODULE
24 = AOSMTCB	TCB IS BUILT FROM SERVICE MODULE
25 = AOBREAK	DOWN LINE BREAK RECEIVED
26 = AOTERM	DOWN LINE TERMINATE RECEIVED
27 = COSMSGR	SERVICE MESSAGE RECEIVED TO SM
28 = COTCBREL	RELEASE TCB. TO SERVICE MODULE
29 = COSMDISP	DISPATCH SERVICE MESSAGE TO SM
2A = COLINOP	LINE OPERITIONAL TO SERVICE MODULE
2B = COLINOP	LINE INOPERATIVE "
2C = COLNDA	LINE DISABLED "
2D = CODLYTCB	DELAYED TCB BUILD "
2E = COHSTDN	HOST DOWN FROM HIP "
2F = COHSTUP	HOST UP FROM HIP "
30 = COCHDM	CONTINUE HOST DOWN MSG "
31 = COCHUM	CONTINUE HOST UP MSG "
32 = AOBRON	TURN ON BREAK TO PMT200M
33 = AOBROFF	TURN OFF BREAK TO PMT200M
31 = AOWK1	WORK CODE 1
32 = AOWK2	WORK CODE 2
33 = AOWK3	WORK CODE 3
34 = AOWK4	WORK CODE 4

SERVICE MODULE

WORK LIST ENTRY PNSMWL 9F2

BOSMWL



CMDATA – LINE ERROR CODE OR CONFIGURE TERMINAL TYPE

CMWKCODE – EVENT WORK CODE
CMTIPWC

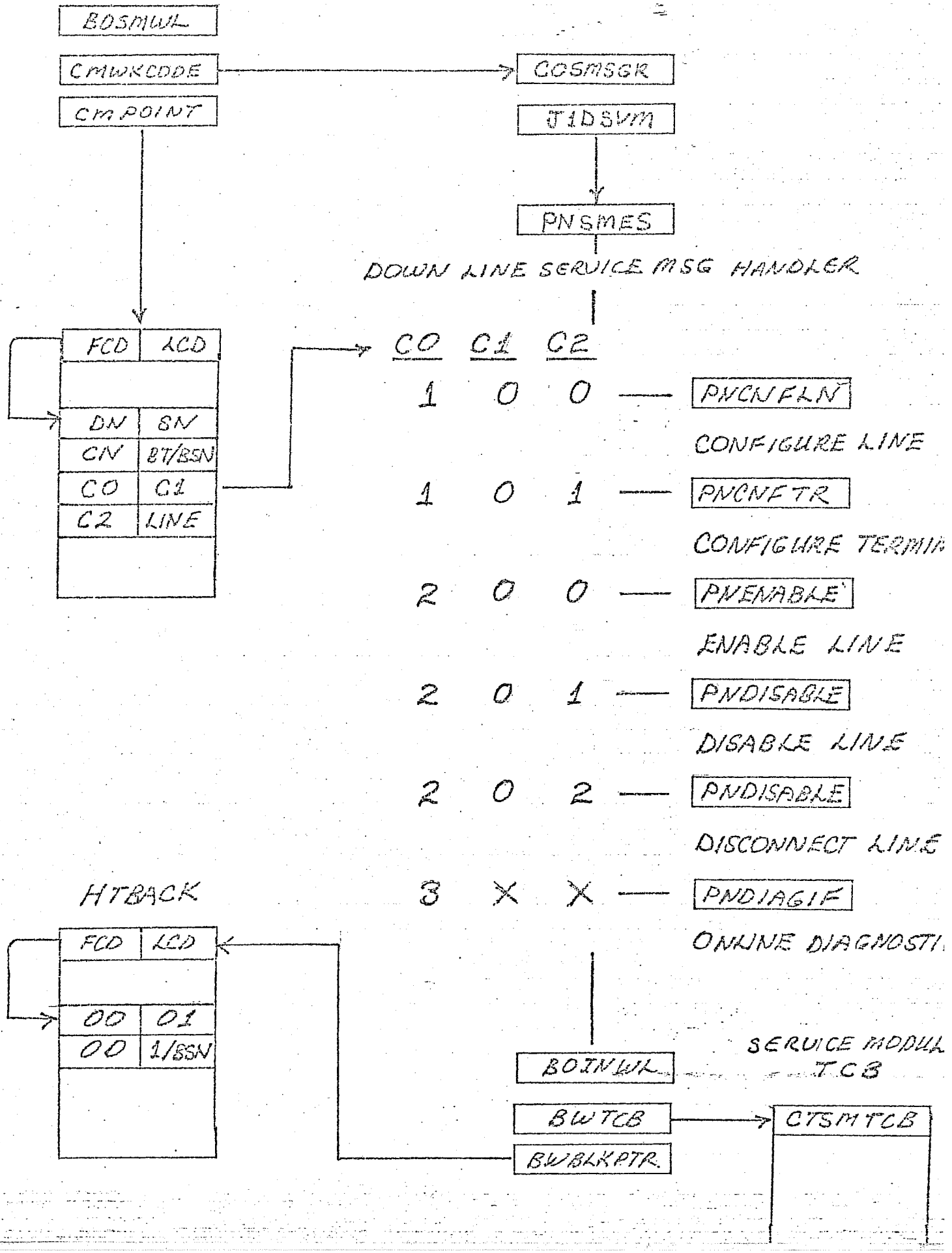
1

TWO FORMS

CMLINO – LINE NUMBER

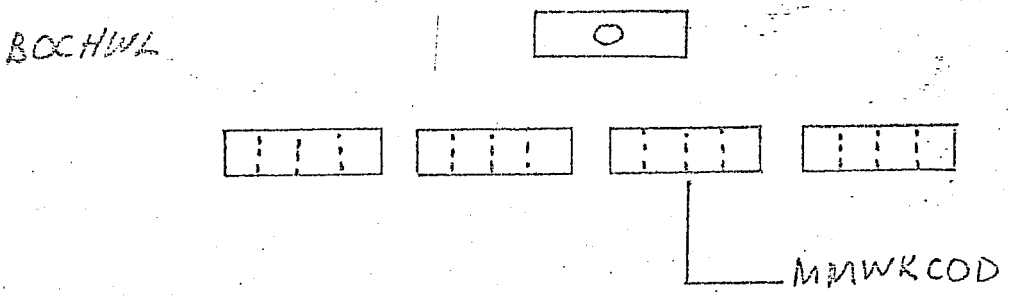
CMPOINT – BUFFER POINTER WORK CODES (CCOSMSGR...COCHL

