UNISYS

V500 SYSTEM MAINTENANCE CONTROLLER

Entry/ Medium Systems Group
PASADENA DEVELOPMENT CENTER

UNISYS RESTRICTED

ENGINEERING DESIGN SPECIFICATION

REVISIONS

REV LTR	REVISION ISSUE DATE	PAGES REVISED. ADDED. DELETED OR CHANGE OF CLASSIFICATION	PREPARED BY	APPROVED BY
A	ECN 51597	Initial Issue 7-16-87	L. Simpson L. Simpson J. Toma	J. Toma J. Toma R. Young R. Young 5/11/87 S. Davis S. Davis 5/19/87 A. Gomez 6/11/87 Alejandro Gomez D. Welbourn D. Welbourn 7/2/87 H. Li H. Li 7/2/87
B	ECN 52093-A	Revised Appendix B - SMC Diagnostics Error Reporting Per REE 100264.	L. Simpson L. Simpson	I. Nasser I. Nasser 4/5/88 A. Gomez Alejandro Gomez 4/5/88 H. Li H. Li 4/8/88

TABLE OF CONTENTS

1	SCOPE.	PAGE 4
2	RELATED DOCUMENTS.	PAGE 4
3	GENERAL DESCRIPTION.	PAGE 5
4	SMC BOARD INTERFACES	PAGE 7
5	ORGANIZATION AND FUNCTIONAL BLOCKS	PAGE 8
5.1	MICROPROCESSOR SECTION	PAGE 8
5.1.1	PERIPHERAL INTERFACES.	PAGE 12
5.1.2	MAINTENANCE PANEL USE.	PAGE 12
5.2	I/O BUS SECTION.	PAGE 16
5.2.1	BUS TRANSACTIONS	PAGE 16
5.2.2	80186 INTERFACE WITH SMC I/O BUS	PAGE 17
5.2.3	BUFFCMND REGISTER (61Eh)	PAGE 17
5.2.4	BUFFSTAT REGISTER (61Ah)	PAGE 18
5.2.5	BUFFLEN REGISTER (616h).	PAGE 18
5.2.6	BUFFADR REGISTER (614h).	PAGE 19
5.2.7	ABUSSTAT REGISTER (618h)	PAGE 19
5.2.8	ABUSCMND REGISTER (61Ch)	PAGE 20
5.3	TEST BUS SECTION	PAGE 21
5.4	PROGRAM MEMORY ERROR INTERFACE	PAGE 23
5.5	CLOCK AND SHIFT CONTROL.	PAGE 24
5.5.1	SHIFT ARRAY AND THE ECL DISCRETE CONTROL LOGIC	PAGE 30
5.6	CLOCK CONTROL & DISTRIBUTION ARRAY	PAGE 35
5.6.1	OPERATIONAL PRAGMATICS	PAGE 35
5.7	MAINTENANCE PROCESSOR COMMUNICATION.	PAGE 37
5.8	SMC ON-BOARD PROGRAM MEMORY.	PAGE 40
6	SMC SELF-MAINTENANCE	PAGE 41
6.1	SWITCHES	PAGE 41
6.2	THE MONITOR REGISTER	PAGE 41
6.3	SMC FAULT DETECTION.	PAGE 42
6.3.1	INTERFACE FAULTS	PAGE 42
6.3.2	INTERNAL FAULTS.	PAGE 42
6.3.3	SMC ERROR REPORTING.	PAGE 42
7	CLOCK RECEIVER AND MAINTENANCE ARRAY	PAGE 43
7.1	MAINTENANCE SIGNALS.	PAGE 43
7.2	UNINITIALIZED HIDDEN STATE ERRORS.	PAGE 44
8	ECCIO GATE ARRAY	PAGE 44
8.1	ECCIO ARRAY SIGNAL DEFINITIONS	PAGE 45
9	MAINTENANCE BUS INTERFACE DIAGRAMS	PAGE 46
9.1	SMC TO PROCESSOR/MODULE/CARD INTERFACE	PAGE 46
9.2	PROCESSOR MAINTENANCE GROUP HIERARCHY.	PAGE 47
9.3	I/O BUS HAMMING CODE	PAGE 48
9.4	I/O BUS ERROR DETECTION DECODE	PAGE 49

TABLE OF CONTENTS

9.5 TIMING DIAGRAMS. . . . . PAGE 50
9.5.1 SHIFTING A CHAIN . . . . . PAGE 50
9.5.2 CLEARING A CHAIN . . . . . PAGE 50
9.5.3 WRITING CONTROL STORES . . . . . PAGE 51
9.5.4 READING RAM. . . . . PAGE 52
9.5.5 WRITING SCRATCHPAD TYPE RAMS . . . . . PAGE 52
9.5.6 ERRORS AND NON-ERRORS. . . . . PAGE 53
9.6 PHASE DETECTION. . . . . PAGE 54
9.7 FAULT TOLERANT SYSTEM INTERCONNECT . . . . . PAGE 61
APPENDIX A MENTOR SIMULATIONS . . . . . PAGE 62
APPENDIX B SMC DIAGNOSTICS ERROR REPORTING. . . . . PAGE 82

TABLE OF ILLUSTRATIONS

Figure 1	SMC Block Diagram	PAGE 6
Figure 2	Microprocessor-Memory Interface	PAGE 9
Figure 3	80186 Internal Block Diagram.	PAGE 10
Figure 4	SMC LED Definitions	PAGE 13
Figure 5	SMC Diagnostic Software Flowchart	PAGE 15
Figure 6	8251 USART Block Diagram.	PAGE 22
Figure 7	Shift Array Timing.	PAGE 25
Figure 8	Shift Array Error Gates	PAGE 27
Figure 9	Shift Array Stop Logic.	PAGE 28
Figure 10	Shift Array Logic Diagram	PAGE 30
Figure 11	Shift Array Bit Assignments	PAGE 31
Figure 12	SMC - Maintenance Bus Interface	PAGE 36
Figure 13	MPCC Block Diagram.	PAGE 38
Figure A1	Clock Simulation (Plan C Count 2)	PAGE 62
Figure A2	Clock Simulation (Plan D Count 2)	PAGE 64
Figure A3	Clock Simulation (Plan B Count 3)	PAGE 66
Figure A4	Clock Simulation (Plan A Count 3)	PAGE 68
Figure A5	Clock Simulation (Plan A Count 1)	PAGE 70
Figure A6	Clock Simulation (Clocks On At T0).	PAGE 72
Figure A7	Clock Simulation (Toggle Stopping Clocks)	PAGE 74
Figure A8	Clock Simulation (Toggle Starting Clocks)	PAGE 76
Figure A9	Clock Simulation (Stop Clocks).	PAGE 78
Figure A10	Clock Simulation (Path Check)	PAGE 80

1 SCOPE

This document defines the System Maintenance Controller for use in the V500 Processor Complex.

2 RELATED DOCUMENTS

1257 6005	Corporate Environmental Standards
A1, A14, A16	Corporate Technical Standards
1993 5279	V500 Architecture
1993 5162	V500 System
1997 5390	V Series Instruction Set
1993 5204	V500 Execute Module
1993 5212	V500 Fetch Module
1993 5220	V500 Memory Control and Cache Module
1993 5253	I/O Memory Concentrator
1993 5329	V500 Data Transfer Module
1993 5337	Fault Detection
1998 5332	V500 Clock Subsystem
1993 5303	V500 Maintenance Subsystem
2323 7431	Common I/O Maintenance
2360 1016	LCP Test Bus on Universal Console
1996 1457	CLKMNT3 Specification
1996 0707	Shift Array Specification

3 GENERAL DESCRIPTION

The V500 System Maintenance Controller (SMC) is a one-card module that connects the Maintenance Processor (MP) to the V500 processor, memory, I/O subsystems, Environmental Control Module (ECM), and the DLP test bus. It provides the V500 processor complex with an interface to the MP when the MP is used as a peripheral device.

There are four main processor-SMC interface sections; one input, one output, and two I/Os. They are:

Stop and error event detection (input)
Clock and shift control signals (output)
Shift chains (I/O)
I/O bus to memory (I/O)

The shift chains provide a moderate speed method of communication; the I/O bus provides a faster path to and from memory. Note that there is one SMC board per cabinet, and it has identical interfaces to both processors in the cabinet.

Following is a block diagram of the SMC board. The board is split into two regions, based on the type of technology used.

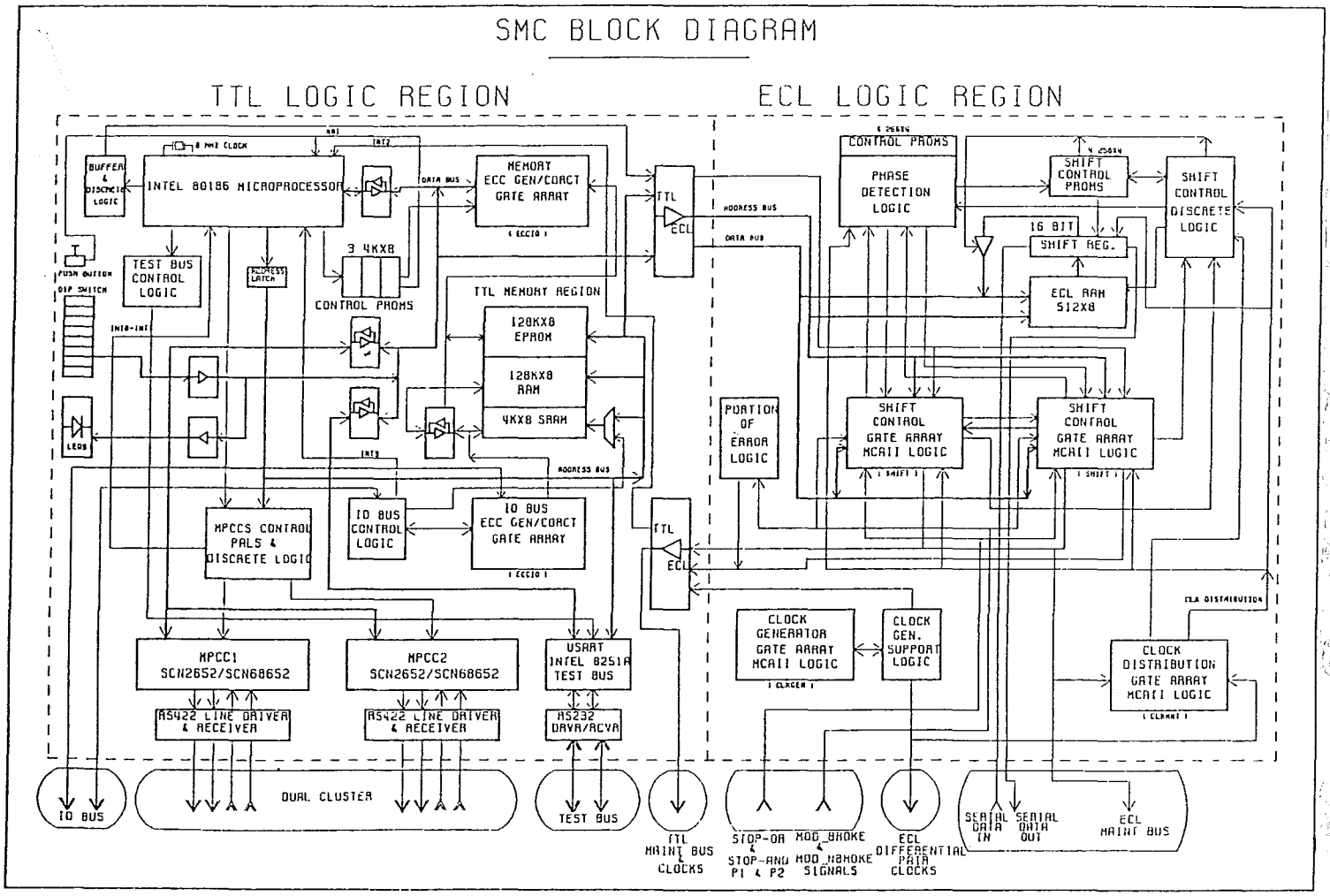


Figure 1 SMC Block Diagram

The analysis begins with the specific interfaces, then moves to the functional blocks of the board.

4 SMC BOARD INTERFACES

The MP to SMC interface path follows the RS-422 electrical specification. This interface consists of a differential pair wire for data, and a differential pair wire for clocks.

The MP is a B27 which has a RS-422 interface that transfers data at a rate of 1.8 MBit/sec. Two independant MP ports are provided per SMC board.

The test bus interface to the DLP subsystem is on the SMC. This is a 19.2K baud two-wire differential interface controlled by the 80186-8251 conbination.

The I/O bus interface (TTL) connects the SMC to the AIOBUS in each cabinet. The bus is 32 bits wide, plus 7 bits of ECC. The SMC card ID on this bus is 000000B7. There is a cabinet jumper that indicates to the SMC which cabinet it is in. The SMC I/O bus ID is 7; the ECC for the ID is B. The I/O ID is strapped on the backplane.

The TTL maintenance bus (Shift, Clear, Clock Enable, etc.) and clock interface have signals that are sourced on the board. TTL Mod Brokens are inputs to the board.

The SMC's ECL interfaces are ECL clocks, ECL maintenance signals (shift, clear, clock enable, etc.), error and stop condition signals, and force refresh signals. Refer to the V500 Clock Subsystem specification for details on the cabinet clock generation circuitry.

5 ORGANIZATION AND FUNCTIONAL BLOCKS

5.1 MICROPROCESSOR SECTION

This section contains 8Ø186 microprocessor, 128 KB of EPROM, 128 KB of RAM, 4 KB of SRAM, and 512 B of ECL RAM. All of the TTL memory region is accessable from the microprocessor through an ECCIO ALS gate array. The array generates ECC, and can detect single and multiple bit errors. The memory interface to the ECC chip is 39 bits wide, where 32 bits are data and 7 bits are ECC.

The ECL RAMs do not have ECC logic. ECL RAM is set up as a 16 bit wide data path that is directly accessed by the microprocessor data bus. Level conversion is required to read and write these RAMs. The conversion is done with discrete 1ØØF124 and 1ØØF125 family chips.

Figure 2 shows the microprocessor and its interface to memory.

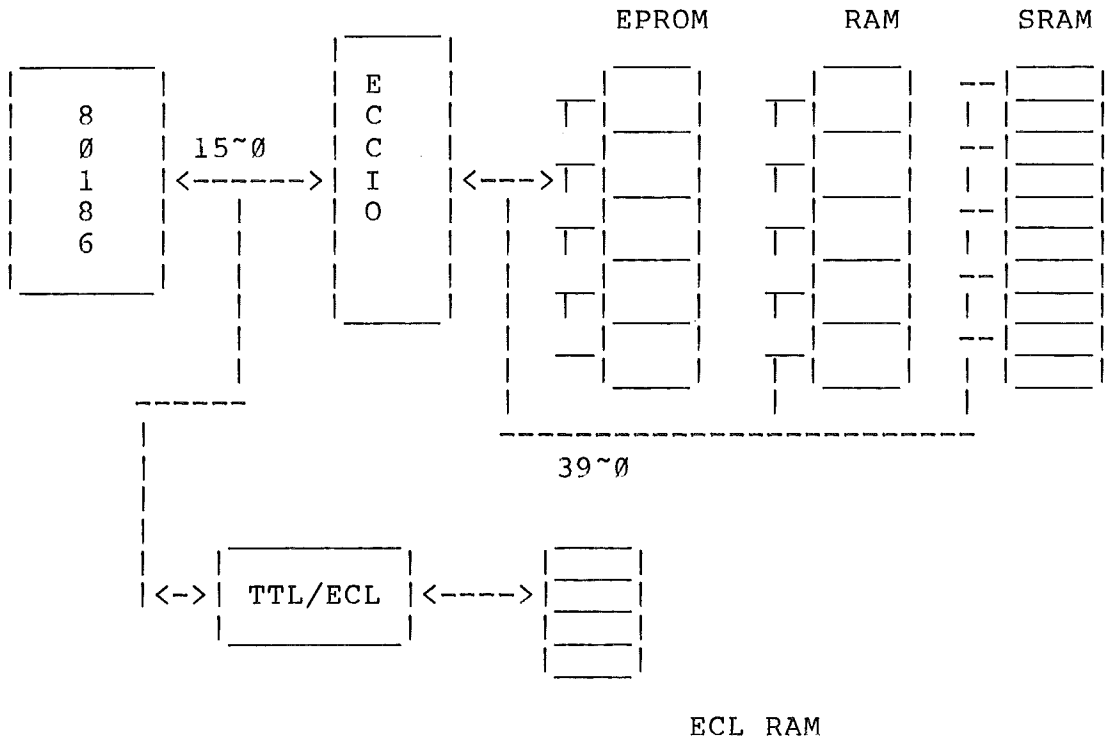


Figure 2 Microprocessor-Memory Interface

Following is a block diagram of the internal blocks of the 80186 microprocessor:

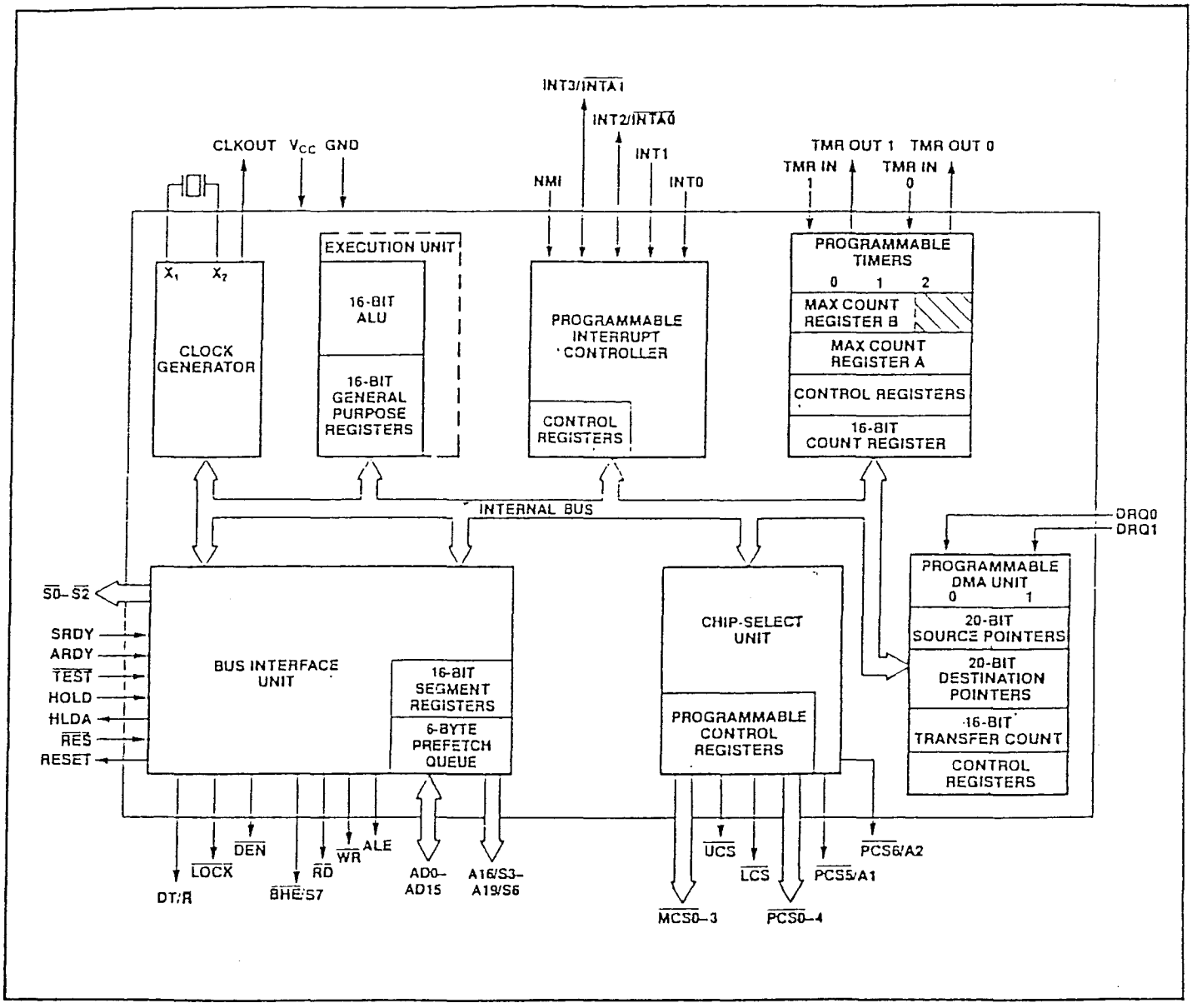


Figure 3 80186 Internal Block Diagram

5.1 MICROPROCESSOR SECTION (Continued)

The EPROMs contain the microcode for the SMC's initialization selftest and higher level software. The higher level software is segmented according to function. The VRTX real-time executive is the software manager for the board. It controls and manages software modules such as the Cluster software (SMC to MP communication), the State Access software (controller for the maintenance commands), and the I/O Bus software (server of the SMC's I/O bus).