SERVICE MODULE
DOWN LINE WORKLIST COMMAND



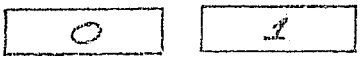
CONSOLE

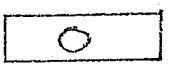
WORK LIST ENTRY : PBCONS



MMWKCOD — EVENT WORK CODE

BQ_INWL



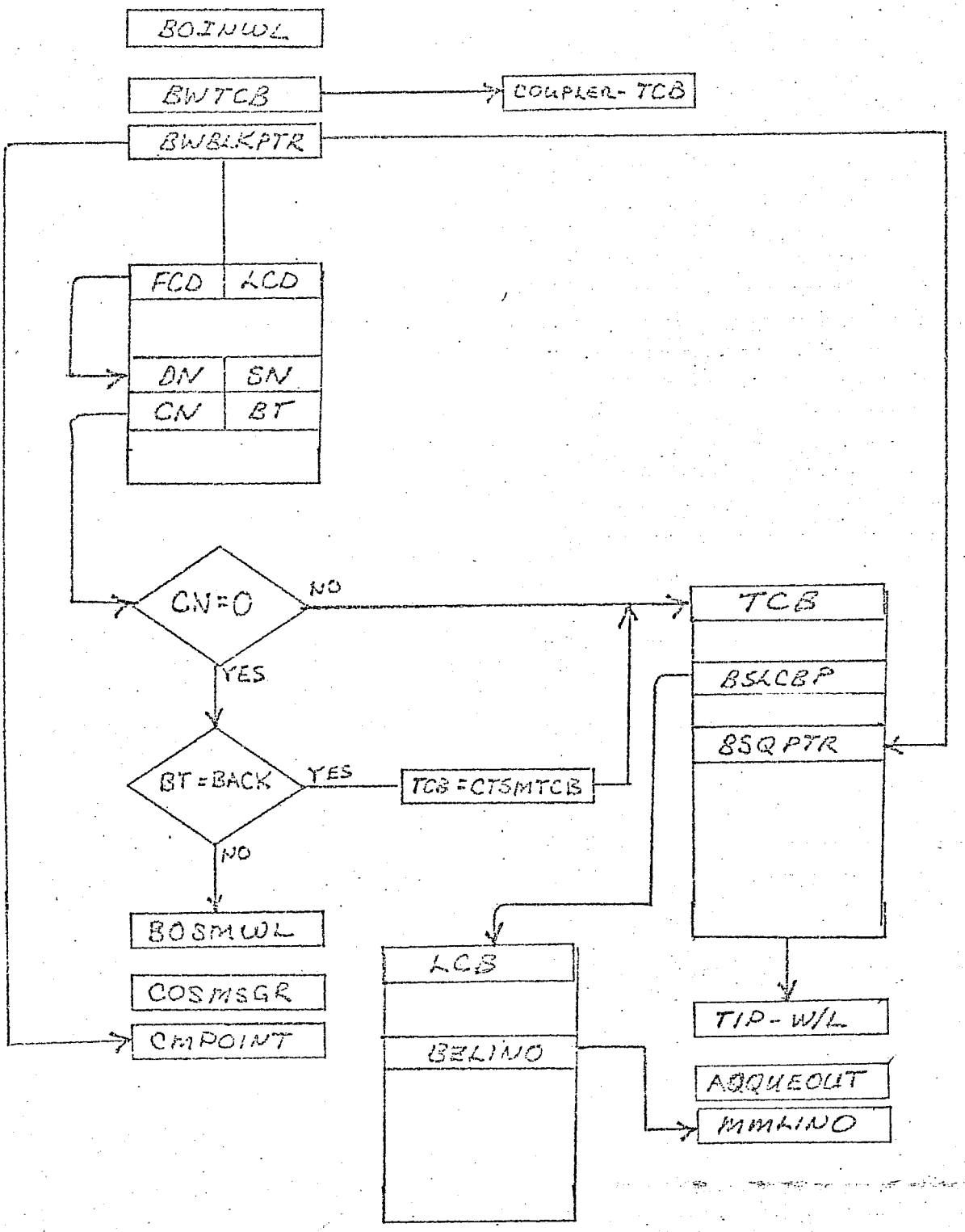


BWTCB – TCB POINTER

BWBLKPTR – DATA BLOCK POINTER

PBINTPROC

INTERNAL PROCESSING WORK LIST

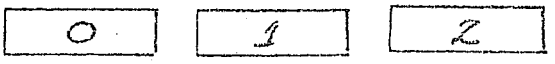


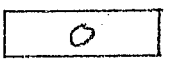
TTY TIP

WORK LIST ENTRY PTTYP

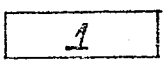
A12

BOTYWL

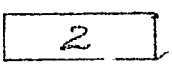




MMWKCOD – EVENT WORK CODE



MMLINO – LINE NUMBER



MMIBP – TCB OR DATA BUFFER POINTER

EVENT WORK CODES

21 = AOQUEOUT		OUTPUT QUEUED FOR LINE
22 = AOSMEN		ENABLE LINE
23 = AOSMDA		DISABLE LINE
24 = AOSMTCB		TCB BUILT
25 = AOBREAK		DOWN LINE BREAK
26 = AOTERM		DOWN LINE TERMINATE

MODE 4 TIP

WORK LIST ENTRY PTMOD4T AIA 9/29

BOM4WL | 0 | 1 | 2 |

MMWKCOD MMLINO MMIBP

0

MMWKCOD — EVENT WORK CODE

1

MMLINO — LINE NUMBER

2

MMIBP — TCB POINTER WHEN NEEDED OR DATA BUFFER

EVENT WORK CODES

21 - AOQUEOUT - OUTPUT QUEUED FROM PNDLBH

22 - AOSMEN - ENABLE LINE FROM SERVICE MODULE

23 - AOSMDA - DISABLE LINE FROM SERVICE MODULE

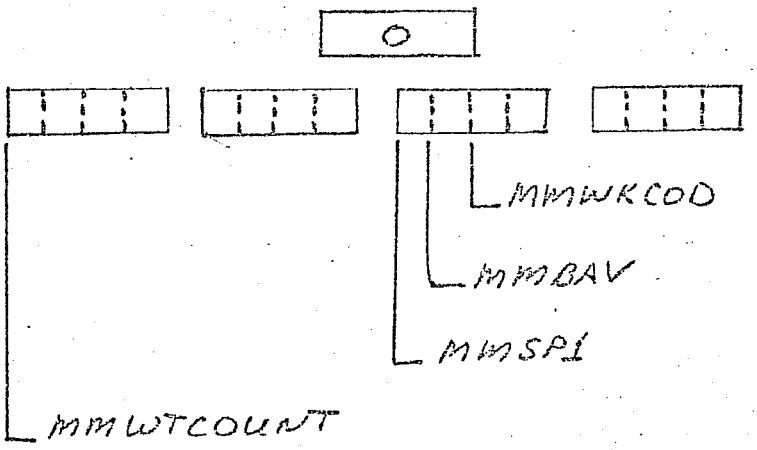
24 - AOSMTCB - TCB BUILT FROM SERVICE MODULE