The basic flow of microcode begins with a detailed initialization and self test module (NOT under the control of executive). When this diagnostic completes, control passes to the executive where the higher level software takes over. If an error is detected that prevents this, the diagnostic stops and displays the error indication on the LEDs.

The 80186 in the SMC uses a 14.7456MHz crystal and its own internal clock generator. This allows it to run synchronously with the USARTs, and minimizes the hardware required for the interface. The 8251 USART clocks are derived from a counter that is clocked by a 7.3728MHz 80186 clock generator. The counter divides the 80186 clock for a 19.2kbit/sec transfer on the test bus interface. The 2652 MPCC receives its clock from an external clock divider to transfer data on SMC to MP CLUSTER at 1.8432MBit/sec.

All data transfer operations on the MP interface are done using the on-chip 80186 DMA channels and internal timers. One DMA channel and one timer are allocated per MPCC chip. The MPCCs are wired so that they interrupt the processor when an end of message is detected while in receive mode.

The DMA interrupt is used to transmit End Of Message. The internal timer counts the number of data bytes transferred. The last count switches MPCC's address multiplexer so that it loads the last data byte in the control register. This is done to set the TEOM bit in the MPCC's control register (PCSARH) in order to meet the MPCC's timing constraints. TEOM is the Transfer End Of Message indicator bit to the MPCC chip.

5.1 MICROPROCESSOR SECTION (Continued)

An internal timer (Timer 2) keeps track of the VRTX timing.

The SMC microprocessor is primarily interrupt driven, using the edge-triggered interrupts in the 80186's fully-nested mode. The various interfaces interrupt it when required.

5.1.1 PERIPHERAL INTERFACES

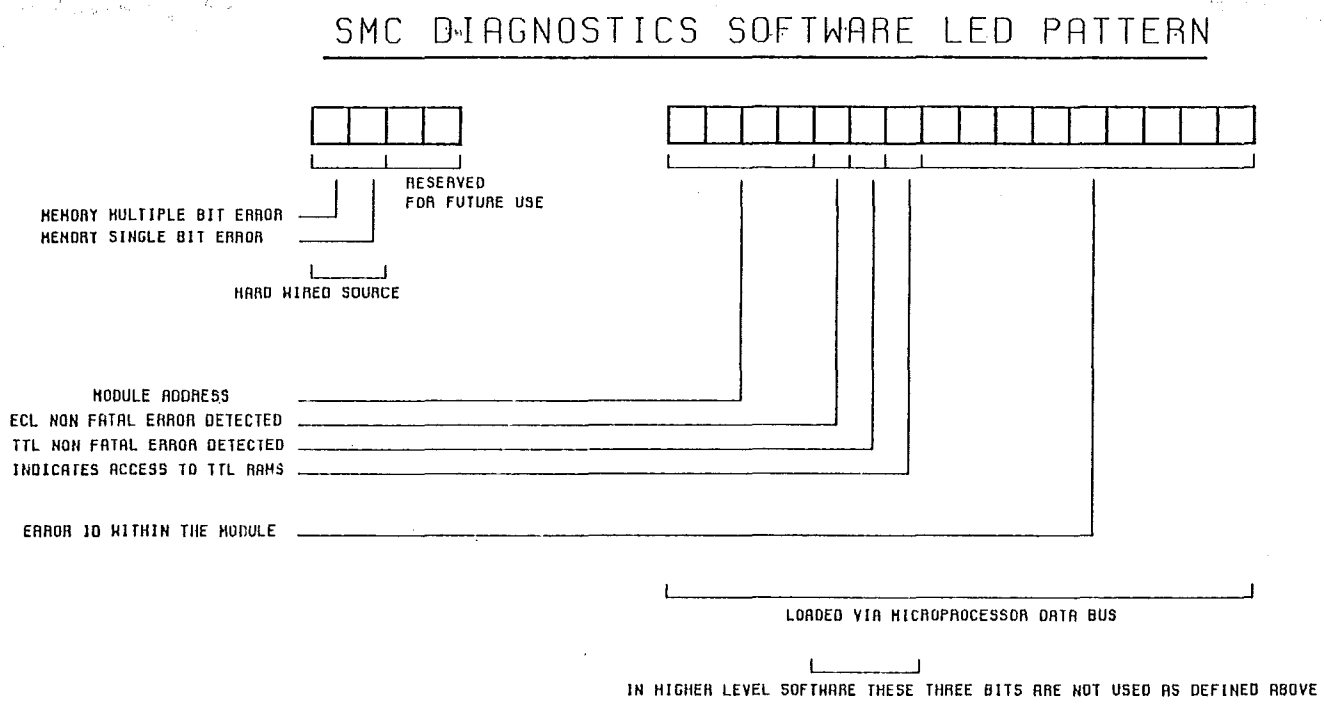
The following table indicates connection and control information for the microprocessor peripheral interfaces.

SOURCE/LOAD	PERIPHERAL CHIP SELECT	PCS ADDRESS RANGE	DMA CHANNEL	INTERRUPT LINE
MP 1	PCS-0	000-127	DMA-0	INT-0
MP 2	PCS-1	128-255	DMA-1	INT-1
Test Bus	PCS-2	256-383	-	INT-2
Shift/Control Array	PCS-3	384-511	-	INT-2
I/O Bus Control	PCS-4	512-639	-	INT-3
reserved	PCS-5	640-767	-	-
Maintenance Panel	PCS-6	768-895	-	NMI

5.1.2 MAINTENANCE PANEL USE

The maintenance panel is treated as a 80186 peripheral, and can be read or written under control of peripheral chip select 6. It contains 20 LEDs, an 8-bit DIP switch, and one pushbutton switch. The DIP switch is read as the LSB of a word.

The LEDs are defined as follows:



ERROR REPORTING

1) FATAL ERRORS
2) NON FATAL ERRORS

Appendix B contains the diagnostic error detection and reporting information.

Figure 4 SMC LED Definitions

5.1.2 MAINTENANCE PANEL USE (Continued)

The pushbutton switch causes a non-maskable interrupt (NMI) that causes the 80186 to read the DIP switches. The DIP switches determine the (software defined) operation to be performed.

If the most significant bit on the DIP switch is set when the pushbutton is pressed, a hard reset is sent to the microprocessor. Hard Reset resets the microprocessor and other logic not used in normal machine operation. It is used to reset the SMC while the machine is running. The remaining bits of the DIP switch define the offline or online function to be performed. This feature facilitates the debug of the board. For further information, refer to the SMC Diagnostic Test specification.

Figure 5 illustrates the SMC diagnostic test sequence.

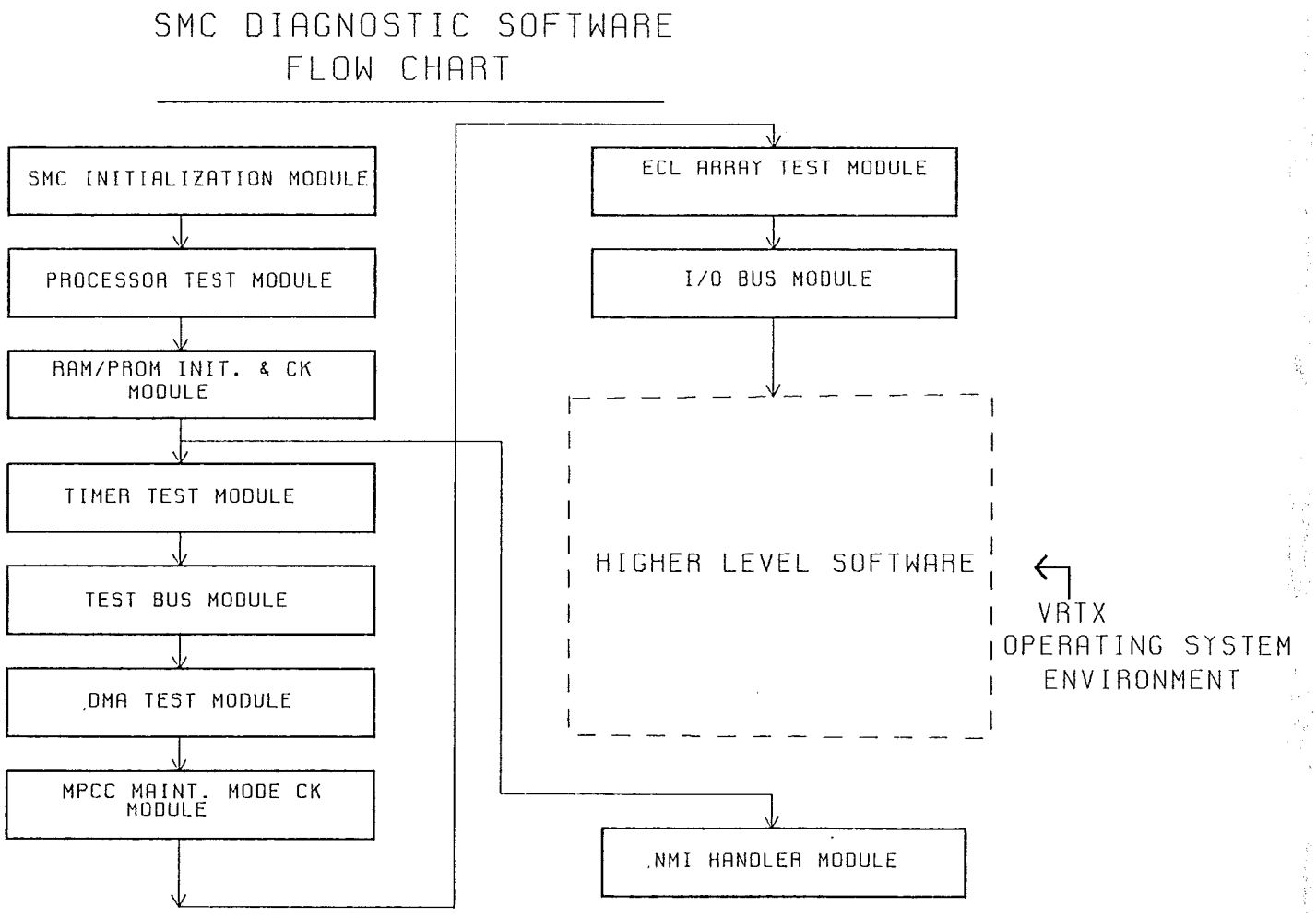


Figure 5 SMC Diagnostic Software Flowchart

5.2 I/O BUS SECTION

The I/O bus section is the fast interface to main memory. It is 32 data bits wide, plus 7 Hamming code bits. The bus uses a 2800ALS array, called ECCIO, common with the IOT, IOMC, and DTM. There are 4KB of SRAM buffer that can be accessed by either the I/O bus control or the uP.

The static RAM is used to transfer data into/out of main memory. During special operations (such as memory clear) the buffer does not have to be reloaded, and a higher average transfer rate is possible.

A set of PALS and PROMs provide the I/O bus control. The registers for this part are on the SMC TTL shift chain, and share a common design with the IOT, IOMC, and DTM.

Single bit bus errors are corrected on the I/O bus. They are reported to the MP for logging purposes via the SMC module broken signal.

5.2.1 BUS TRANSACTIONS

Bus transactions include:

Select an IOT
Select an IOMC to read or write memory
Be selected by an IOT to do an I/O

When a transaction is complete, the bus control will force the INT-3 line true.

Refer to the V500 DTM Specification for information on I/O requests to the SMC.

5.2.2 80186 INTERFACE WITH SMC I/O BUS

The SMC has only one I/O bus (ABUS). The registers used to interface with the microprocessor are:

NAME	ADDR	TYPE	USED BY	DESCRIPTION
BUFFADDR	614H	R/W	Buffer Control	Buffer start address
BUFFLEN	616H	R/W	Buffer Control	Message/reply length
ABUSSTAT	618H	R	IOBC A	Status register
BUFFSTAT	61AH	R	Buffer Control	Status register
ABUSCMND	61CH	W	IOBC A	Command register
BUFFCMND	61EH	W	Buffer Control	Command register

Note that the registers are treated as ports, and use an I/O-mapped addressing scheme (PBA=400h peripheral base address). The registers are defined in the following sections.

5.2.3 BUFFCMND REGISTER (61Eh)

Bit	Function
15	not used
14	Non-maskable interrupt enable
13	I/O interrupt enable
12	not used
11	Interrupt acknowledge to IOBC A
10	Resets single/multiple bit errors latches
9	Dead, causes module broken to get set
8	Busy set; sets module busy
7	Busy reset; resets module busy
6	Ignore error bit; used for fault detection testing
5	not used
4	Release buffer bit; used to release the buffer
3	CFORCEERROR1; used for fault detection testing
2	CFORCEERROR0; used for fault detection testing
1	Ownership hold bit
0	Request buffer ownership bit

5.2.4 BUFFSTAT REGISTER (61Ah)

Bit	Function
15	OFFLINEH (active high); module is off-line
14	not used
13	not used
12	NONMASKINTEN (active high); status of NMI enable
11	MBE (active high); memory multiple bit error status
10	SBE (active low); memory single bit error status
9	MBE (active high); Bus A multiple bit error status
8	SBE (active high); Bus A single bit error status
7	not used
6	SMC's I/O bus module broken status (active high)
5	SMC's I/O bus module not broken status (active high)
4	SMC's cabinet jumper status (active low)
3	not used
2	CACCGRANTL; indicates that the 80186 owns the I/O buffer (active low)
1	not used
0	AACCGRANTL indicates that Bus A owns the I/O buffer (active low)

5.2.5 BUFFLEN REGISTER (616h)

This register is used by the microprocessor to inform the buffer control of the length of a message or reply to be sent on an I/O bus. When the microprocessor is acting as an initiator, it loads BUFFLEN with the message length in hex (number of I/O bus words minus 1). For example, to send a message seventeen words long, BUFFLEN is loaded with hex 0010. Note that the length is the length of the message only, and does not include the Target module ID word.

When the module is a responder sending a reply, BUFFLEN is loaded with the length of the reply minus one before issuing the Continue command to IOBC.

For a one-word message or reply, BUFFLEN is loaded with 0000h.

BUFFLEN is not read or written while the I/O control (IOBC) is accessing the I/O buffer. The I/O buffer is the 1Kx32 MCS1-region fast RAM on the board.

5.2.6 BUFFADR REGISTER (614h)

BUFFADR is used by the I/O bus buffer control to address the buffer. The address contained in this register is a word address relative to the beginning of the I/O buffer.

BUFFADR is cleared automatically whenever buffer ownership changes. It is incremented by the I/O buffer control whenever a word is read or written in the I/O buffer. The microprocessor can read or write BUFFADR only when it is NOT being used by the I/O bus control.

5.2.7 ABUSSTAT REGISTER (618h)

Bit	Function
15	AIIOINTH (active high); source of I/O bus interrupt
14	not used
13	ABIDERR(2) (active low); bus arbitration error
12	ABIDERR(1) (active low); bus arbitration error
11	ABIDERR(0) (active low); bus arbitration error
10	AERRLINE (active high); error report from the responder
9	AMBE (active high); multiple bit error detected by I/O Bus ECCIO gate array
8	ASBE (active low); single bit error detected by I/O Bus ECCIO gate array.
7	ASELRES(2) (active high); responder selection result.
6	ASELRES(1)
5	ASLERES(0)
4	ABIDRES(2) (active high); bus arbitration result.
3	ABIDRES(1)
2	ABIDRES(0)
1	APREEMPTL (active low); indicates that another requestor wants to preempt the buffer.
0	AOPCOMP (active low); indicates operation completion.

5.2.8 ABUSCMND REGISTER (61Ch)

bit	function
15	AIGNOREERR (active high); used for fault detection testing
14	AFORCEERR(6)
13	AFORCEERR(5)
12	AFORCEERR(4)
11	AFORCEERR(3)
10	AFORCEERR(2)
9	AFORCEERR(1)
8	AFORCEERR(0)
7	AIEMERG (active high); used with the ASTART bit to specify emergency priority during bus arbitration.
6	AIIOGOT (active high); causes IIOACKL to IOMC.
5	not used, must be zero.
4	not used, must be zero.
3	ARESET (active high); resets I/O bus control logic.
2	ADISC (active high); disconnect - no reply required.
1	ACONTINUE (active high); continue - send reply in the I/O buffer.
0	ASTART (active high); start I/O bus command.

5.3 TEST BUS SECTION

The Test Bus section consists of a 8251A USART (see Figure 6), a driver/receiver pair, and an interface to the 80186 bus. Protocol is provided by the 80186. Messages are received from the MP or test bus, and relayed to the test bus or MP. The MP contains the test data base. The 80186 addresses the 8251 as follows:

PBA + 256: READ/WRITE DATA
PBA + 264: READ/WRITE STATUS

This is PCS-2 with address bit 3 selecting STATUS or DATA.

The 8251 is connected to the LSB of the 80186 data bus, and uses a clock derived from the 1.8432MHz MPCC clock for internal use. The 8251 is programmed as follows to produce 19.2kbit/second transfer rate:

MODE INSTRUCTION = 5Eh

This means 16X mode, 8 bit character length, odd parity, and one stop bit.

The TXRDY (Transmitter Ready) and RXRDY (Receiver Ready) signals are OR'ed and sent to the INT2 interrupt. When the microprocessor receives TXRDY, it loads the 8251 transmitter buffer with 8 bits of data.

When the microprocessor receives RXRDY, it reads the data from the receive buffer. This read is done as soon as possible so as to avoid the possibility of a data overrun. An overrun error occurs if the microprocessor does not read the buffer before it is overwritten by the next character.

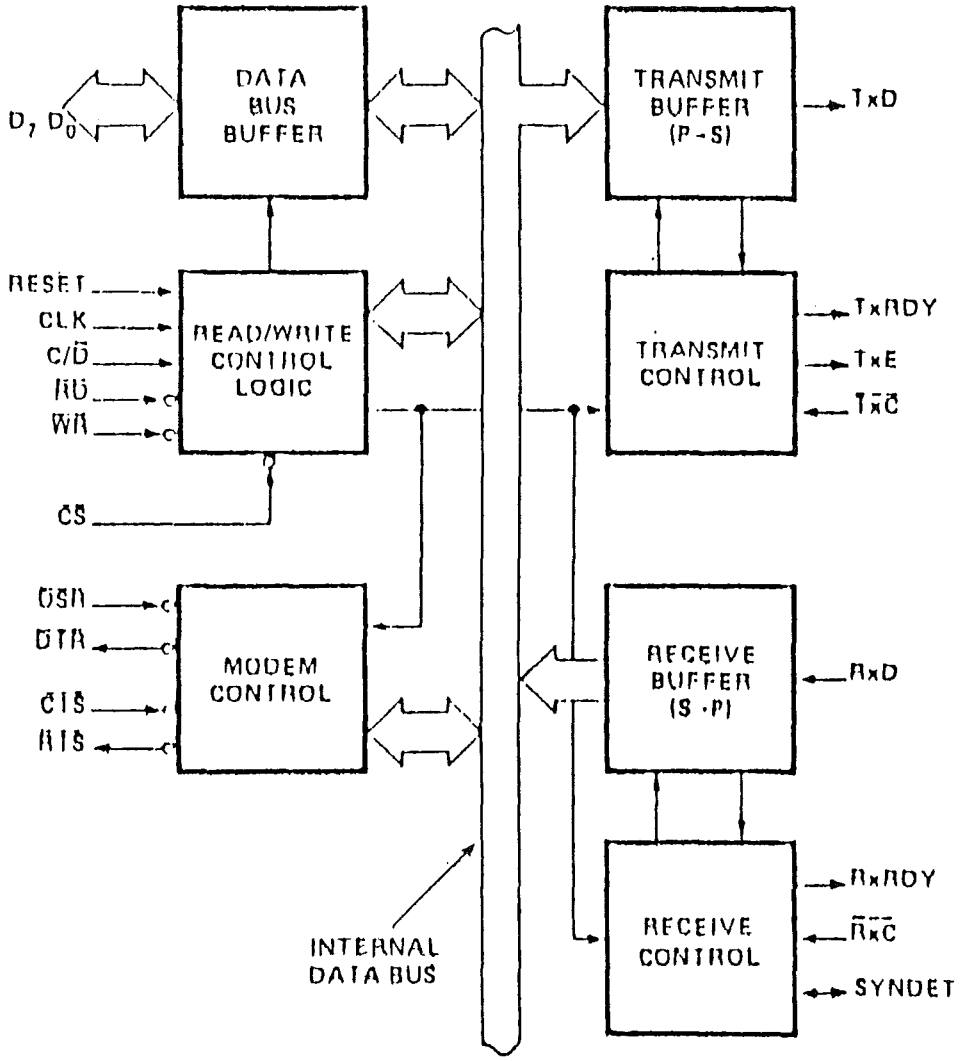


Figure 6 8251 USART Block Diagram

5.4 PROGRAM MEMORY ERROR INTERFACE

Program memory errors are flagged as single or multiple bit errors. These error conditions are monitored during the correction window in the microprocessor's memory read cycle. (see Sections 9.3 and 9.4 for the encode and decode tables). The error is registered in a PAL.

To reset the error, bit 10 in the I/O Bus command register (BUFFCMND) is set by a parallel write from the 80186. A single or multiple bit error generates a non-maskable interrupt (NMI).

During system initialization, the RAM ECC can be bad. Therefore, the ECC error reporting of NMIs is disabled by a hardware qualifier bit (bit 15 in the I/O Command register port BUFFCMND; 61Eh). In order to load this register, the data clock of the SMC TTL chain must be enabled. This is necessary because the I/O Command register uses the data clock.

5.5 CLOCK AND SHIFT CONTROL

Two MCA 2500 ECL arrays (SHIFT arrays) are used. They contain the following interfaces:

- 37 clock enable registers
- 37 clock stop select registers (MASK/TOGGLE registers)
- Error signals from the V500 system.
- Stop signals from the V500 system.
- Control lines to the V500 system (SHIFT,CLEAR,DANGER)
- Select lines to the V500 system (MODULE_EN,CHAIN_EN)
- Data lines to the V500 system (SHIFT_IN,SHIFT_OUT)

1. Cards have one clock enable for each clock. Therefore, cards with only ECL or TTL logic get one clock enable. Cards with both ECL and TTL logic receive two clock enables.

2. Each clock enable bit has a corresponding clock start/stop select bit. If this bit is set, it enables the corresponding clock enable bit to toggle from its previous state if a Toggle, Counterstop, or Stop event occurred (see Sections 5.5.1). Note that unselected clocks are not affected. Various stop modes are under software control.

   Figure 7 shows the logical block diagram of a shift array and the timing relationships for shifting a chain.

5.5 CLOCK AND SHIFT CONTROL (Continued)

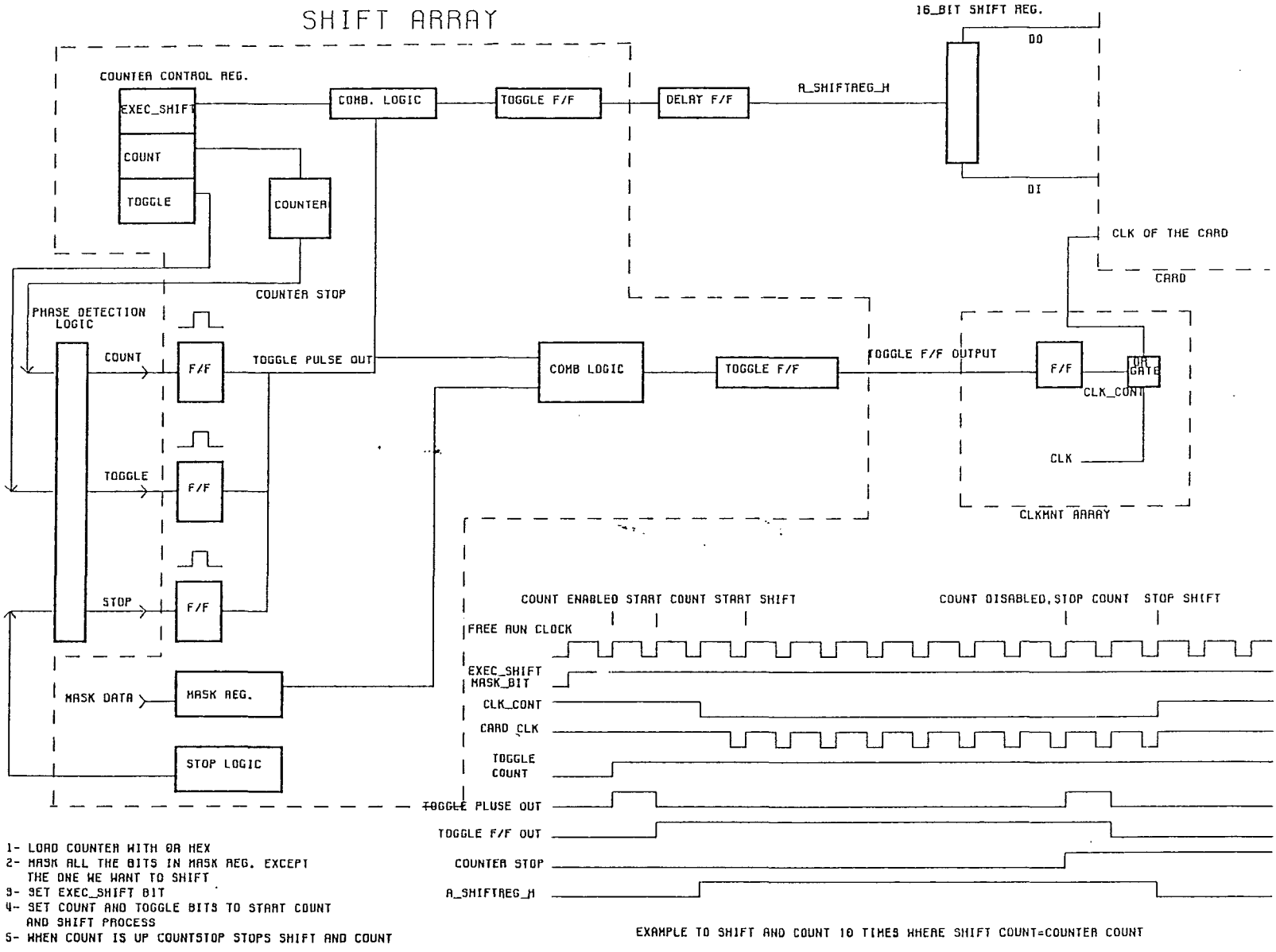


Figure 7 Shift Array Timing

5.5 CLOCK AND SHIFT CONTROL (Continued)

3. There are two groups of system errors. The first group consists of the XM and MCACM errors. These error conditions result in immediate clock freezes of some or all clocks in the system. For processor hardware error recovery, XM errors result in immediate clock freezes of the XM and Fetch modules in the processor with the faulty XM. For MCACM errors, the clocks to both processors are shut off immediately.

   The second group consists of all other modules, which report errors to the SMC. These are handled at microprocessor speed. The errors are registered in the shift arrays and in an external register at port address 582h. The occurrence of the error is reported to the microprocessor via the INT2 interrupt line.

5.5 CLOCK AND SHIFT CONTROL (Continued)

Errors are gated into the shift array as shown in Figure 8; the connection to the external registers is similar.

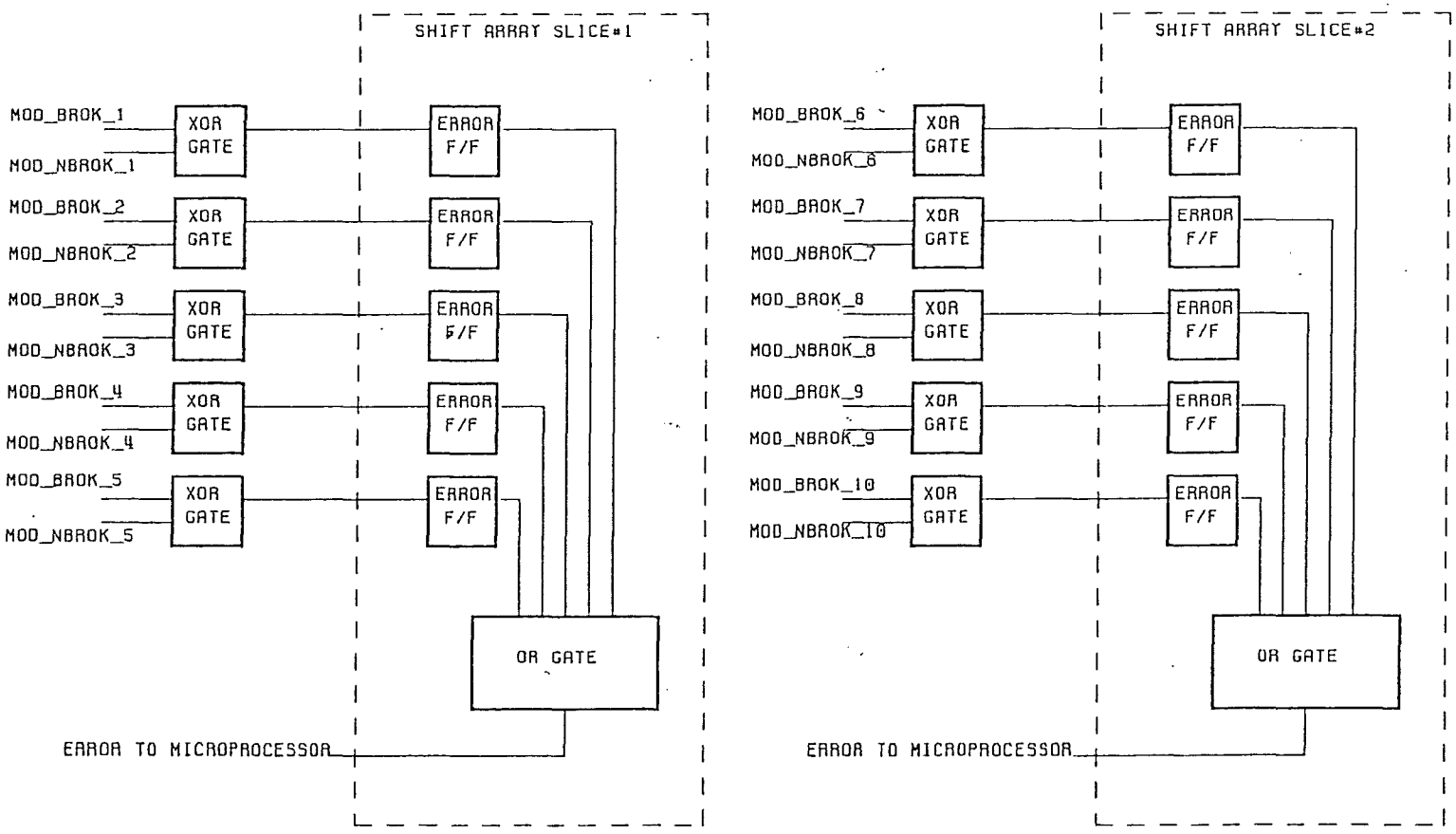


Figure 8 Shift Array Error Gates

5.5 CLOCK AND SHIFT CONTROL (Continued)

4. The Stop logic monitors the Stop And/Or wires from both processors, and the I/O subsystem.

The shift array stop logic is shown in Figure 9.

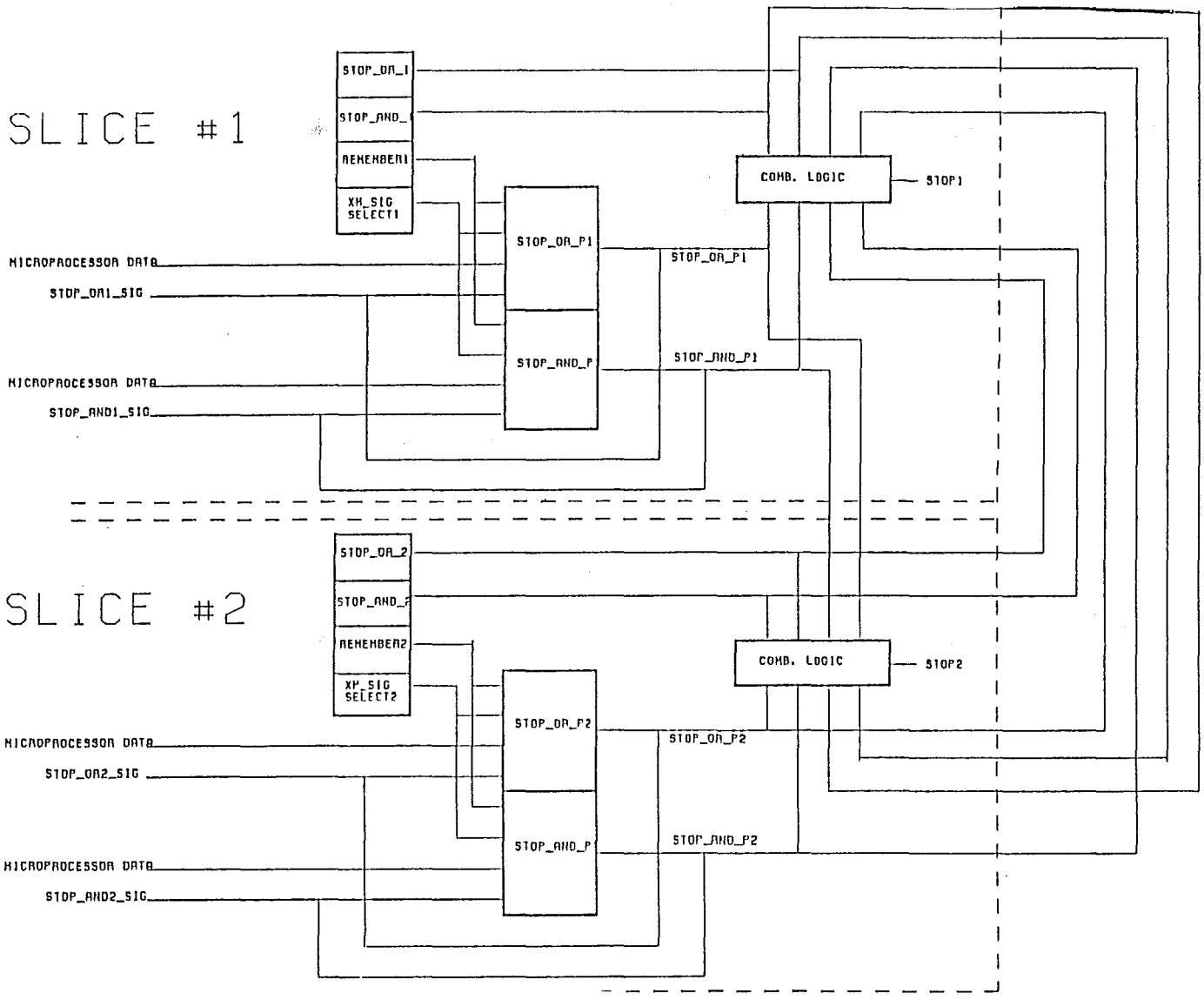


Figure 9 Shift Array Stop Logic

5.5 CLOCK AND SHIFT CONTROL (Continued)

The I/O subsystem stop AND/OR lines and second processor's stop AND/OR lines are wired ORed. The setup involves the selection of the modules to be stopped, selection of the type of stop event (for example, stop OR on the first processor), and setting up of the stop event in the maintenance chains of the desired modules.

When the stop event occurs, the selected module's clock enables are stopped as soon as possible with the proper phase relationship preserved between the ECL and TTL clocks. Refer to the Shift Array specification for possible combinations for stopping events.

Event stops at a SI point can be done by hardware in SMC shift arrays. This logic looks at the And of MCACM_IDLE with XMSTOPTIME to determine if a SI condition has occurred. The result is registered in the PROHALT1 and PROHALT2 flip/flops in external register port (582h), bits 2 and 3, respectively. Note that the occurrence of stop events and SI events are reported to the microprocessor via the INT2 interrupt.

5. The control signals for the processor are a part of the maintenance bus. The SHIFT signal is the control line that places the modules in shift mode. The CLEAR places the modules in clear mode. The DANGER signal qualifies the RAM write enable and de-skew logic in the CLKMNT3 gate array.