CONTINUE

WORK LIST ENTRY PTGOOP

BOCWKQ | 0 | 1 | 2 | 3 |

0	1	2	3
MMWTCOUNT MMSP1 MMBAV MMWKCOD	MMLINO	MMRETADR	MMWTERR



MMWTCOUNT — WAIT COUNT

MMSP1 — NOT USED

MMBAV — BUFFER ADDRESS VALID FLAG

MMWKCOD — EVENT WORK CODE

1

MMLINO — LINE NUMBER

2

MMRETADDR — RETURN ADDRESS

3

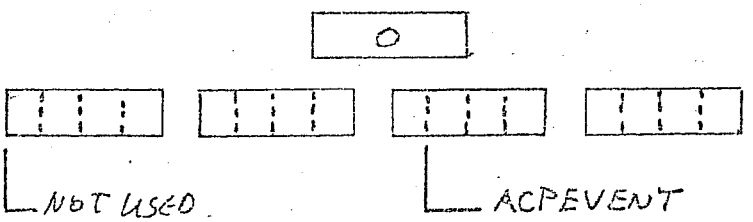
MMWTERR — ERROR FLAGS AT WAIT RETURN TIME
SEE LCB WORD 5 BZERROR SAME
FORMAT

COUPLER

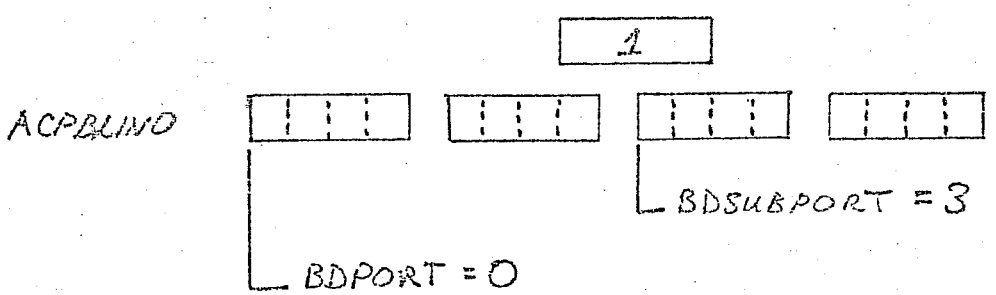
WORK LIST ENTRY PTSTAR

BDCØWL | 0 | 1 | 2 |

ACPEVENT ACPBUNO ACPBOBUF



ACPEVENT – EVENT WORK CØDE



2

ACPBOBUF – BUFFER POINTER NOT USED

BUWKENTRY

SOFTWARE PRIORITY WORK LIST ENTRY TABLE

JSWLAD = FWA PRIORITY-1 WORK LIST ENTRY (A7C)=98

EACH ENTRY 6 WORDS LONG AND CONTAINS LAST WORK LIST ENTRY THAT PRIORITY PROGRAM WAS QUITTING OR GETTING FROM A WORK LIST

SOFTWARE PRIORITY

1 = MEMPARITY, PROTECT, POWER FAIL PROGRAM

2 = MUX LOOP ERROR PROGRAM

MUXR 3 = MUX SUBSYSTEM PROGRAMS

5 = COUPLER NO-2 (2558-1)

6 = CONSOLE

CPL 7 = COUPLER NO-1

8 = TAPE CASSETTE

RTC 9 = REAL TIME CLOCK

13 = OUTPUT DATA DEMAND RECEIVED MLIA

14 = INPUT LINE FRAME RECEIVED MLIA

OPS 17 = OPS LEVEL PROGRAMS

PBTIMAL (CBITMTBL)

COTOPGMS

0	1	2
CBTIMER	CBINTVAL	CBADDR

CBTIMER - TIME REMAINING 1 = EXPIRED

CBINTVAL - TIME INTERVAL TO INITIALIZE CBTIMER

CBADDR - PROGRAM ENTRY ADDRESS WHEN TIMER EXPIRES

		TIME SEC
0 - COLCBTMSCN	PBLCBTMSCN	.5
1 - COADJUST	PBADJUST	.5
2 - COTUP	PBTUP	.5
3 - COTIMEOFDAY	PBTIMODAY	1.0
4 - COT1SEC	PMT1SEC	1.0
5 - COPSTAT	PNPSTAT	60.0
6 - COIOTMP	PBIOTMR	.5
7 - COCECNT	PNCECNT	10.0

EVERY .5 SEC WORK LIST ENTRY ENTERED INTO PBTIMAL WORKLIST BY LEVEL 8 PROCEDURE PBTIMER WHICH IS CALLED BY REAL TIME CLOCK INTERRUPT

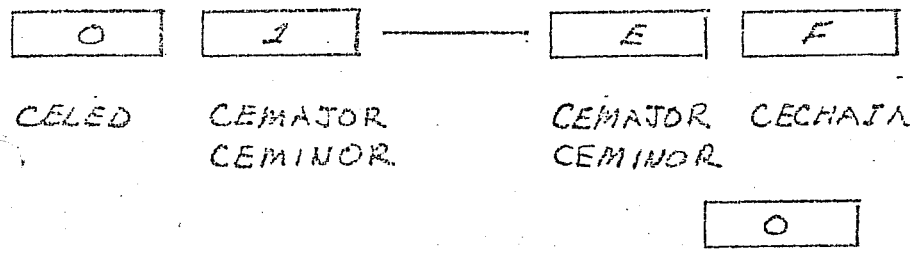
PBLCBTMSCN

RUNS EVERY .5 SEC CHECKS ACTIVE CONTROL BLOCK TABLE FOR PTOWAIT TIME EXPIRED. CALLED FROM PBTIMAL USING CBLTMTBL ENTRY 0

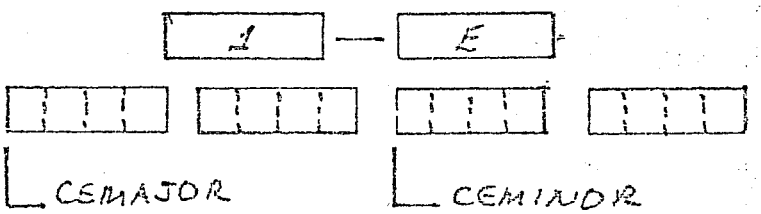
ENTRY

CELCBACT – POINTER TO LIST OF ACTIVE LCB

LCBLST – ACTIVE LCB LIST

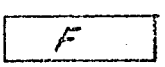


CELED – LAST ENTRY DISPLACEMENT



CEMAJOR – INDEX INTO LCB ARRAY

CEMINOR – INDEX INTO TDM LCB ARRAY



CECHAIN – LINK TO NEXT BUFFER IN LCBLST

SETSDATA IN LCB

BEPDMO = 0 CLA ERROR COUNT

BELTIMER = BELTIMER – 1 WAIT TIME COUNTER

IF BELTIMER = 0

BELINSTA.HTWAIT = FALSE	WAIT ACTIVE
BELINSTA.HTTRUN = FALSE	WAIT TIMMER RUNNING
BEERAC = BEERAC + BEERROR	ACCUMULATED CLA ERRORS
BEERROR = 0	CLA ERRORS IN LAST .5 SEC
BEWAITMASK = 0	WAIT EVENT
BERETADDR = 0	RETURN ADDRESS

PBLCBTMSCN

EXIT

BBLINSTA.HTLEVEL = HLOPS EXIT TO SRET

BBLINSTA.HTLEVCA = HLMLCA ENTER W/L MMEWULQ

MMWOTERR = BBERROR
MMBAV = FALST
MMWTCOUNT = BBWTCOUNT
MMWORCOD = ADCONTINUE
MMRETADDR = BBRETADDR

P1 = SOFTWARE LEVEL CONTROL IS TO BE RETURNED TO

P2 = EVENT OR EVENTS TO WAIT FOR

P3 = TIME TO WAIT * .5 SECONDS

SETS DATA IN LCB

BEWTCOUNT = BEWTCOUNT +1 — WAIT COUNT

BELTIMER = P3 — WAIT TIME

BELIWSTA.HTTRUN = TRUE — WAIT TIMMER RUNNING

BELIWSTA.HTWAIT = TRUE — WAIT ACTIVE

BELIWSTA.HTLEVEL = P1 — SOFTWARE LEVEL HLOPS+HL

BEWAITMASK = P2 — WAIT EVENT CODE

BER1SAVE = R1 — R1 SAVE

BER2SAVE = R2 — R2 SAVE

BERETADDR = RETURN ADDRESS

EXIT

P1 = HLOPS RETURN SRETOPS TO MONITOR

P1 = HLMUX RETURN SRETMUX TO PMWORP

WORKNST ENTERED

P3 = -1 WL = BOT200 — MMWKCOD = ADDLON

MMLINO = BDLINO

MMWTCOUNT = BEWTCOUNT

P2 = BEWSTAT EVENT ALREADDY OCCURRED

WL = BOCWLO — MMBAV = FALSE

MMWTCOUNT = BEWTCOUNT

MMLINO = BDLINO

MMWKCOD = ADCONTINUE

MMRETADR = BERETADDR

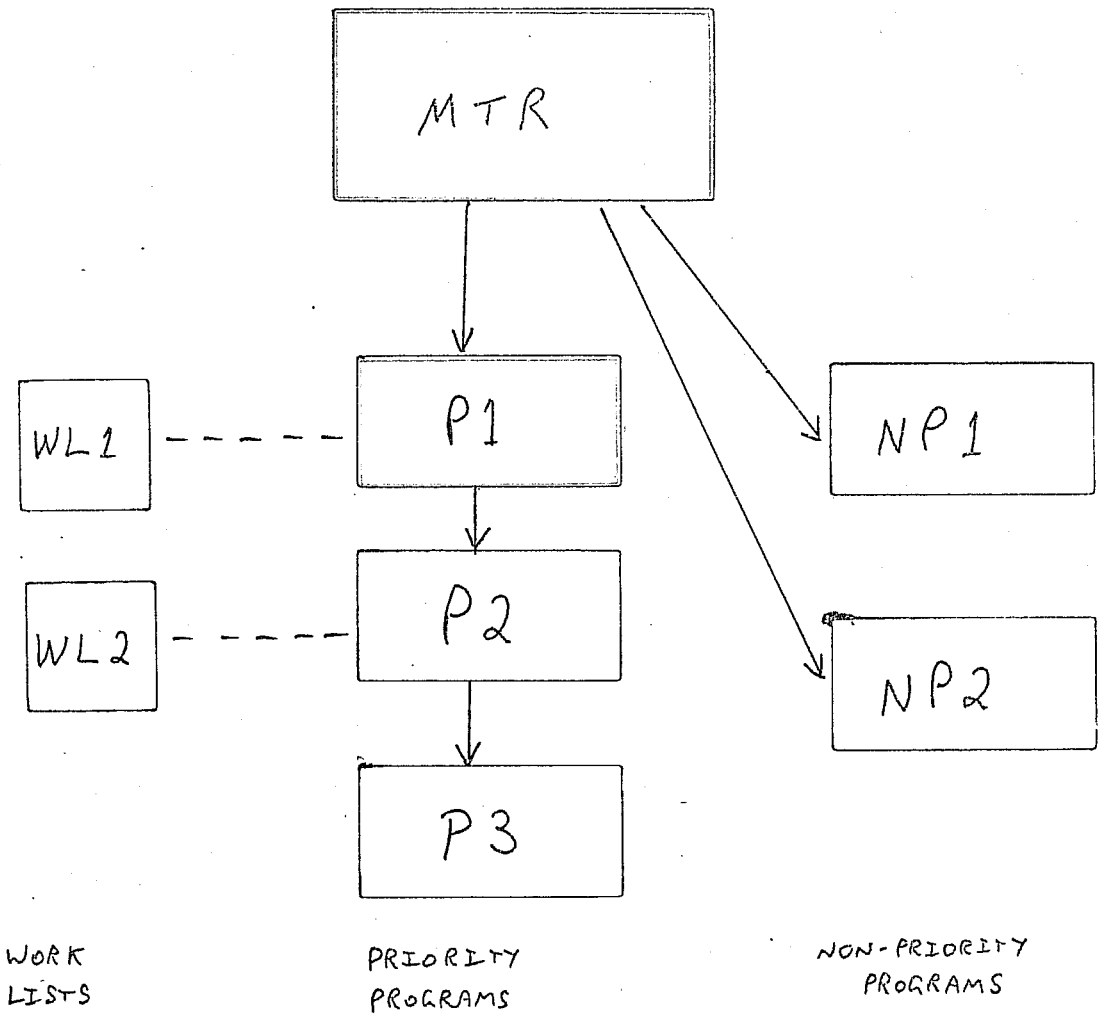
00	80	NULL	NO-OP
01	81	SOH	START OF HEADING
02	82	STX	START OF TEXT
03	83	ETX	END OF TEXT
04	84	EOT	END OF TRANSMISSION
05	85	ENQ	ENQUIRY
06	86	ACK	ACKNOWLEDGE
07	87	BEL	ALERT
08	88	BS←	BACKSPACE
09	89	HT	HORIZONTAL TAB
0A	8A	LF↓	LINE FEED ADVANCE CURSOR ONE LINE DOWN
0B	8B	VT	VERTICAL TAB
0C	8C	FF	FORM FEED
0D	8D	RETURN	RESET CURSOR TO BEGINNING OF LINE
0E	8E	INVERSE	START INVERSE VIDIO
0F	8F	END INVERSE	END INVERSE VIDIO
10	90	DLE	DATA LINK ESCAPE
11	91	DC1	DEVICE CONTROL 1
12	92	DC2	DEVICE CONTROL 2
13	93	DC3	DEVICE CONTROL 3
14	94	DC4	DEVICE CONTROL 4
15	95	SKIP→	ADVANCE CURSOR ONE CHAR
16	96	LINE CLEAR	ENTER SPACE TO END OF LINE
17	97	EBT	END OF TRANSMISSION BLOCK
18	98	CLEAR	ENTER SPACE IN ENTIRE DISPLAY
19	99	RESET	RESET CURSOR TO FIRST CHAR IN DISPLAY
1A	9A	CURSOR↑	MOVE CURSOR UP ONE LINE
1B	9B	ESC	ESCAPE
1C	9C	FS	FILE SEPARATOR
1D	9D	GS	GROUP SEPARATOR
1E	9E	RS	RECORD SEPARATOR
1F	9F	US	UNIT SEPARATOR