6. The select signals determine which module is selected for shifting or clearing chains. Module Enable selects the desired module; the Chain Enables (also called card enables) select the particular chain within the module.

7. The data paths to the processor are Serial Data In and Serial Data Out. These paths create a common serial bus between the SMC and the rest of the cabinet's cards.

5.5.1 SHIFT ARRAY AND THE ECL DISCRETE CONTROL LOGIC

The function of this hardware is to read, write, rotate or clear chains. This logic is also used to read or write any RAM, and to provide clock enable signals to all cards.

Figure 10 shows a logic diagram of the shift arrays; Figure 11 shows the bit assignments of the two slices.

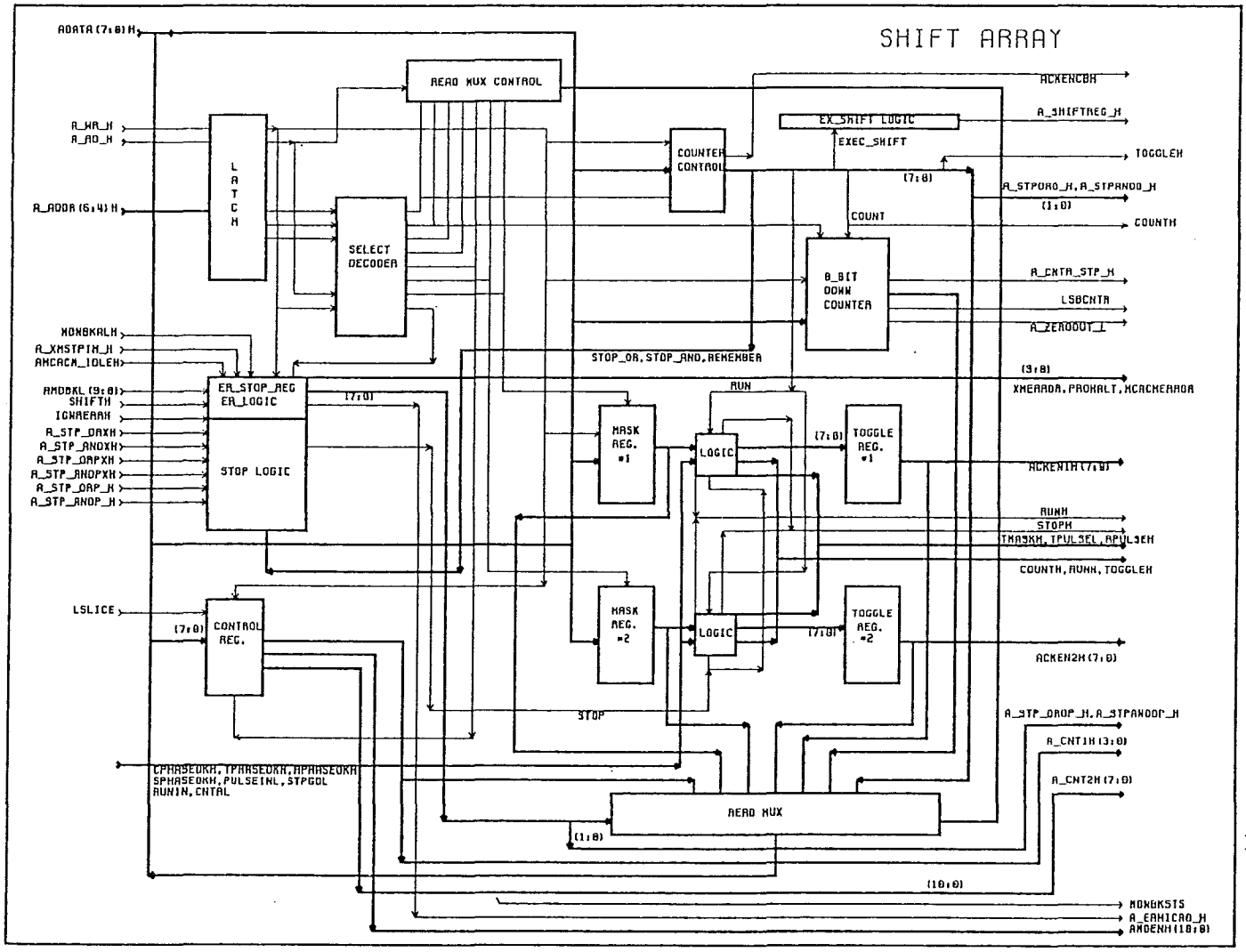


Figure 10 Shift Array Logic Diagram

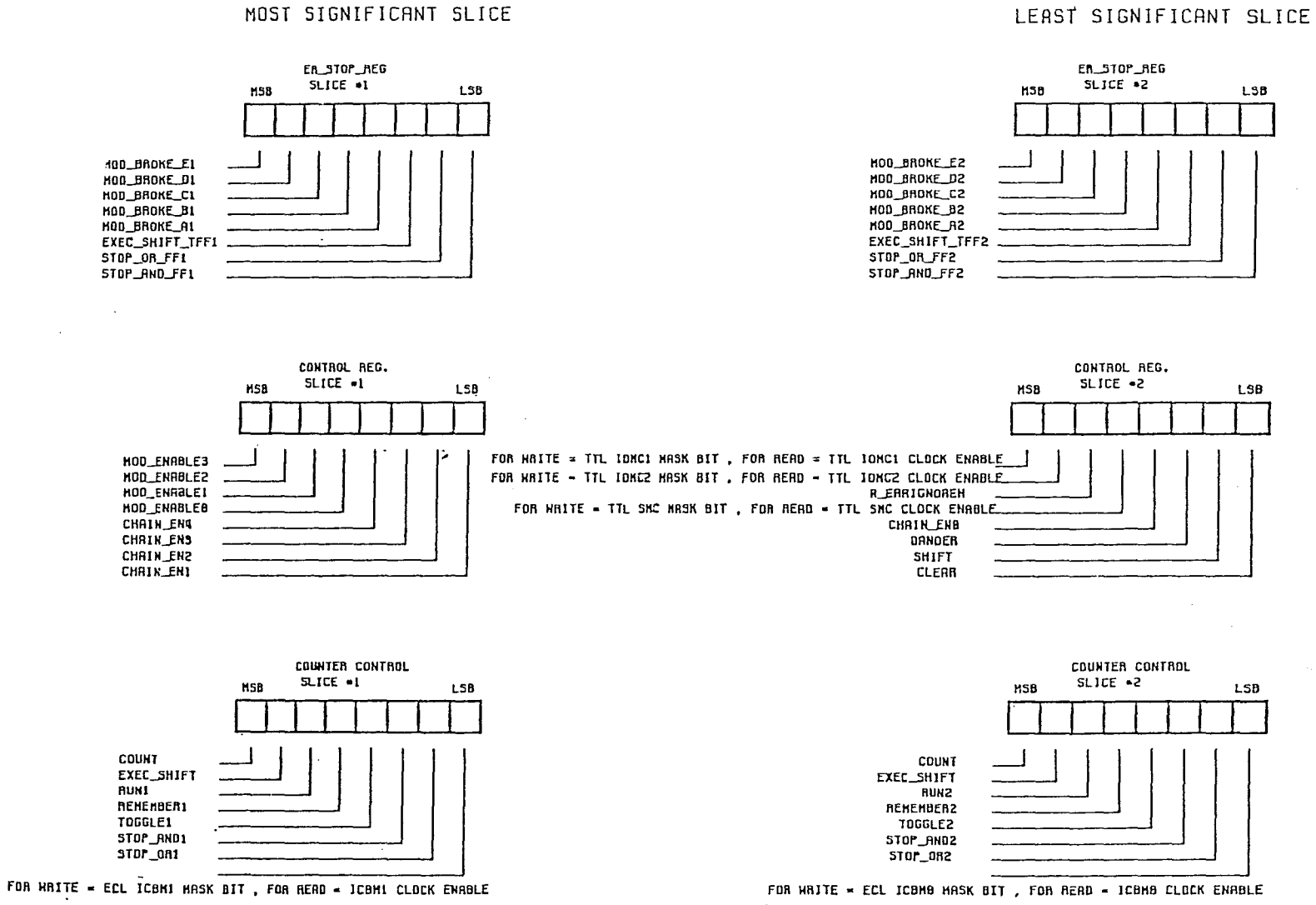


Figure 11 Shift Array Bit Assignments

5.5.1 SHIFT ARRAY AND ECL DISCRETE CONTROL LOGIC (Continued)

The Shift array consists of two identical arrays, each containing half of the logic. Each half controls one processor, four memory cards, and half of the potential I/O.

A port address corresponds to 16 bits of data, half of which is from one slice, and the other half from the other slice.

There are six logical blocks in the array. The blocks are addressed by the 80186 (with PCS-3), in conjunction with address bits 6, 5, and 4. This places these blocks at the location specified by Peripheral Base Address plus 384 through 496 (180h - 1F0h).

The 80185 address bits select functions as follows:

80186 address bits 6	5	4	port	select
0	0	0	580	Counter
0	0	1	590	Counter Control
0	1	0	5A0	Control register
0	1	1	5B0	Mask/Toggle register 1
1	0	0	5C0	Mask/Toggle register 2
1	0	1	5D0	Error Stop register
1	1	0	5E0	Clock Enable register Group 1
1	1	1	5F0	Clock Enable register Group 2

The Counter is a 16-bit down counter that counts clocks. If shift mode is selected, the Counter counts the number of bits a chain is shifted. If clear mode is selected, the selected chain(s) clear for <count> clocks. The Counter can be disabled to rotate a chain continuously, or to run clocks. Counter Control sets the mode for the Counter, and is duplicated in both arrays.

5.5.1 SHIFT ARRAY AND ECL DISCRETE CONTROL LOGIC (Continued)

The Control register contains data sent to the clock distribution arrays on the various cards, and extends over both arrays.

A Mask/Toggle register is used for clock control. It is set by the 80186 with a mask bit for each chain. When a selected event is found, (such as a stop event or Counter = 0) each bit in the Clock Enable register corresponding to a 1 bit in the Mask/Toggle register is complemented. A bit can be set in the Counter Control register by the 80186 to allow an immediate complement. This bit complements itself (if set).

A method of stopping all clocks would be to read the Clock Enable register, write it into the Mask/Toggle register, and write the Toggle Immediate bit to the Counter Control register.

The SHIFT2 array Clock Enable register definitions are as follows:

CLOCK ENABLE REGISTER BIT DEFINITION

	SLICE 1		MOD_SEL	CHAIN		SLICE 2		MOD_SEL	CHAIN
	CLK_EN2 REG:					CLK_EN2 REG:			
	BIT 31	XMX-1	7	6		BIT 23	XMX-2	2	6
	BIT 30	XMD-1	7	4		BIT 22	XMD-2	2	4
	BIT 29	XMC-1	7	2		BIT 21	XMC-2	2	2
	BIT 28	FETCHD-1	8	6		BIT 20	FETCHD-2	3	6
	BIT 27	FETCHI-1	8	4		BIT 11	FETCHI-2	3	4
	BIT 26	FETCHC-1	8	2		BIT 18	FETCHC-2	3	2
	BIT 25	MCACMD-1	9	4		BIT 17	MCACMD-2	4	4
1	BIT 24	MCACMA-1	9	2		BIT 16	MCACMA-2	4	2
	CLK_EN1 REG:					CLK_EN1 REG:			
	BIT 15	IOMC-1	1	4		BIT 07	IOMC-2	5	2
	BIT 14	DTM5	6	C		BIT 06	DTM2	6	6
	BIT 13	DTM4	6	A		BIT 05	DTM1	6	4
	BIT 12	DTM3	6	8		BIT 04	DTM0	6	2
	BIT 11	MDC7	A	10		BIT 03	MDC3	A	8
	BIT 10	MDC6	A	E		BIT 02	MDC2	A	6
	BIT 09	MDC5	A	C		BIT 01	MDC1	A	4
	BIT 08	MDC4	A	A		BIT 00	MDC0	A	2

5.5.1 SHIFT ARRAY AND ECL DISCRETE CONTROL LOGIC (Continued)

The maintenance chain on each card is addressed by the card chain number, plus one (i.e. the maintenance chain on DTM1 is 5).

Module enable lines are sourced from the most significant slice of the shift array.

Refer to the Shift Array specification for additional information regarding this array.

5.6 CLOCK CONTROL & DISTRIBUTION ARRAY

5.6.1 OPERATIONAL PRAGMATICS

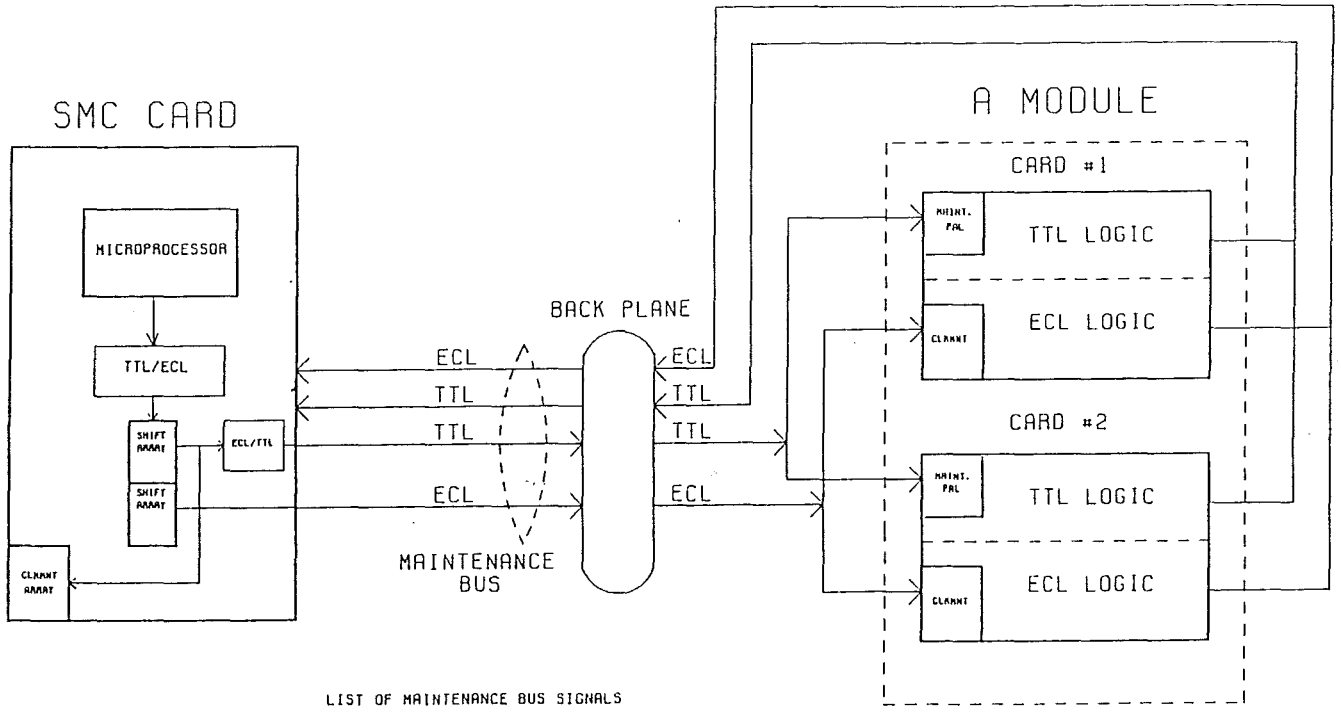
Some brief definitions:

Signal	I/O	Description
CLEAR_M	O	Clear the maintenance chain
CLEAR_D	O	Clear the normal chain
SHIFT_M	O	Put the maintenance chain into shift mode
SHIFT_D	O	Put the normal chain into shift mode
RAM_WE	O	A one clock signal to load new RAM data
MOD_BROKEN	I	Turn off clocks right now.
MOD_NOT_BROK	I	Turn off clocks right now.
STOP_ME	Int	The AND of MOD_BROKEN with /MOD_NOT_BROK
SCAN	x	D-Drive signal for testability of this array
MARYSEL(15:0)	I	Maintenance array clock select (16 jumpers)
ITS_ME	Int	B.P. select = B.P. ID
CHAIN_EN0	I	LSB of the 5 chain enable signals
CLOCK(52:0)	O	53 clock lines, one to each possible array
DANGER	I	Signal from SMC for RAM_WE, or clock skew

Timings are given in Section 9.5, except POWER_ON_CLEAR which comes from the ECM when power is turned on or off. The SMC distributes a buffered clear to the clock distribution array of each card. The clock distribution array initializes its clock circuit and produces CLEAR and CLOCKs to its card. This CLEAR is of uncertain duration.

Refer to the CLKMNT3 Specification for details on functionality of this chip.

Figure 12 shows the interface of SHIFT and CLKMNT arrays, and their roles in providing the system with maintenance bus interface.



LIST OF MAINTENANCE BUS SIGNALS

1) MODULE ENABLES
2) CLOCK ENABLES
3) CARD ENABLES (CHAIN ENABLES)
4) SHIFT ENABLE
5) CLEAR ENABLE
6) DANGER
7) SERIAL DATA IN
8) SERIAL DATA OUT
9) MODULE BROKEN SIGNALS
10) MODULE NOT BROKEN SIGNALS
11) STOP OR SIGNALS
12) STOP AND SIGNALS

Figure 12 SMC - Maintenance Bus Interface

5.7 MAINTENANCE PROCESSOR COMMUNICATION

The interface to the MP uses either of two on-board 2652 MPCC (Multi-protocol Communication Controller) chips. They are selected by PCS-0 and PCS-1, in conjunction with the three least significant 80186 address bits. PCS-0 addresses a four bit control register on an I/O write if ADRESS LINE 5=1. If address line 5=0, then the MPCC0 registers are accessed. PCS-1 does the same for MPCC1. The two MPCCs are wired in an 8-bit configuration; their internal registers are addressed as follows:

A5	A4	A3	A2	A1	A0	register
0	X	X	0	0	0	RDSR lower
0	X	X	0	1	0	RDSR upper
0	X	X	1	0	0	TDSR lower
0	X	X	1	1	0	TDSR upper
0	X	1	0	0	0	PCSAR lower
0	X	1	0	1	0	PCSAR upper
0	X	1	1	1	0	PCR upper

EXTERNAL STATUS REGISTER

80186 address bus bit	name	function
4	INTERNAL	Selects the external status register
3	MM+	Maintenance mode
2	RxE+	Receive enable
1	TxE+	Transmit enable
0	TxCLEN+	Transmit clock enable

Figure 13 is a block diagram of the MPCC chip.