20	A0	SPACE	40	C0	@	60	E0	`	74	F4
21	A1	!	41	C1	A	61	E1	a	75	F5
22	A2	"	42	C2	B	62	E2	b	76	F6
23	A3	#	43	C3	C	63	E3	c	77	F7
24	A4	$	44	C4	D	64	E4	d	78	F8
25	A5	%	45	C5	E	65	E5	e	79	F9
26	A6	&	46	C6	F	66	E6	f	7A	FA
27	A7	'	47	C7	G	67	E7	g	7B	FB
28	A8	(	48	C8	H	68	E8	h	7C	FC
29	A9	)	49	C9	I	69	E9	i	7D	FD
2A	AA	*	4A	CA	J	6A	EA	j	7E	FE
2B	AB	+	4B	CB	K	6B	EB	k	7F	FF
2C	AC	,	4C	CC	L	6C	EC	l		
2D	AD	-	4D	CD	M	6D	ED	m		
2E	AE	.	4E	CE	N	6E	EE	n		
2F	AF	/	4F	CF	O	6F	EF	o		
30	B0	0	50	D0	P	70	F0	p		
31	B1	1	51	D1	Q	71	F1	q		
32	B2	2	52	D2	R	72	F2	r		
33	B3	3	53	D3	S	73	F3	s		
34	B4	4	54	D4	T					
35	B5	5	55	D5	U					
36	B6	6	56	D6	V					
37	B7	7	57	D7	W					
38	B8	8	58	D8	X					
39	B9	9	59	D9	Y					
3A	BA	:	5A	DA	Z					
3B	BB	;	5B	DB	[					
3C	BC	<	5C	DC	\					
3D	BD	=	5D	DD	]					
3E	BE	>	5E	DE	^					
3F	BF	?	5F	DF	_					

20	A0	–
21	A1	J
22	A2	K
23	A3	L
#4	A4	M
25	A5	N
26	A6	Ø
27	A7	P
28	A8	Q
29	A9	R
2A	AA	V
2B	AB	$
2C	AC	*
2D	AD	↑
2E	AE	↓
2F	AF	>
30	B0	+
31	B1	A
32	B2	B
33	B3	C
34	B4	D
35	B5	E
36	B6	F
37	B7	G
38	B8	H
39	B9	I
3A	BA	<
3B	BB	.
3C	BC	)
3D	BD	≥
3E	BE	ESCAPE
3F	BF	;
40	C0	
41	C1	1
42	C2	2
43	C3	3
44	C4	4
45	C5	5
46	C6	6
47	C7	7
48	C8	8
49	C9	9
4A	CA	0
4B	CB	=
4C	CC	≠
4D	CD	≤
4E	CE	%
4F	CF	[
50	D0	SPACE
51	D1	/
52	D2	S
53	D3	T
54	D4	U
55	D5	V
56	D6	W
57	D7	X
58	D8	Y
59	D9	Z
5A	DA	]
5B	DB	↓
5C	DC	⊆
5D	DD	⊢
5E	DE	≡
5F	DF	^

9B	ESCAPE
01	SOH
83	EOT
70	SITE ADDRESS
E0	STATION ADDRESS
98	REJECT
93	READ
86	ACK

20	A0	SPACE
21	A1	!
22	A2	"
23	A3	#
24	A4	$
25	A5	%
26	A6	&
27	A7	'
28	A8	(
29	A9	)
2A	AA	*
2B	AB	+
2C	AC	,
2D	AD	-
2E	AE	.
2F	AF	/
30	B0	0
31	B1	1
32	B2	2
33	B3	3
34	B4	4
35	B5	5
36	B6	6
37	B7	7
38	B8	8
39	B9	9
3A	BA	:
3B	BB	;
3C	BC	<
3D	BD	=
3E	BE	>
3F	BF	?
40	C0	@
41	C1	A
42	C2	B
43	C3	C
44	C4	D
45	C5	E
46	C6	F
47	C7	G
48	C8	H
49	C9	I
4A	CA	J
4B	CB	K
4C	CC	L
4D	CD	M
4E	CE	N
4F	CF	O
50	D0	P
51	D1	Q
53	D3	S
54	D4	T
55	D5	U
56	D6	V
57	D7	W
58	D8	X
59	D9	Y
5A	DA	Z
5B	DB	[
5C	DC	\
5D	DD	]
5E	DE	^
5F	DF	_
61	E1	a
62	E2	b
63	E3	c
64	E4	d
65	E5	e
66	E6	f
67	E7	g
68	E8	h
69	E9	i
6A	EA	j
6B	EB	k
6C	EC	l
6D	ED	m
6E	EE	n
6F	EF	o
70	F0	p
71	F1	q
72	F2	r
73	F3	s
74	F4	t
75	F5	u
76	F6	v
77	F7	w
78	F8	x
79	F9	y
7A	FA	z
7B	FB	{
7C	FC	!
7D	FD	}
7E	FE	~
7F	FF	▯
9B		ESCAPE