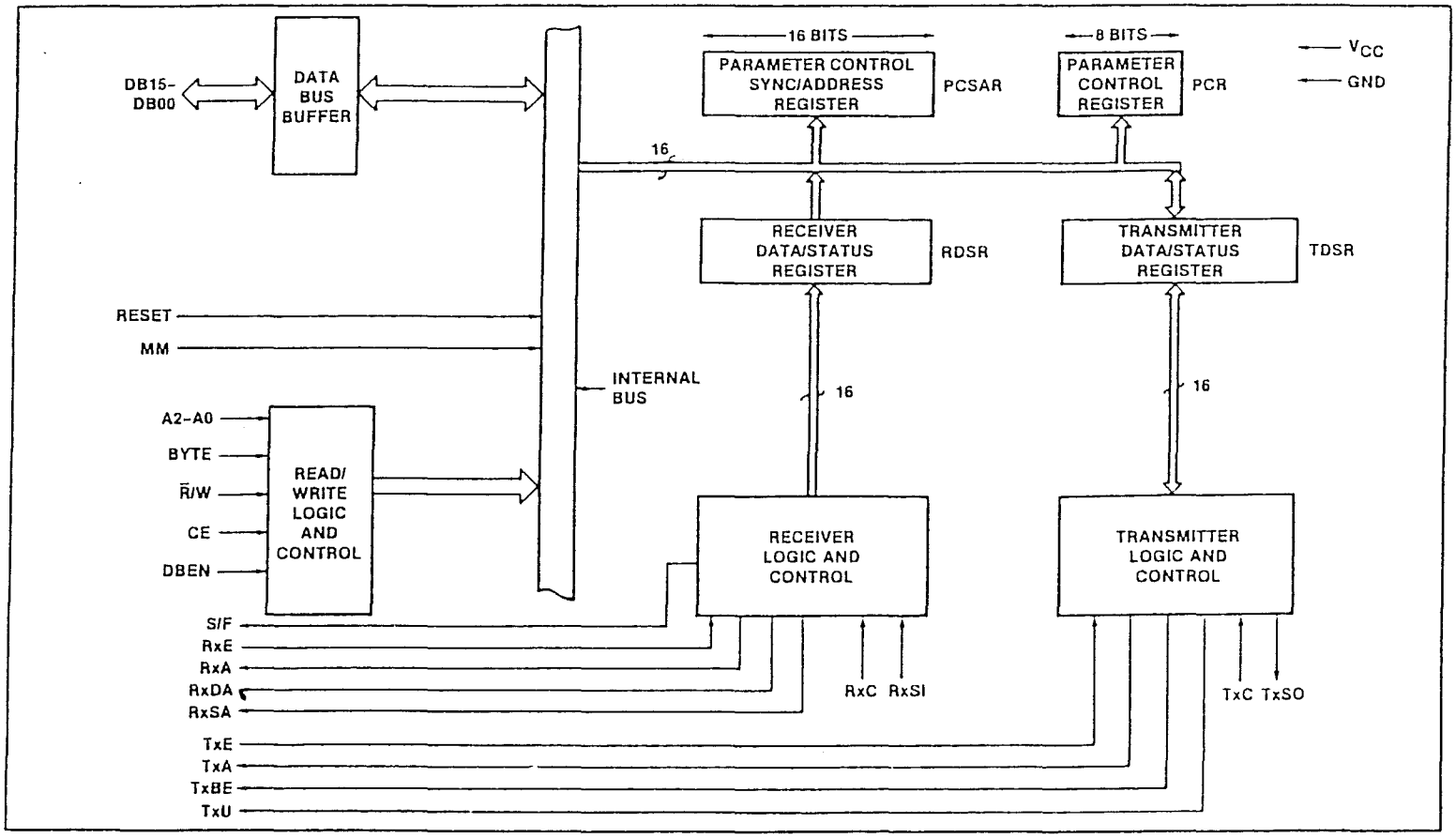


Figure 13 MPCC Block Diagram

5.7 MAINTENANCE PROCESSOR COMMUNICATION (Continued)

The MPCCs are wired into the 80186 DMA channels in destination synchronized mode when the SMC transmits information. They are in source synchronized mode when the SMC receives information from MP. This means that the MPCCs generate DMA request signals. These signals tell the DMA channels when to send the message from memory to the MPCC's transmitter buffer in transmit time (destination sync mode).

The DMA request signals also tell the DMA channels when to read the MPCC's receive buffer and when to dump it in receive time (source sync mode). Data transfer is automatic, once the initialization procedures are finished. INT-0 or INT-1 signal the end of a receive cycle. The DMA interrupt signals the end of a transmit. The 80186 TIMER-2, in conjunction with VRTX operating system, keeps track of timeouts on the cluster.

The datacom clock is generated by a divide-by-four circuit. It transfers data at 1.8432MBit/second.

5.8 SMC ON-BOARD PROGRAM MEMORY

The SMC memory chip selects access to all the memory mapped I/O and memory chips on the board. They are defined as follows:

LCS	128K bytes of slow RAM (TTL)
MCS0	not used
MCS1	4K bytes of fast RAM (TTL)
MCS2	512 bytes of fast RAM (ECL)
MCS3	not used
UCS	128K bytes of EPROM (TTL)

MCS2 address 30200h is used to access a memory mapped register; see the Phase Detection portion of this specification.

The I/O bus logic is restricted to using the fast TTL RAM. The shift control logic is restricted to the ECL RAM. The 80186 can use any of it, but is locked out of the fast TTL RAM when the I/O bus is running, and must be programmatically kept out of the ECL RAM when any chains are shifting.

6 SMC SELF-MAINTENANCE

The SMC has provisions for stand-alone maintenance. SMC maintenance is monitored by LEDs, and controlled by eight DIP switches and one pushbutton switch.

6.1 SWITCHES

The switches allow debug in offline or online mode. They control looping on any test module. The switches are read-only (via PCS-6) on the LSB of the data lines.

6.2 THE MONITOR REGISTER

The SMC displays its failures on 20 LEDs. The LEDs report errors as follows:

LED	error
20	Memory ECCIO detected multiple bit error
19	Memory ECCIO detected single bit error
18 - 17	not used
16 - 13	Module address of software being executed
12	ECL logic non-fatal error
11	TTL logic non-fatal error
10	TTL RAM access in progress; not an error
9 - 0	See Appendix B for descriptions of these LEDs

6.3 SMC FAULT DETECTION

The SMC uses two types of fault detection; one for internal faults and another for interface faults.

6.3.1 INTERFACE FAULTS

The serial RS-422 interface with the B27 uses CRC-CCITT (X**16+X**12+X**5+1) code, as recommended for the ANSI Advanced Data Communication Control Procedure (ADCCP) of which this interface is a subset (SDLC).

The SMC has one parallel interface, the I/O bus. That bus has Hamming code error detection.

6.3.2 INTERNAL FAULTS

Internal faults are detected by the microprocessor selftest. This test includes a check of all RAM, a loop-back test all UARTs, an echo test to the shift control array, and a check-sum test of the EPROM. Also, when a value is sent to the shift control array, it is read back and compared (optionally during C.S. load).

Program RAM and EPROM uses the same Hamming code used by the IOT and I/O bus. The I/O bus logic is tested using path tests. The I/O bus control detects faults while the machine is running.

6.3.3 SMC ERROR REPORTING

Depending on the error type, the SMC either reports it to the MP, or freezes and lets the MP detect an interface timeout. RAM or EPROM single bit errors have the error vector reported to the MP. Refer to Appendix B for details.

7 CLOCK RECEIVER AND MAINTENANCE ARRAY

The clock receiver and maintenance array is a common array for every V500 ECL logic card. It accepts inputs from the SMC, and produces CLOCK, SHIFT, CLEAR, SDO, and RAM_WE outputs.

7.1 MAINTENANCE SIGNALS

The SMC supplies the RAM_WE signal to prevent RAM data corruption during other maintenance operations. This signal is one clock wide, and in conjunction with a user-provided RAM select signal, can alter control store contents. This signal allows the shift chains to be altered or shifted without modifying the RAMs.

The CLOCK provided to each ECL module is normally de-skewed. As an option, the clock skew can be changed for margin testing. This is described in the clock subsystem specification.

The CLEAR signal clears all visible state when asserted and clocks are present. It, with clocks, is asserted for a minimum of 1 microsecond. This is required for I/O, and can also allow some registers or counters to clear by shifting zeroes.

SHIFT works the same way as in previous machines. It is asserted while clocks are turned off. Clocks are turned on and off intermittently to shift the chain. When the chain has been restored to its original position, possibly containing modified data, and clocks are off, then SHIFT is negated. This is intended to be the normal way to read or write a register.

SDI (Serial Data In) and SDO (Serial Data Out) are provided by each individual module/card. SDO must be disabled whenever a shift chain is not shifting (SHIFT is false). This is because several modules/cards can have SDO tied together. Also, due to bus timing and loading, SDO must be driven by a flip/flop-to-driver combination with no other logic inserted.

The SMC sources SDI to each group of cards. If the module or card is not shifting, then SDI has no significance to that module or card.

7.2 UNINITIALIZED HIDDEN STATE ERRORS

Hidden state, such as scratchpad RAMs, control stores, etc, are likely to cause detected errors when uninitialized. The error ignore bit is used to mask these until proper initialization is complete. This is bit 5 of the control register of SHIFT slice number two.

8 ECCIO GATE ARRAY

The primary functions of the ECCIO array are to correct single bit errors and detect double bit errors. The array generates 7 bits of ECC for 32 bits of data. To generate new ECC, the GENECC and the HOLDECC control signals must be held at 1; all other control signals are held at Ø (excluding RAMSEL and the tristate control signals). The ECCIO array can also assemble an entire word, byte by byte. The HOLD, HOLD1, HOLD2, HOLD3, HOLD4, and HOLDECC control signals are used to hold 8 bits of data in the flip/flops.

8.1 ECCIO ARRAY SIGNAL DEFINITIONS

Signal	Function
CORRECT	1 = Correct single bit errors
RAMSEL	1 = Select data from RAM Ø = Select data from microprocessor (memory ECCIO) and backplane (IOBUS ECCIO)
SHIFT	1 = Allow data to be shifted into and out of array flip/flops
GENECC	1 = Generate 7 bits of ECC for 32 bits of data
CLEAR	1 = Reset flip/flops
CLOCK	Receive microprocessor clock (memory ECCIO) and system TTL clock (IOBUS ECCIO)
SHIFT_IN	Serial data to the chip
HOLD	1 = Hold all 39 bits in shift chain
HOLDØ1	1 = Holds bits Ø through 7 only; HOLD must be Ø
HOLDØ2	1 = Holds bits 8 through 15 only; HOLD must be Ø
HOLDØ3	1 = Holds bits 16 through 23 only; HOLD must be Ø
HOLDØ4	1 = Holds bits 24 through 31 only; HOLD must be Ø
HOLDECC	1 = Holds bits 32 through 38 only; HOLD must be Ø

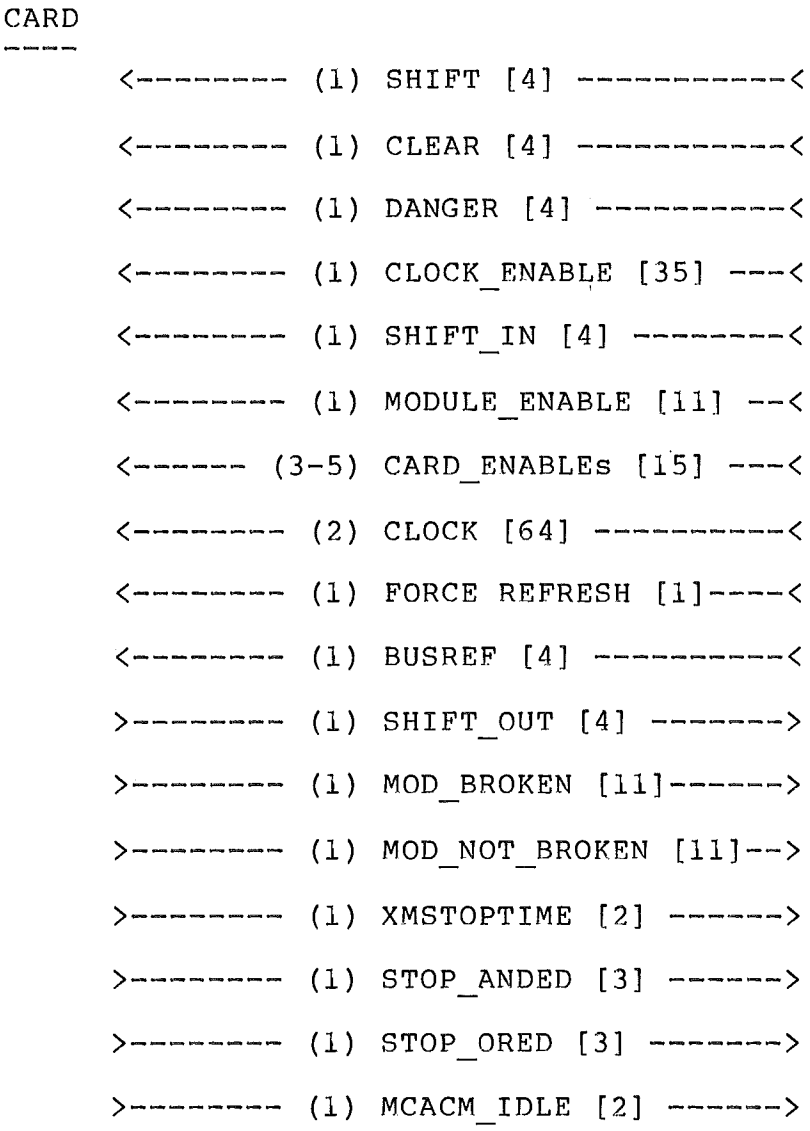
The following signals control the direction of data flow; a 1 puts the chip in transmit mode, and a Ø puts the chip in receive mode.

The signals and the paths they control are:

Signal	Path	Bits
DC_BP1	Microprocessor or backplane data path	(15:Ø)
DC_BP2	' ' ' '	(31:16)
DC_BP3	' ' ' '	(38:32)
DC_RAM1	RAM data path	(31:Ø)
DC_RAM2	' '	(38:32)

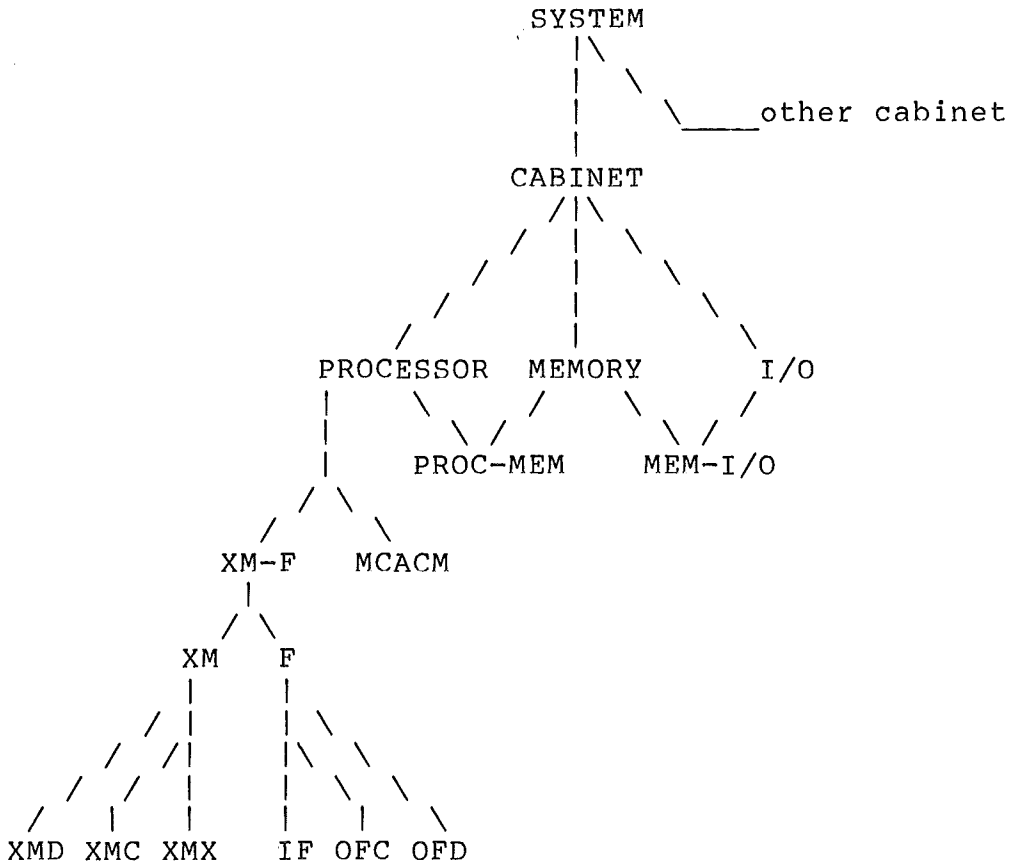
9 MAINTENANCE BUS INTERFACE DIAGRAMS

9.1 SMC TO PROCESSOR/MODULE/CARD INTERFACE



Numbers in parentheses indicate the number of signals; numbers in square brackets indicate the number of fanouts.

9.2 PROCESSOR MAINTENANCE GROUP HIERARCHY



Any node can be clocked as a group; for example, MEM-I/O, XM, or OFC. This drawing in included as a suggested grouping for maintenance processor implementation of starting and stopping clocks.

9.3 I/O BUS HAMMING CODE

data bits	ECC bits 6	5	4	3	2	1	0
00	X	X			X		
01	X	X					X
02	X	X				X	
03	X	X	X				
04	X		X	X			
05		X	X		X		
06	X		X				X
07	X		X			X	
08		X	X				X
09		X	X			X	
10			X	X	X	X	X
11	X		X		X		
12				X	X	X	
13				X		X	X
14			X	X			X
15		X		X			X
16	X			X		X	
17	X			X	X		
18	X	X		X			
19		X	X	X			
20			X		X		X
21	X	X	X	X	X		
22		X			X	X	
23	X				X	X	
24		X			X		X
25	X				X		X
26			X		X	X	
27				X	X		X
28					X	X	X
29		X				X	X
30	X					X	X
31			X			X	X

This code is created by selective exclusive ORing of the data bits. The data bits marked with X are used in the creation of the Hamming bit.

9.4 I/O BUS ERROR DETECTION DECODE

syndrome bits 654	syndrome bit values 3210															
	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
	2	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
	3	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
000		*	32	33	+	34	+	+	28	35	+	+	13	+	27	12	+
001		36	+	+	31	+	20	26	+	+	14	+	+	+	+	+	10
010		37	+	+	29	+	24	22	+	+	15	+	+	+	+	+	+
011		+	08	09	+	05	+	+	+	19	+	+	+	+	+	+	+
100		38	+	+	30	+	25	23	+	+	+	16	+	17	+	+	+
101		+	06	07	+	11	+	+	+	04	+	+	+	+	+	+	+
110		+	01	02	+	00	+	+	+	18	+	+	+	+	+	+	+
111		03	+	+	+	+	+	+	+	+	+	+	+	21	+	+	+

* = No bits in error
\+ = Multiple errors

In this table, the leftmost columns (654) are the 3 most significant bits of the syndrome. The 4 least significant bits of the syndrome are across the top (0000 on the left, 1111 on the right).