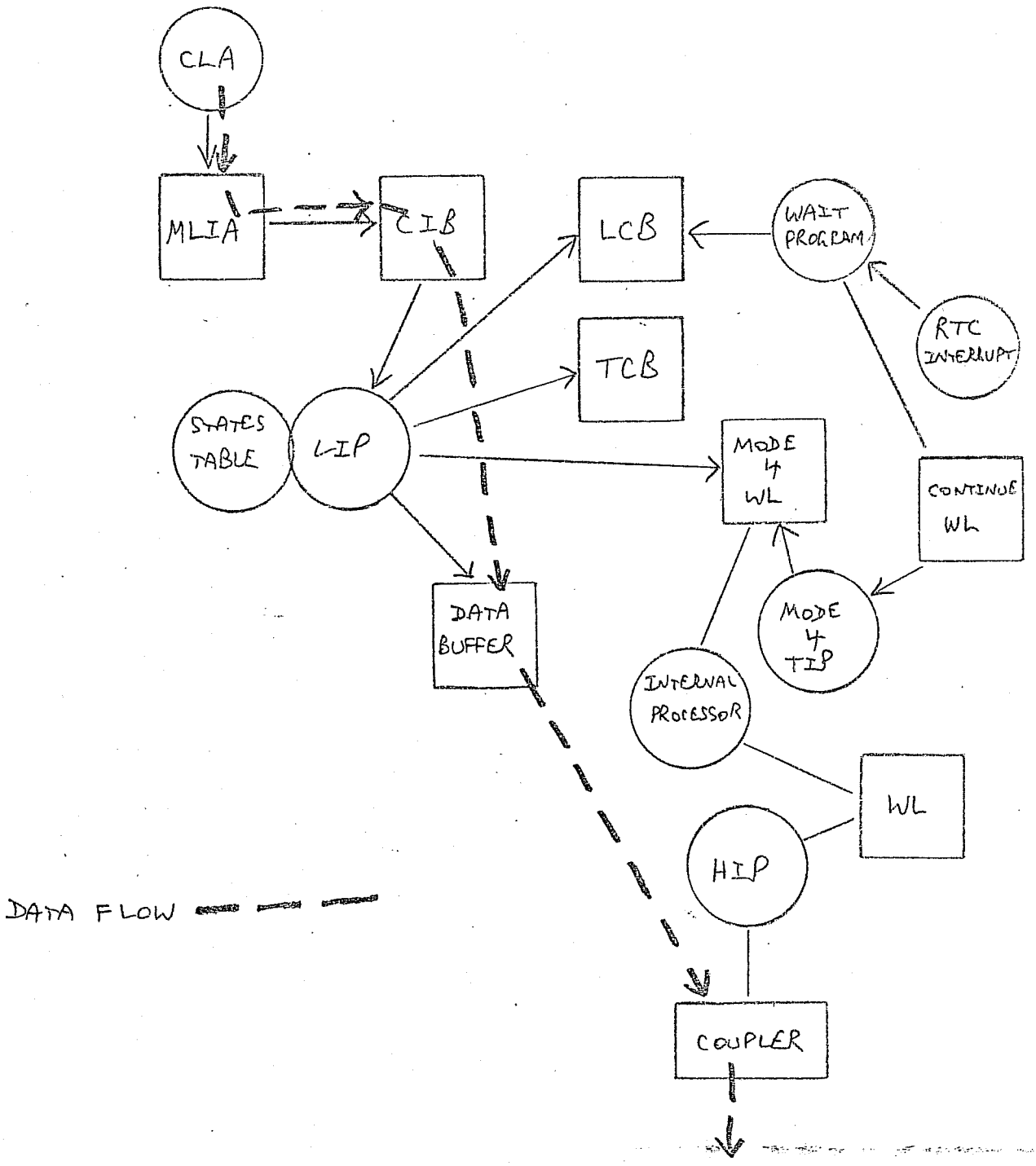


MONITOR LOOP EXECUTES EACH PRIORITY PROGRAM THEN EXECUTES ONE NON-PRIORITY PROGRAM. NEXT LOOP EXECUTES EACH PRIORITY PROGRAM THEN NEXT NON-PRIORITY PROGRAM.

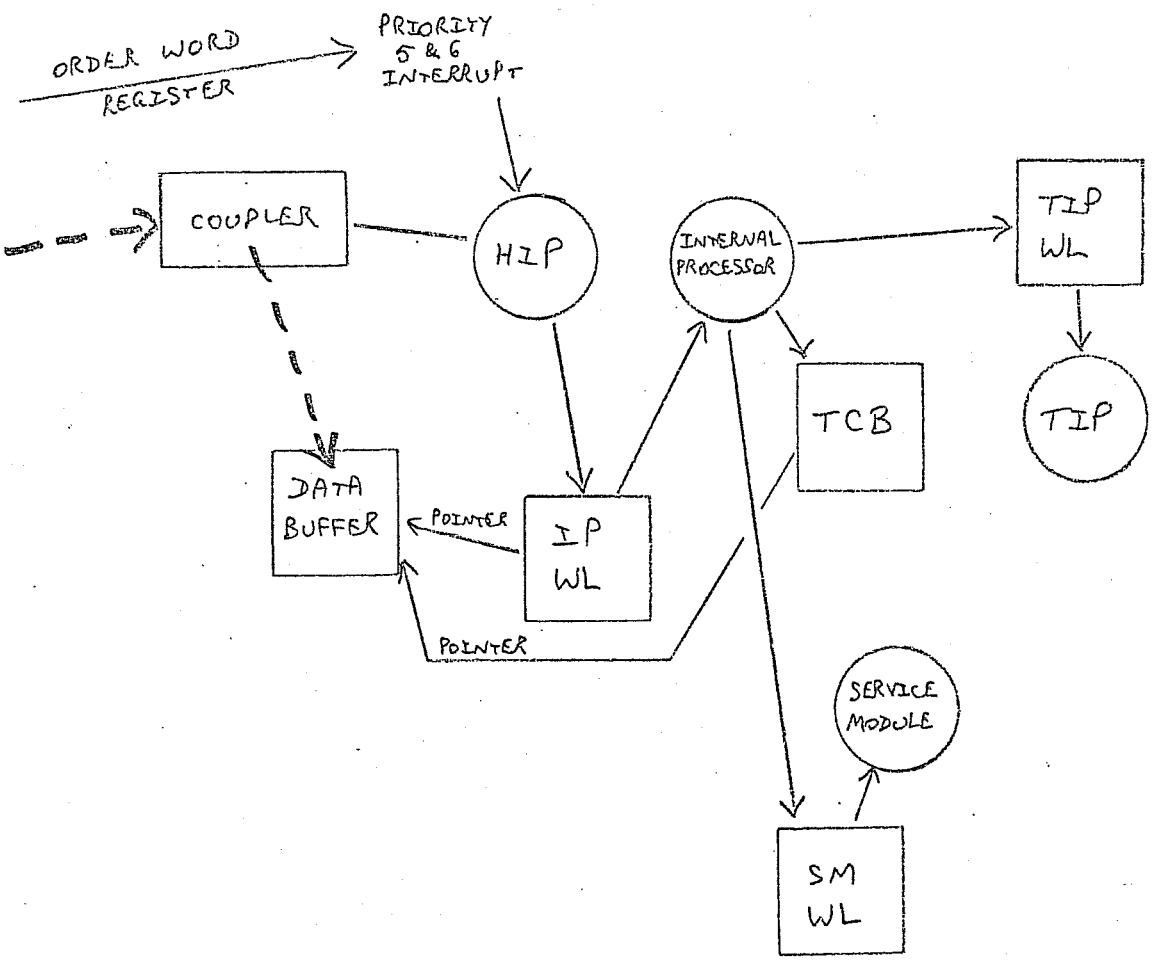
EACH PROGRAM MAY REQUIRE SERVICE BY OTHER ROUTINES — IF SO IT PUTS AN ENTRY INTO THE OTHER ROUTINE'S WORKLIST AND THAT REQUEST IS EXECUTED NEXT TIME THAT PROGRAM IS ENTERED.