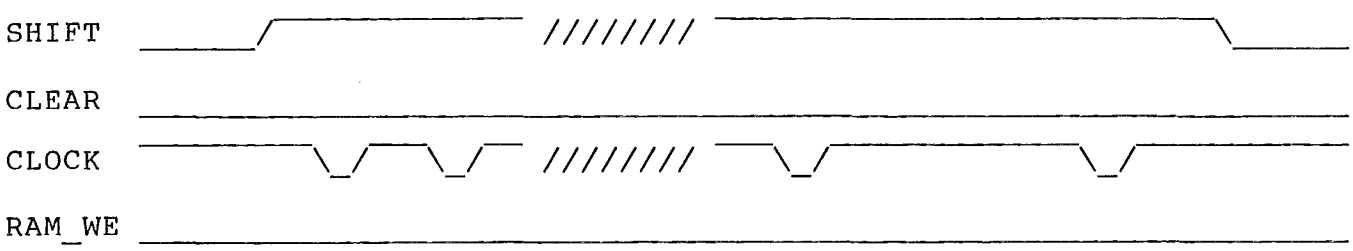
As an example, a syndrome bit value of 000 0001 indicates that there was a single bit error on ECC bit 32; that error was corrected. A syndrome bit value of 001 0001 indicates a multiple bit error. The failing bits cannot be identified; there is no correction.

The syndrome is generated based on 32 data bits, plus 7 ECC bits. Single bit errors are correctable by the ECCIO chip; multiple bit errors are not.

9.5 TIMING DIAGRAMS

Module select, card select, and clock enables are not shown. They must be in the proper state to enable a given chain.

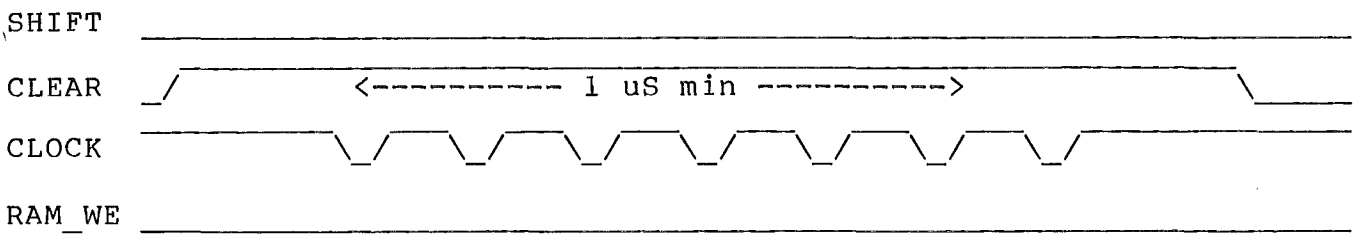
9.5.1 SHIFTING A CHAIN



All signals are active high; the rising edge of the clock is the decision edge.

SHIFT is asserted (turned on) when clocks are not running. Some time later, clocks are emitted and the chain shifts. When the chain has been shifted completely around with any data insertions or changes, and clocks are off, then SHIFT will be negated. Precautions must be taken in the module design to avoid scratchpad type RAM corruption when SHIFT is true (e.g. gate SHIFT/ with the RAM write enable).

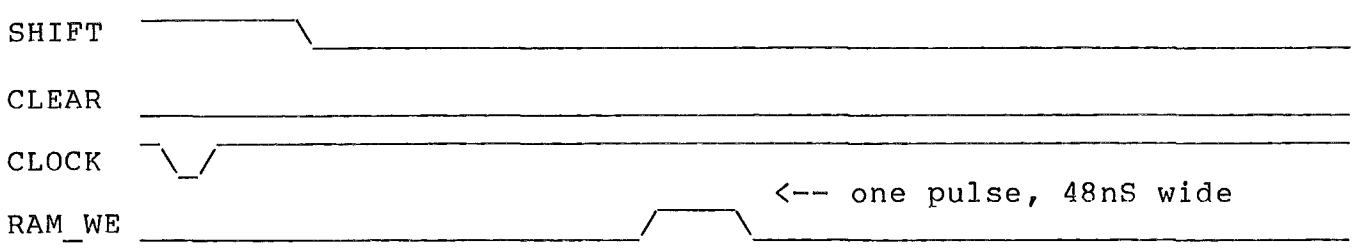
9.5.2 CLEARING A CHAIN



All signals are active high; the rising edge of the clock is the decision edge.

CLEAR is asserted when clocks are not emitted. Clocks are then emitted for at least lusec, then CLEAR is negated.

9.5.3 WRITING CONTROL STORES



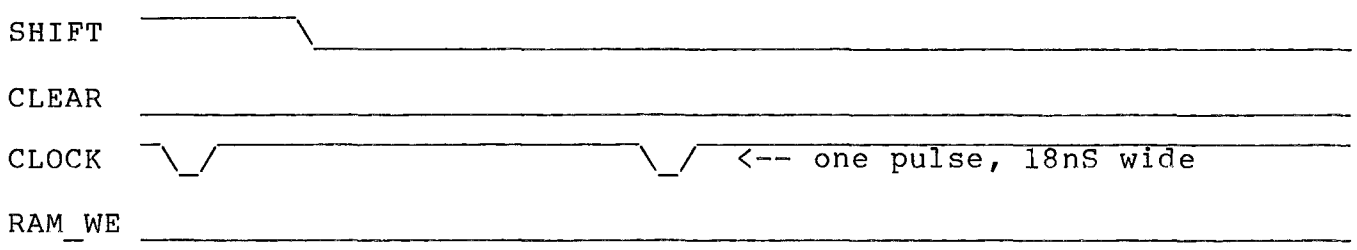
All signals are active high; the rising edge of the clock is the decision edge.

The setup data is shifted into the chain, then SHIFT goes false. RAM_WE is then asserted for one system clock period.

Later, SHIFT (restore the chain) or CLEAR may be asserted. Any CRAM select data may be ignored, because the chain will either be restored, or CSLOAD will continue, or a CLEAR will happen.

Note that RAM_WE is derived from the DANGER signal; it can be TRUE if the clock is manually skewed. To protect against CRAM corruption, an enable bit should be gated with RAM_WE so that both RAM_WE and the enable must be true to write CRAM.

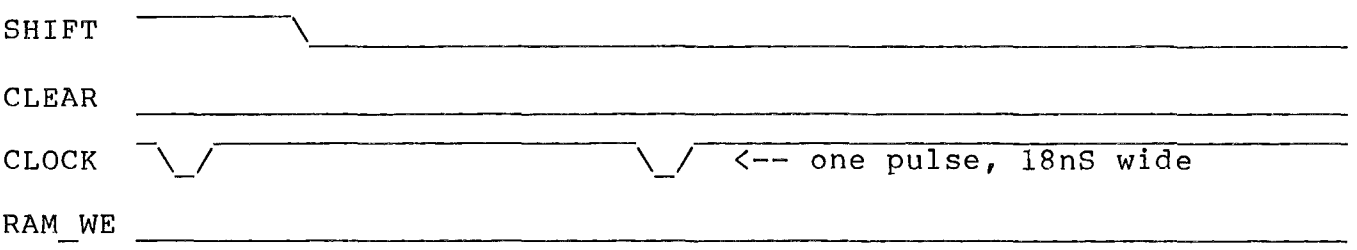
9.5.4 READING RAM



All signals are active high; the rising edge of the clock is the decision edge.

The setup data will be shifted into the chain. Some time after this, SHIFT will go false. After more time, one clock will be emitted. Later, SHIFT (read out the chain to get the data) will happen. Still later, either CSVERIFY will continue or the chain will be restored by commands from the MP to the SMC.

9.5.5 WRITING SCRATCHPAD TYPE RAMS

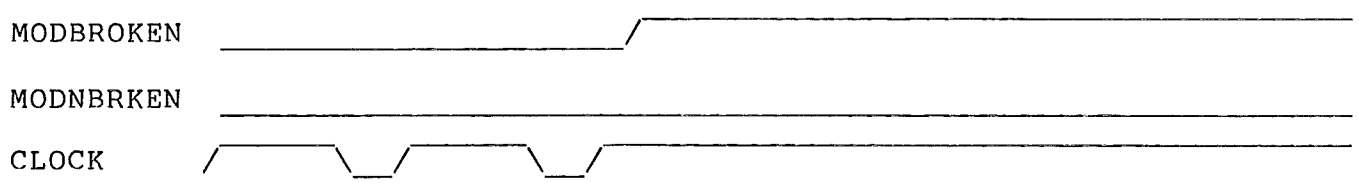


All signals are active high; the rising edge of the clock is the decision edge.

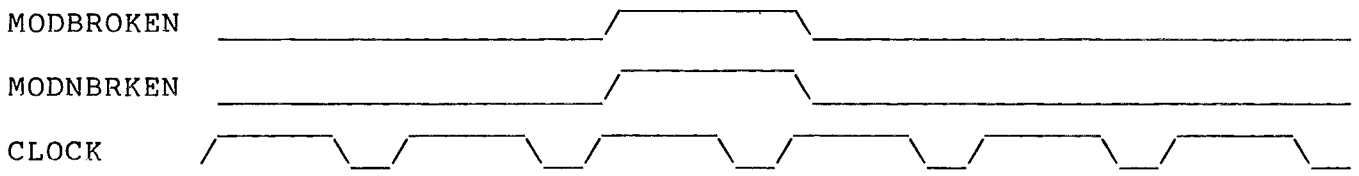
The setup data will be shifted into the chain. After this, SHIFT will go false, then one clock will be emitted. Later, SHIFT will be asserted to restore the chain.

9.5.6 ERRORS AND NON-ERRORS

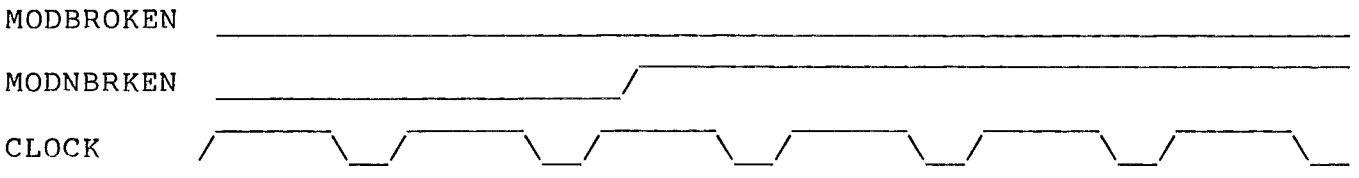
CASE A



CASE B



CASE C



CASE A: A MODBROKEN condition causes the clock to stop.

CASE B: When both MODBROKEN and MODNBRKEN conditions are active, MODNBRKEN negates MODBROKEN and this causes the clock to continue running.

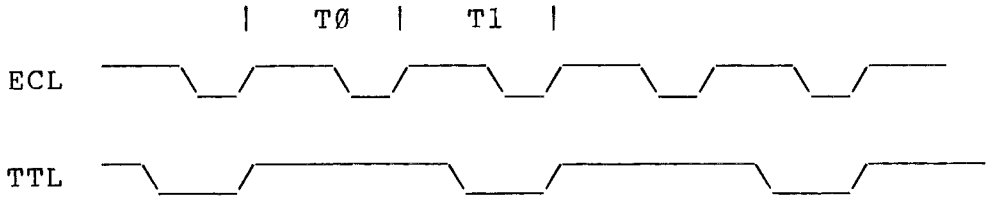
CASE C: When active, MODNBRKEN condition has no effect on the clock signal. This condition is wrong but not fatal. It is logged.

9.6 PHASE DETECTION

This section provides the functional definition of the clock phase detector and its discrete support logic in the System Maintenance Controller. The phase detector, along with the shift arrays and discrete shift control logic, provides control for the maintenance bus, stop logic, and error logic.

The V500 uses both ECL and TTL logic. Because ECL operates at twice the rate of TTL, the system must have a method of synchronizing the ECL and TTL clocks.

The clocks have the following relationship:



When the system is stopped, it must recall the phase relationship between the ECL and TTL clocks. By so doing, it can re-establish the phase relationship when it is restarted, and thereby maintain the clock synchronization on TTL and ECL boards.

The phase detection logic controls the clock enable lines. Boards that have both TTL and ECL logic have one clock enable line for TTL clock control, and a separate enable line for ECL clock control. The ECL clock enable on the SMC is generated from the A_SHIFTREG_H signal.

The clock enable scheme for V500 single and dual cabinet systems is as follows:

1. All boards that are not mixed logic boards will have one clock enable line.

2. Boards with both TTL and ECL logic will have two clock enable lines.

9.6 PHASE DETECTION (Continued)

These two rules apply except for the following:

signal	single cab (1)	dual cab (2)
AIØENIØMC1$P	IOMC1	IOMC1
AIØENIØMC2$P	IOMC1	IOMC1
BIØENIØMC1$P	IOMC1	IOMC1
AØØENICBM$ØP	IOMC2	ICBM
AØØENICBM$1P	IOMC2	ICBM
BIØENIOMC2$P	IOMC2	not used

1. The maximum single cabinet configuration consists of two IOMC modules and no ICBM module.

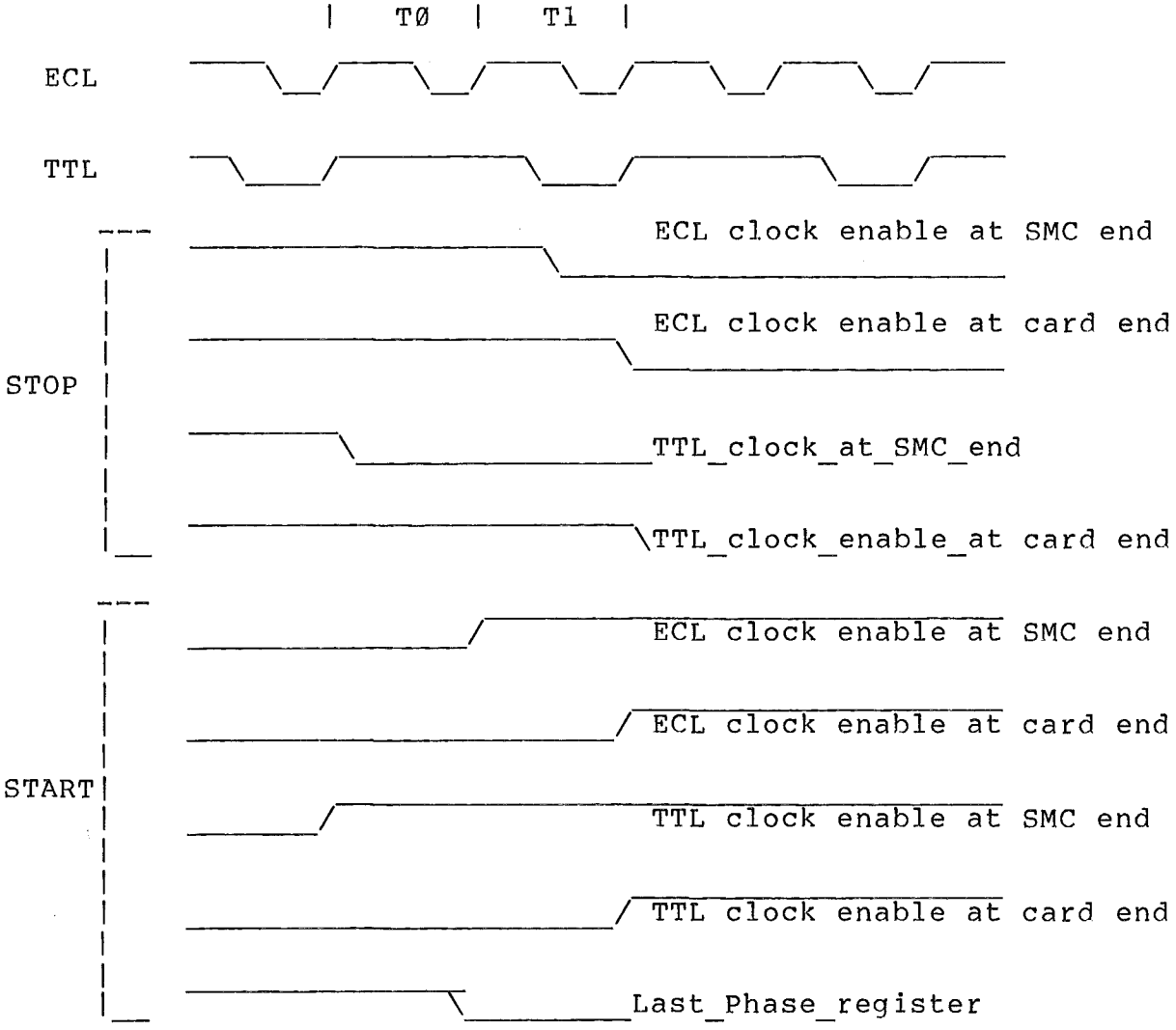
2. The required configuration for two cabinet configurations is 1 IOMC module in each cabinet, and 1 ICBM module in the system.

The table shows that AØØENICBM$ØP and AØØENICBM$1P are used by IOMC2 in single cabinet configurations, and by the ICBM in dual cabinet configurations.

The following timing charts illustrate the timing relationship that must be maintained in order to keep the ECL and TTL boards synchronized.

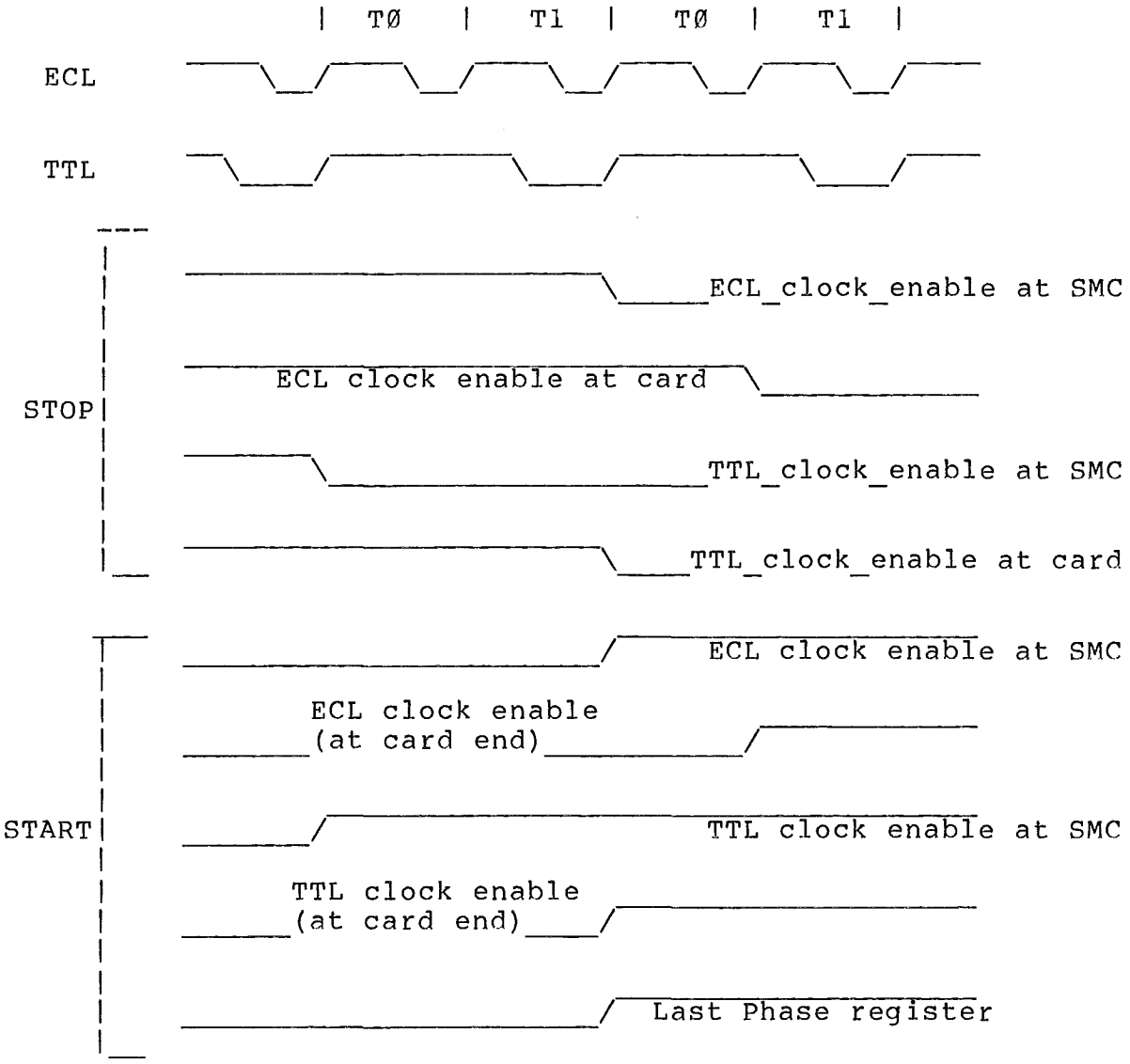
9.6 PHASE DETECTION (Continued)

The first chart shows a stop at the end of the TØ phase and a start at the beginning of the T1 phase.



9.6 PHASE DETECTION (Continued)

This chart shows a stop at the end of the T1 phase and a start at the beginning of the T0 phase.



9.6 PHASE DETECTION (Continued)

In the previous timing charts, the falling edge of the ECL clock enable at the SMC end is the reference point for determining the last phase at the time the clocks were stopped.

As shown in Table 1, the number of TTL clocks can be adjusted by +1 or -1 as required. This is done automatically and requires no software intervention. The counter in the shift array is loaded with twice the value, because each TTL clock spans 2 ECL clocks.

Plan	Number of clocks	Last Phase transition	Clock enable becomes active
A	odd	T0-T1	ECL: 2 clocks after TTL clock enable. TTL: After number of clocks in shift array - 1.
B	odd	T1-T0	ECL: 1 clock after TTL clock enable. TTL: After number of clocks
C	even	T0-T1	ECL: 1 clock after TTL clock enable. TTL: After number of clocks in shift array.
D	even	T1-T0	ECL: 2 clocks after TTL clock enable. TTL: After number of clocks in shift array.

Table 1 TTL & ECL Clock Enable Schemes

An example of each of the conditions is included in Appendix A. These simulations are done for both odd and even numbers of clocks using the clock burst function, and also for single clocking TTL and ECL cards.

9.6 PHASE DETECTION (Continued)

The software must set the single clock bit in the ECL address register at location 30200h. The register is loaded via ECL latched address lines. The single clock bit is required in order to load the shift array counter.

The ECL address register is defined as follows:

Bit	Function
0	ECLREAD/INS
1	ECLSHIFTH (was ECLFULL/HALF)
2	ECLCTRCLR
3	CLKERR0REN1H
4	CLKERR0REN2H
5	SINGLECLK
6	not used
7	not used

The Last Phase register is a part of an external register for the SMC board. The register's address is 581h; it is treated as an I/O port. The software can read and write the register (in parallel) from the microprocessor.

The Last Phase bit can be read, set, or reset by the microprocessor, depending on the type of operation. For example, to start from phase T0 after clearing the system, write a zero to the Last Phase bit.

Note that in a shift, the Last Phase register is not loaded with the last phase that the clocks stopped.

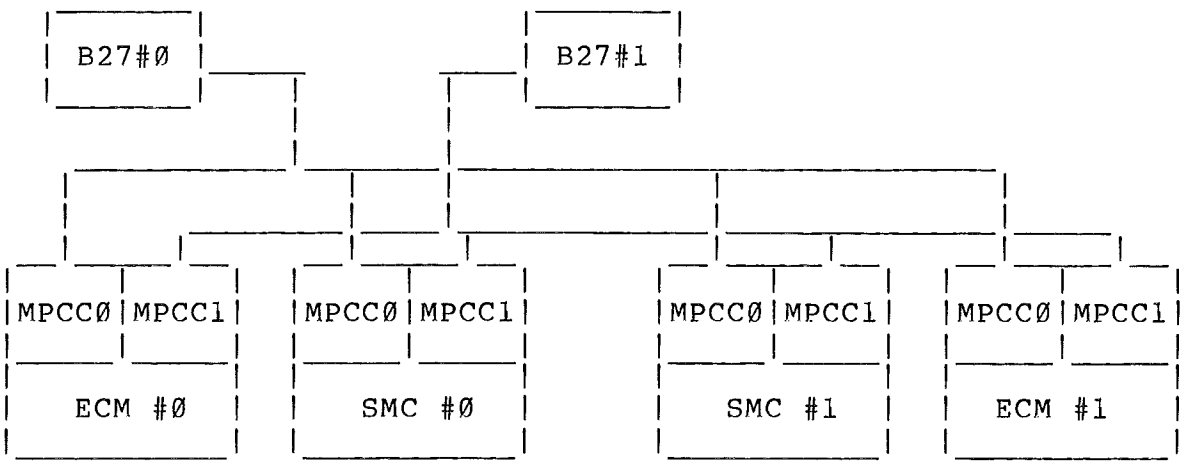
See the clock subsystem specification for dual cabinet features. The ability to stop and start clocks in different cabinets at the same time is not available.

9.6 PHASE DETECTION (Continued)

Bits in the External Register are used as follows:

Bits	Function
0, 1	IOMC2 and ICBM error registers; same as other registers in ER_STP_REG.
2, 3	PROHALT1 is from the MS slice of the shift array; PROHALT2 is from the LS slice of the shift array. These signals can be read to determine if the source of the INTERRUPT2 on the SMC 80186 processor is generated from them or not.
4	MDNBRKSTS is the Module Not Broken status register. It indicates whether the error condition is due to Module Not Broken signals. After clearing the error registers inside and out, (bits 0 and 1 of this register), the shift array clears MDNBRKSTS.
5	The Last Phase register bit is loaded when the clocks are stopped by a toggle, count up, or stop function. This bit is loaded if the clock burst (count up) does not involve a shift chain operation. When using the toggle function to turn clocks on and off, start from the most convenient phase.

9.7 FAULT TOLERANT SYSTEM INTERCONNECT



Either MP can control the configuration. Only one MP is required. The second MP is for fault tolerance.

APPENDIX A MENTOR SIMULATIONS

Figures A1 through A10 illustrate Phase Logic simulations. In these diagrams, "PLAN" refers to the Plan shown in Table 1, and the "COUNT" value determines if the number of clocks is odd or even.

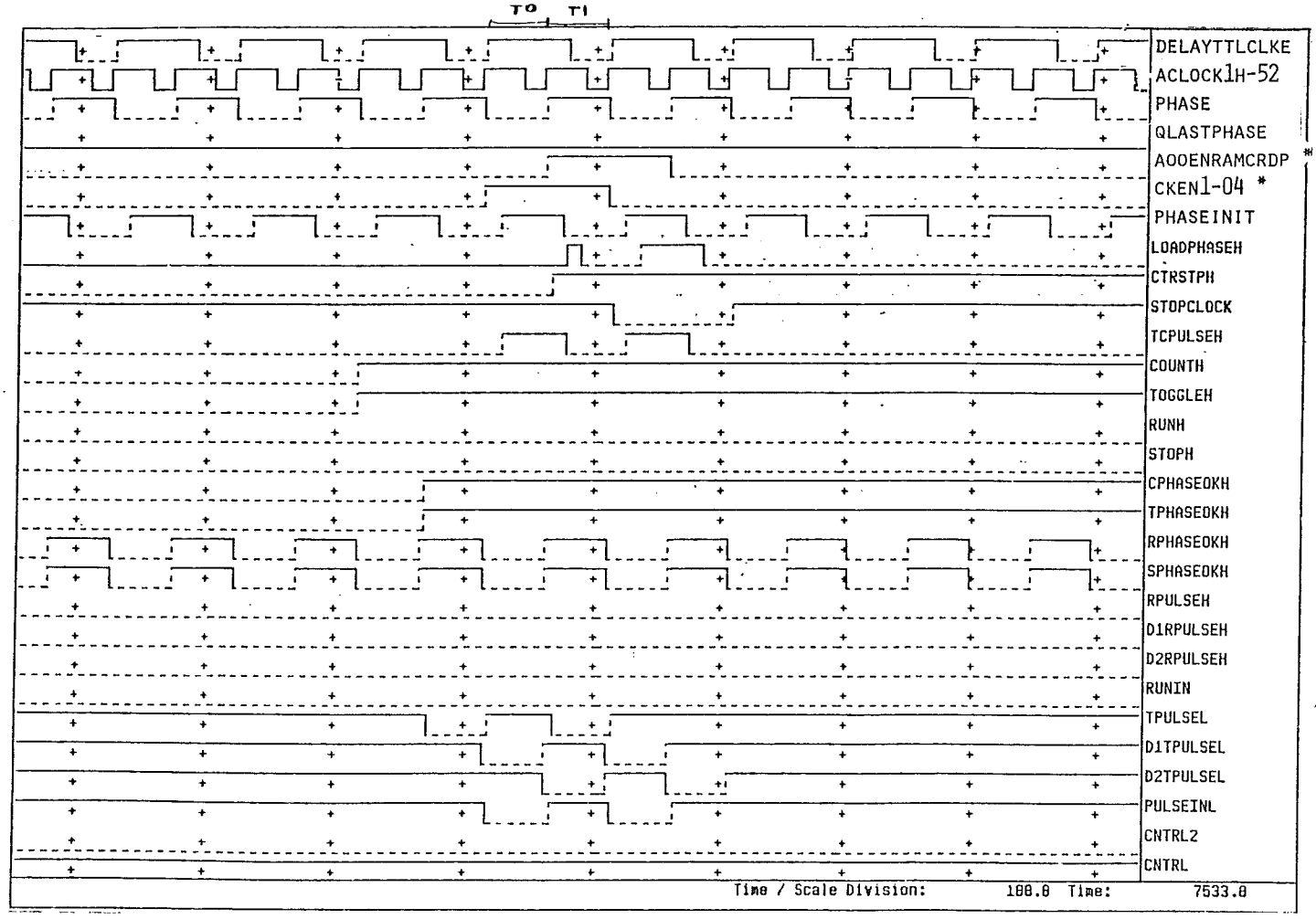


Figure A1 Clock Simulation (Plan C Count 2)

*These two signals, AOOENRAMCRDP and CKEN1-04, are representatives of ECL and TTL clock enable respectively.

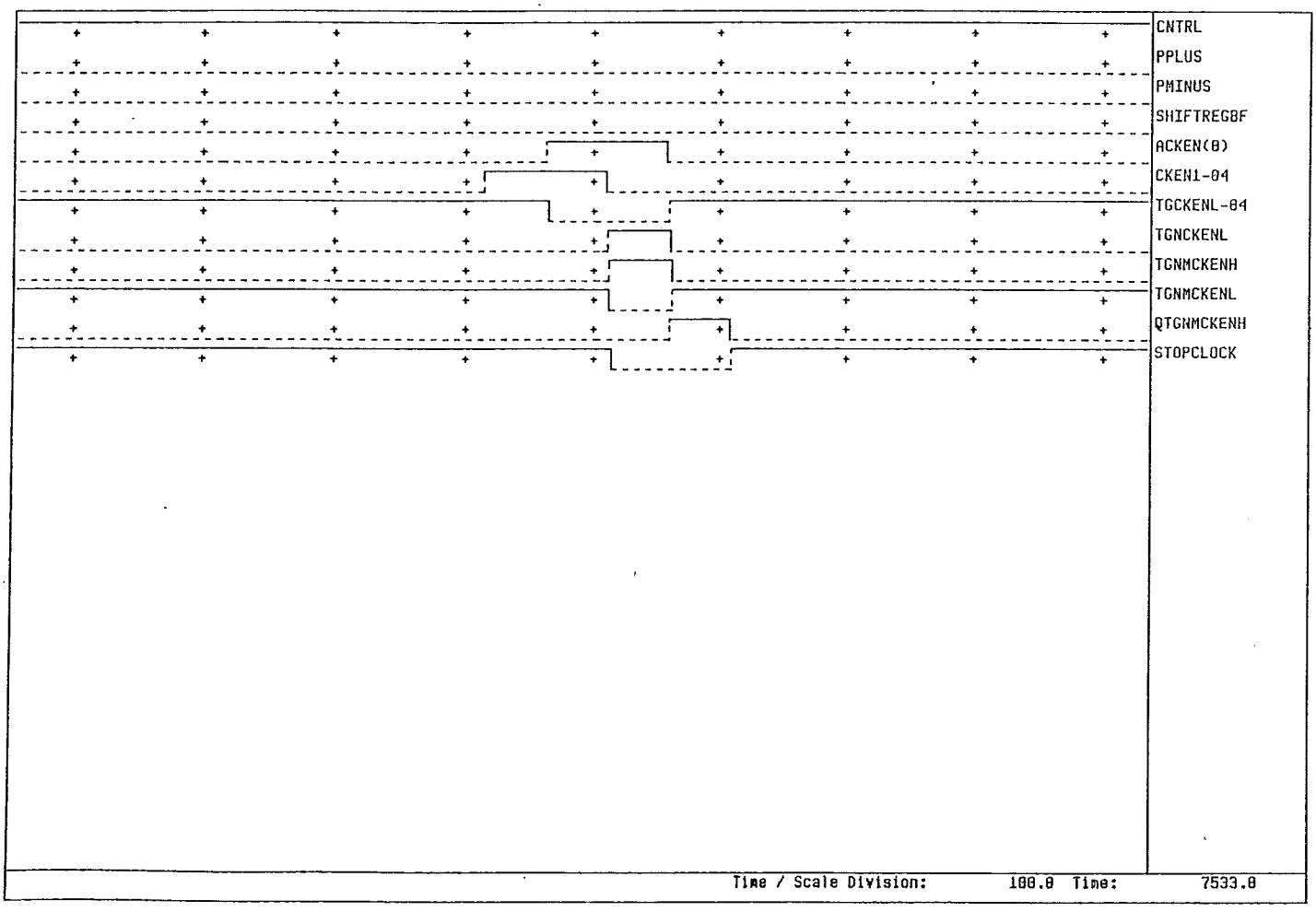


Figure A1 Clock Simulation (Plan C Count 2)

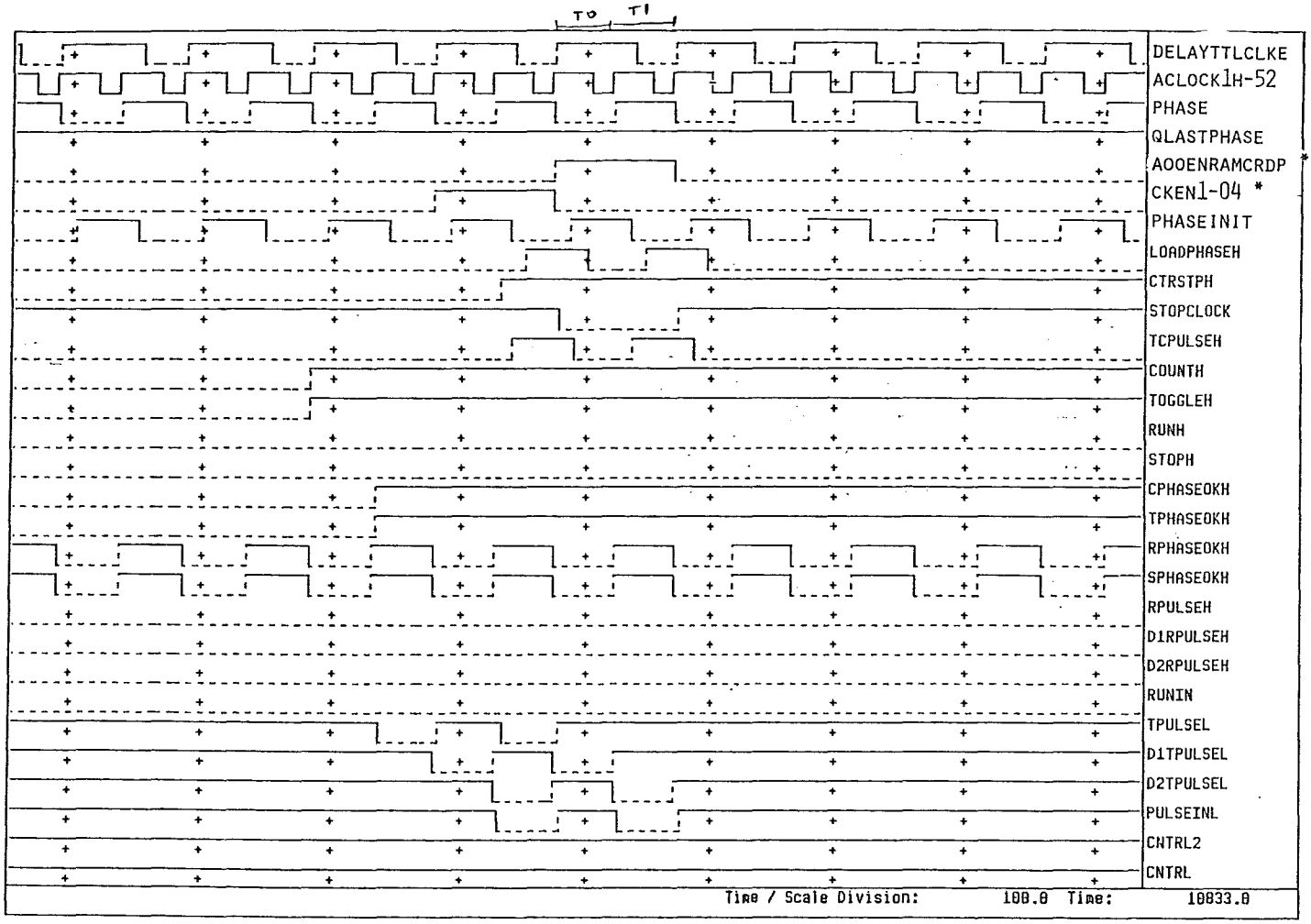


Figure A2 Clock Simulation (Plan D Count 2)

*These two signals, AOOENRAMCRDP and CKEN1-04, are representatives of ECL and TTL clock enable respectively.