GENERAL OVERVIEW

INPUT FROM TERMINAL.



OUTPUT FROM HOST

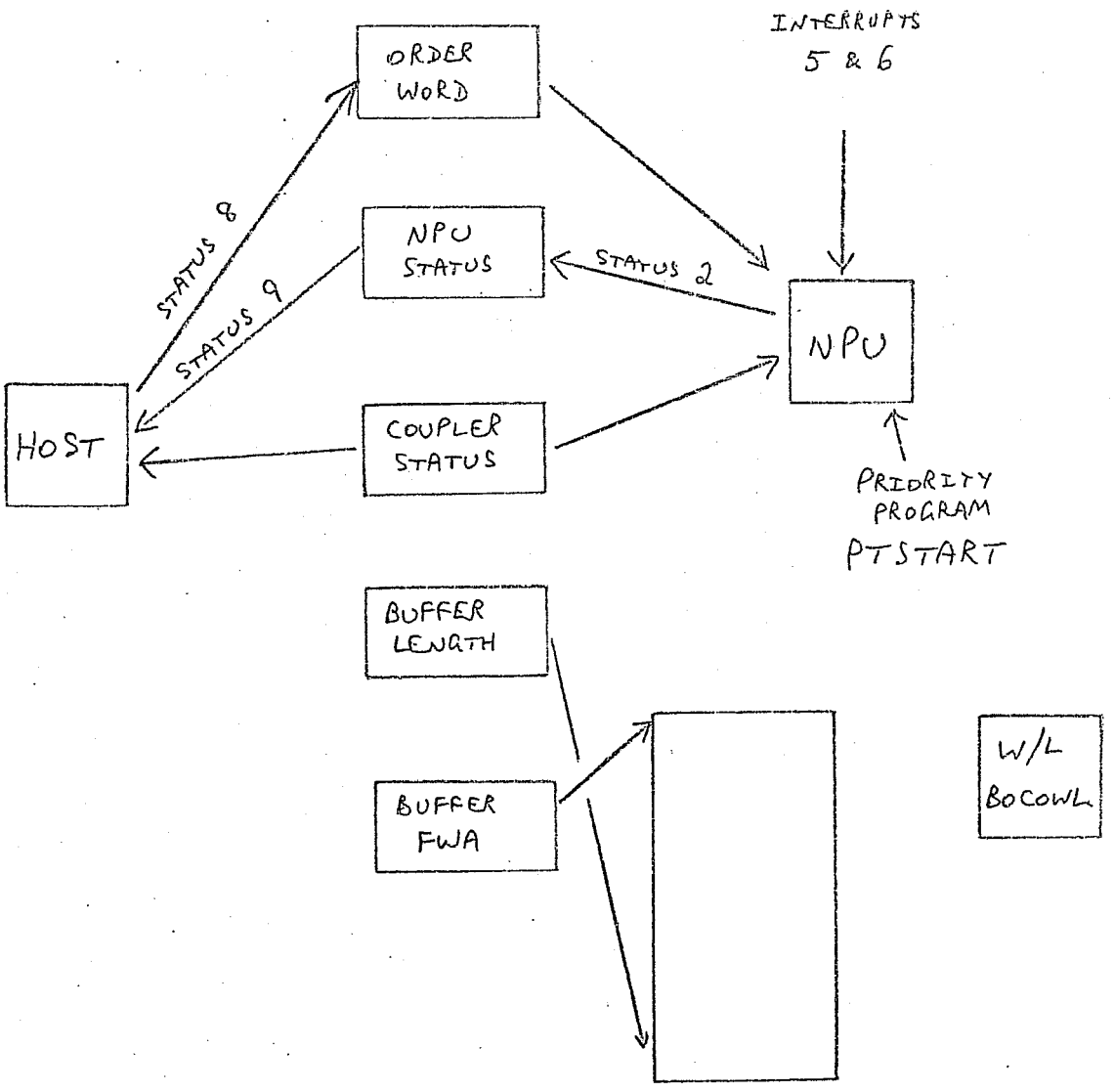


ANY DATA BUFFERS WITH CONNECTION NUMBERS OF ZERO ARE FOR SERVICE MODULE (I.E. COMMANDS FOR 2550 ITSELF RATHER THAN FOR A TERMINAL).

OUTPUT FROM HOST

- CYBER FUNCTIONS COUPLER WITH 'DATA READY'.
- INTERRUPT TO HIP - WHICH SETS UP BUFFER.
- HIP SETS COUPLER STATUS 'READY'.
- CYBER TRANSFERS DATA VIA COUPLER.
- INTERRUPT TO HIP ON END OF DATA.
- HIP SETS ENTRY IN INTERNAL PROCESSOR WL.
- IP EXAMINES BUFFER AND DETERMINES TIP TYPE, VIA CONNECTION NUMBER.
- IP SETS RELEVANT TIP (OR SERVICE MODULE) WORKLIST ENTRY.
- TIP OR SM PROCESSES DATA BUFFER WHEN ACTIVATED BY WL ENTRY.

COUPLER - NPU COMMUNICATION



BREAKPOINT

WORD 16B POINTS TO TABLE USED FOR WRAP-AROUND-SNAP (WRAPSNAP TABLE).
TABLE IS 9 WORDS

1	FWA DUMP BUFFER
2	FWA DUMP BUFFER
3	LWA DUMP BUFFER
4-5	AREA TO DUMP
6-7	"
8-9	"

ELEMENTS OF A PASCAL PROGRAM

- LABEL Declaration
- CONST Definition
- TYPE Definition
- VAR Declaration
- VALUE Declaration
- Procedure & Function Declarations
- Statement Part

LABEL Declaration

```
LABEL
  10,20,30;
```

CONST Definition

```
CONST
  K1 = 1;
  K10 = 10;
  MAXCNT = 10000B;
```

TYPE Definition

```
TYPE
  T1 = PACKED ARRAY[1..10] OF CHAR;
  T2 = 0..15;
```

VAR Declaration

```
VAR
  A : INTEGER;
  B : T1;
  C : RECORD
        F1 : T2;
        F2 : CHAR;
        F3 : ARRAY[0..5] OF INTEGER;
      END;
```

VALUE Declaration

```
VALUE
  A = 100;
  B = [=ABCDEFGHIJ=];
```

Procedure & Function Declaration

```
PROCEDURE TEST(X,Y : INTEGER;
               VAR Z : INTEGER);
  BEGIN
    Z := X + Y;
  END;
```

Statement Part

```
BEGIN
  Statement list
END.
```

ARRAYS

```
VAR
  AREA : ARRAY[1..5,1..10] OF INTEGER;
  I : INTEGER;

BEGIN
  I := AREA[2,7] + 10;
  AREA[2*I-9,3] := 32000;
END;
```

RECORDS

```
VAR
  INREC : RECORD
            FLD1 : INTEGER;
            FLD2 : 0..20;
            FLD3 : RECORD
                     XRAY : -10..10;
                     POSTVE : BOOLEAN;
                     IND : CHAR;
                   END;
            FLD4 : CHAR;
          END;

BEGIN
  INREC.FLD2 := 19;
  INREC.FLD3.IND := 'W';
END;
```

RECORD VARIANTS

```
TYPE
   T1 = RECORD
          F1 : CHAR;
          F2 : INTEGER;
            CASE  F3 : INTEGER OF
             0 :  {CNTX : INTEGER;
                   AREAX : ARRAY[1..5] OF CHAR};
             1 :  {FLAG1 : BOOLEAN};
             2 :  {B2 : -10..10};
             3 :  {TABLE : ARRAY[1..10,1..10] OF INTEGER};
        END;

VAR
   A : T1;
   B : T1;

BEGIN
   A.CNTX := 10;
   B.FLAG1 := TRUE;
END;
```

SETS

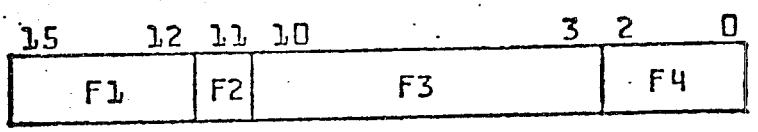
```
TYPE
  T1 = (E1,E2,E3,E4,E5,E6,E7,E8,E9,
        E10,E11,E12,E13,E14,E15,E16);
  T2 = SET OF T1;
  T3 = 0..15;

VAR
  SET1 : T2;
  SET2 : T2;
  SET3 : SET OF T3;

BEGIN
  SET1 := SET1 ∧ SET2;
  SET1 := SET1 ∨ (E1,E5,E8);
  IF E1 IN SET1
    THEN SET2 := SET1;
  SET3 := (5,9,12,14);
END;
```

BIT FIELDS -

```
PCKDWD : PACKED RECORD
            F1 : 0..15;
            F2 : BOOLEAN;
            F3 : 0..255;
            F4 : (A,B,C,D,E)
         END;
```



DYNAMIC VARIABLES

```
TYPE
  PTR = ↑LNKLST;
  LNKLST = RECORD
             LNKPTR : PTR;
             A : INTEGER
           END;

VAR
  PTRV1 : PTR;
  PTRV2 : PTR;
  I : INTEGER;

BEGIN
  PTRV2 := NIL;
  FOR I := 1 TO 10 DO
    BEGIN
      NEW(PTRV1);
      PTRV1↑.LNKPTR := PTRV2;
      PTRV1↑.A := 0;
      PTRV2 := PTRV1;
    END;
END;
```

OPERATORS

Arithmetic

+	addition
-	subtraction
*	multiplication
div	division
mod	modulus

Relational {arithmetic}

=	equal
≠	not equal
<	less than
≤	less than or equal
>	greater than
≥	greater than or equal

Boolean

¬	not
V	or
Λ	and

Set

V	union
Λ	intersection
-	difference

Relational {set}

=	equal
≠	not equal
≤	set inclusion
≥	set inclusion
in	membership

STATEMENTS

Assignment statement

```
I := J + 1;
```

Procedure statement

```
PROC1;
PROC2(X,Y,Z);
```

Compound statement

```
BEGIN
  I := J;
  PROC1
END;
```

IF statement

```
IF I = J
  THEN PROC1
  ELSE PROC2(A,B,C);
```

CASE statement

```
CASE NODE OF
  1 : BRNCH := 1;
  2 : BRNCH := 2;
  3 : BRNCH := 3
END;
```

REPEAT statement

```
REPEAT
  I := J + 1;
  PROC1
UNTIL I > 10;
```

WHILE statement

```
WHILE I ≤ 10 DO
  BEGIN
    I := J + 1;
    PROC1
  END;
```

FOR statement

```
FOR I := 1 TO 10 DO
  BEGIN
    J := I * 100;
    PROC1;
    K := J - 3
  END;
```

GOTO Statement

```
PROCEDURE A;
   LABEL 10;
   PROCEDURE B;
        BEGIN
        GOTO 10;
     10:GOTO EXIT10;
        END
   BEGIN
 10:Statement;
   END
```

WITH Statement

```
WITH LCB[A] DO
  BEGIN
    F1:=2;
    F2:=FALSE
  END
    equivalent to
  LCB[A].FA:=2;
  LCB[A].F2:=FALSE;
```

EMPTY Statement

```
GO TO 10;
10:;
```