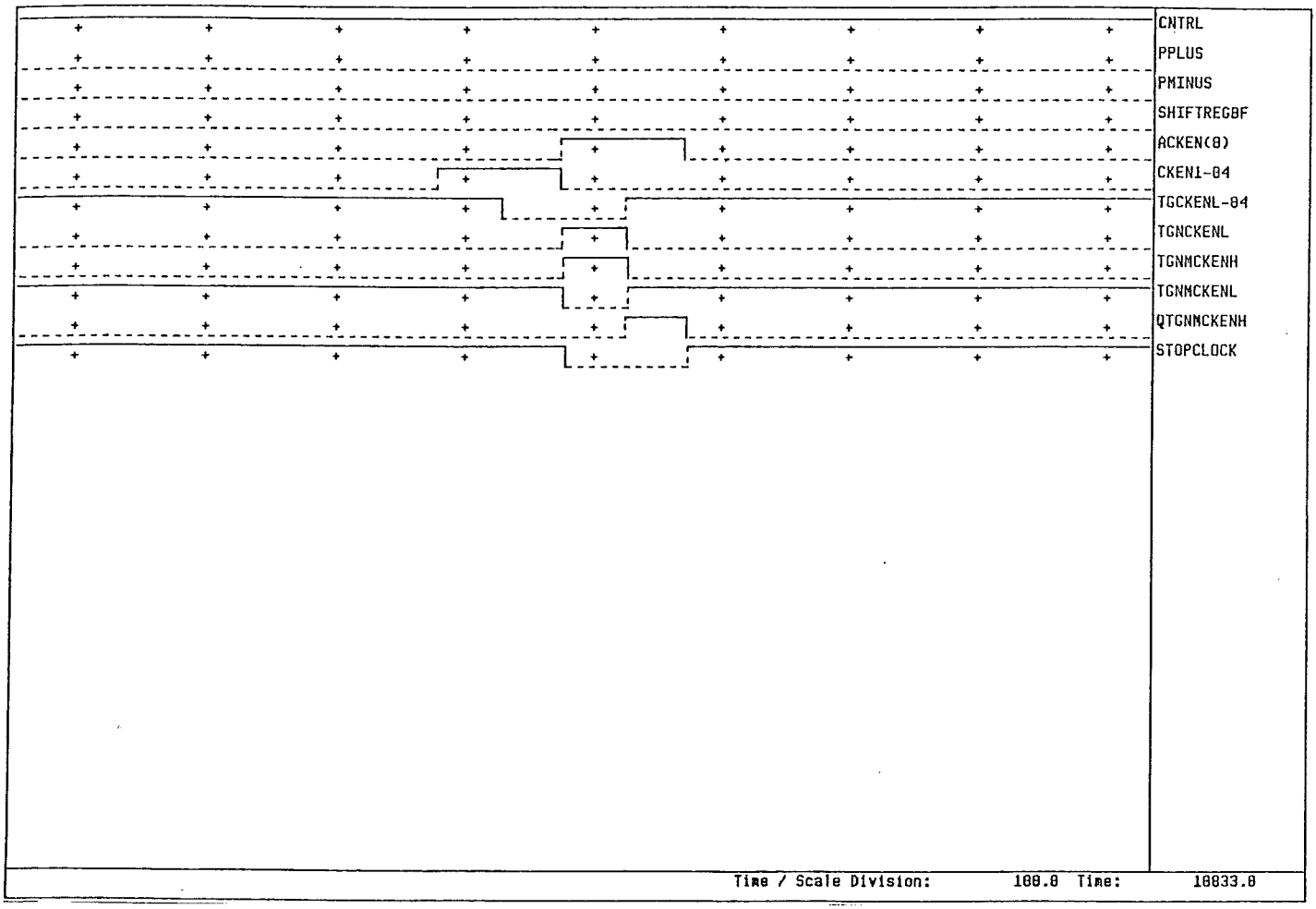


Figure A2 Clock Simulation (Plan D Count 2)

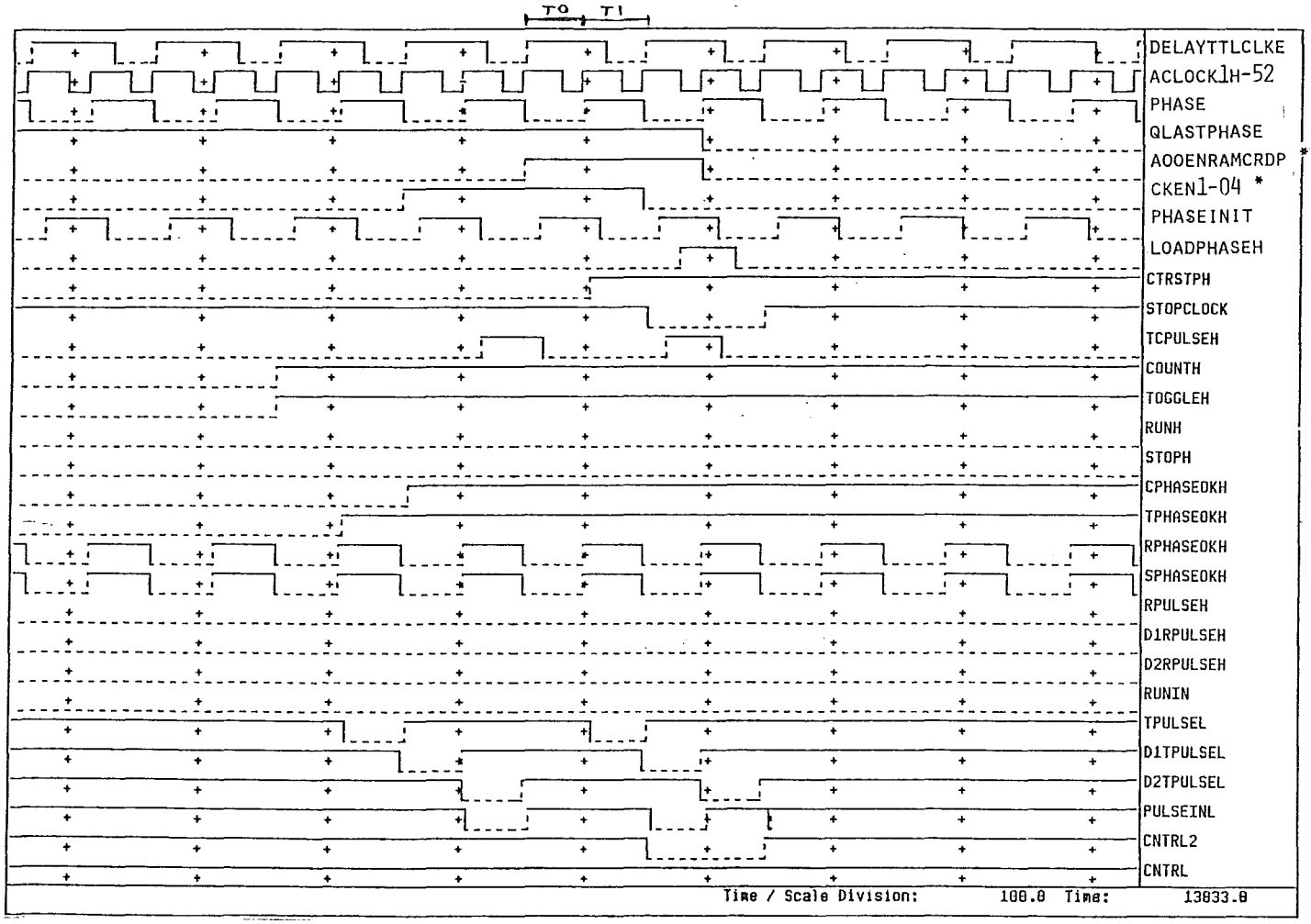


Figure A3 Clock Simulation (Plan B Count 3)

*These two signals, AOOENRAMCRDP and CKEN1-04, are representatives of ECL and TTL clock enable respectively.

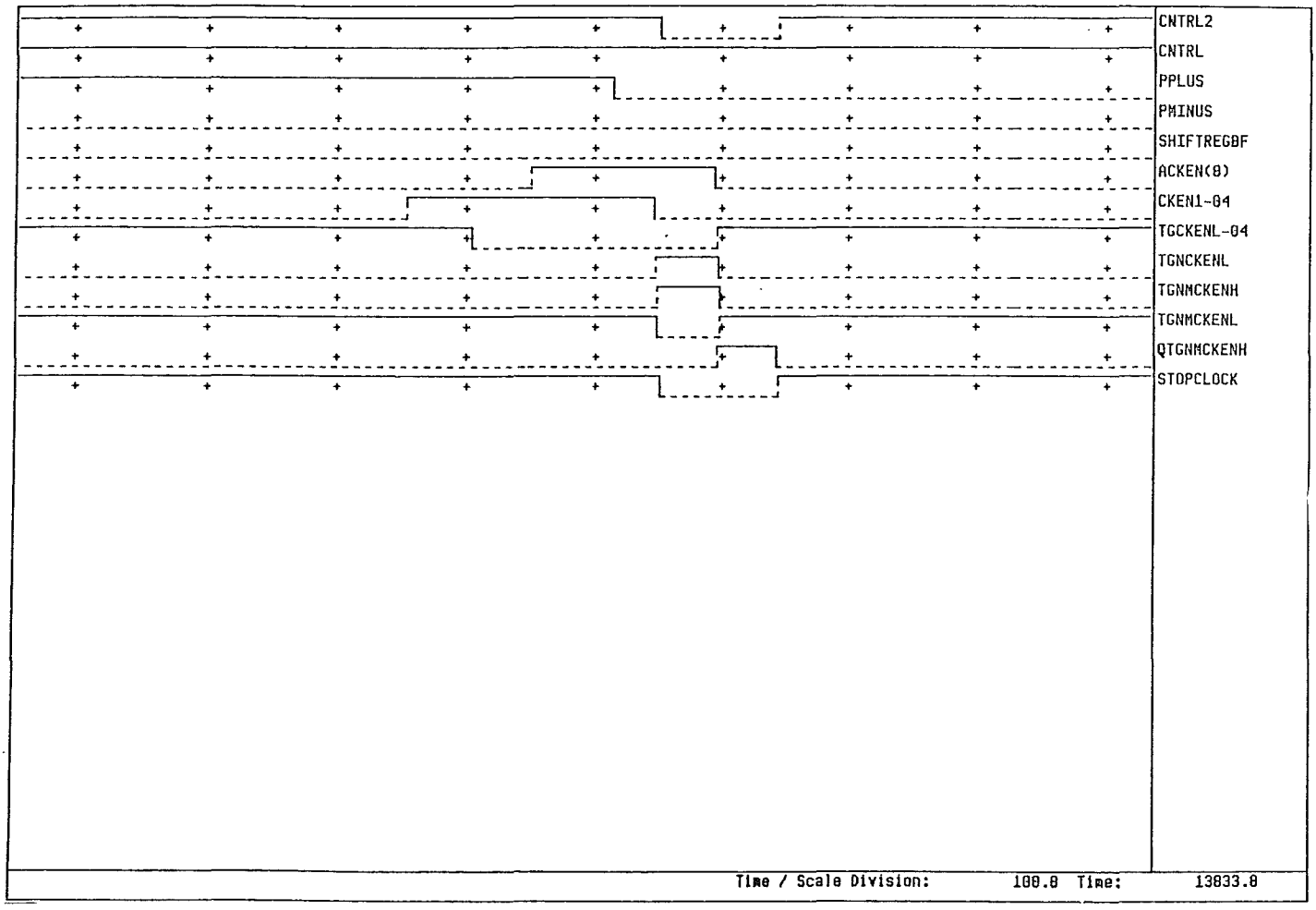


Figure A3 Clock Simulation (Plan B Count 3)

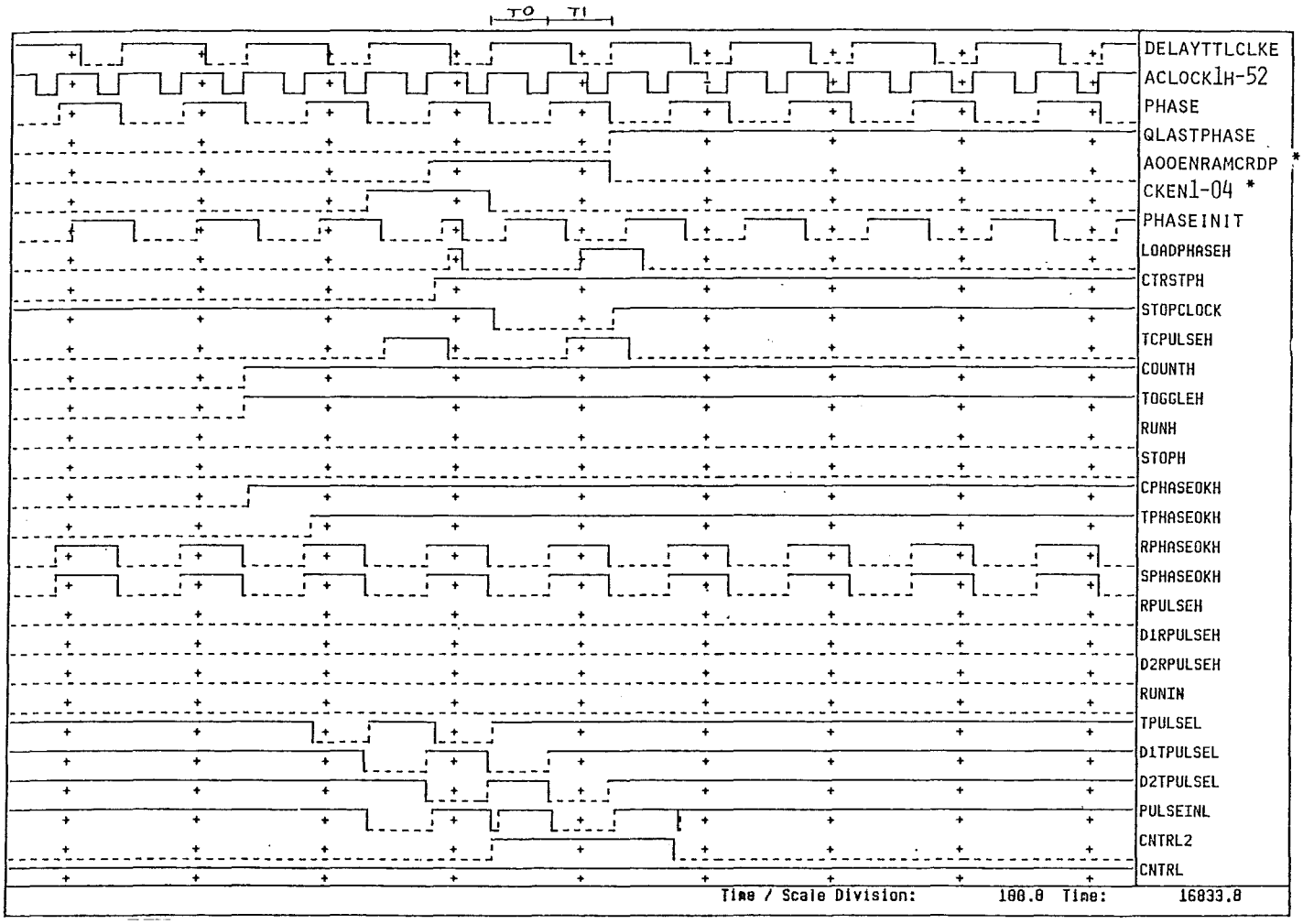


Figure A4 Clock Simulation (Plan A Count 3)

*These two signals, AOOENRAMCRDP and CKEN1-04, are representatives of ECL and TTL clock enable respectively.

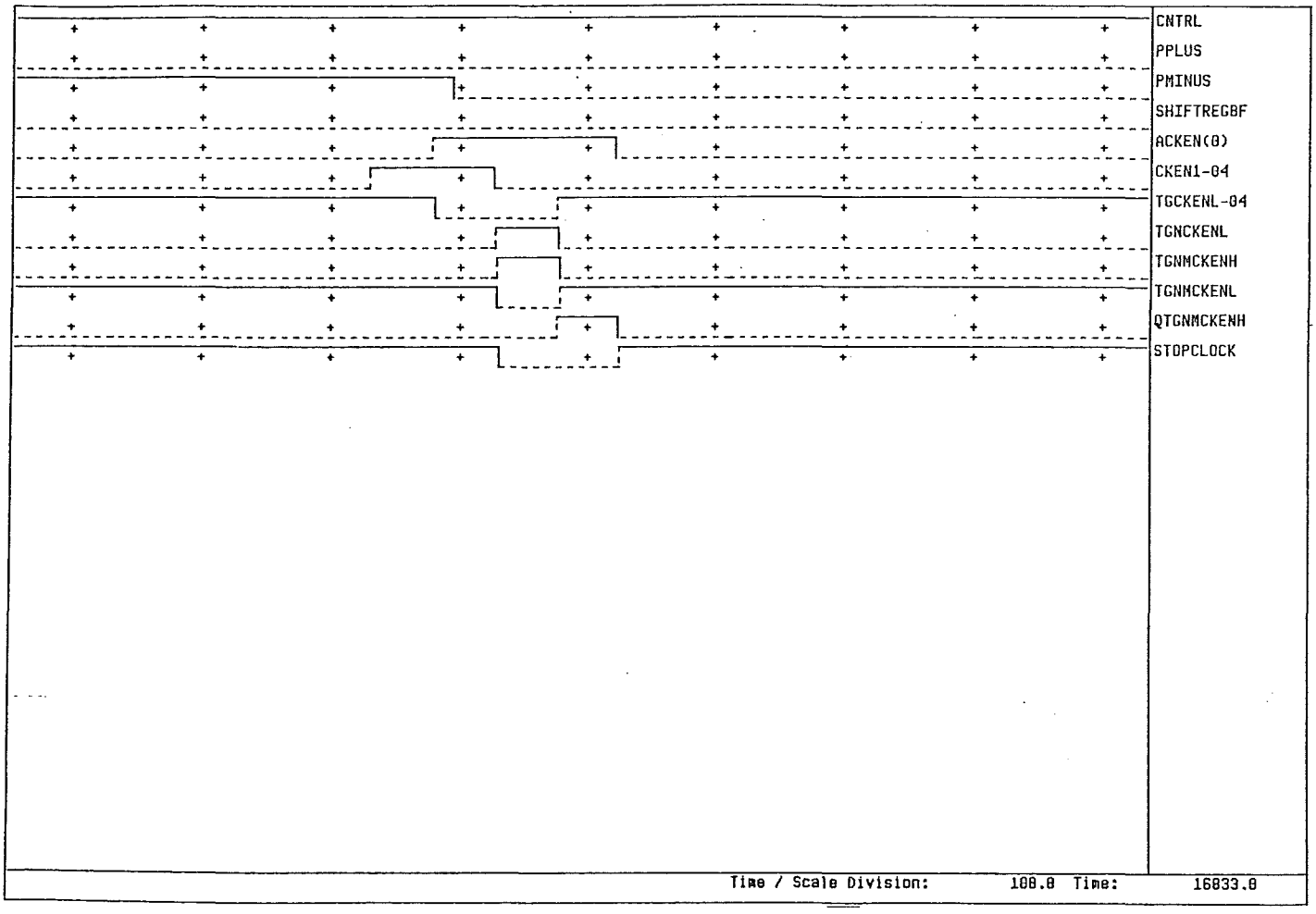


Figure A4 Clock Simulation (Plan A Count 3)

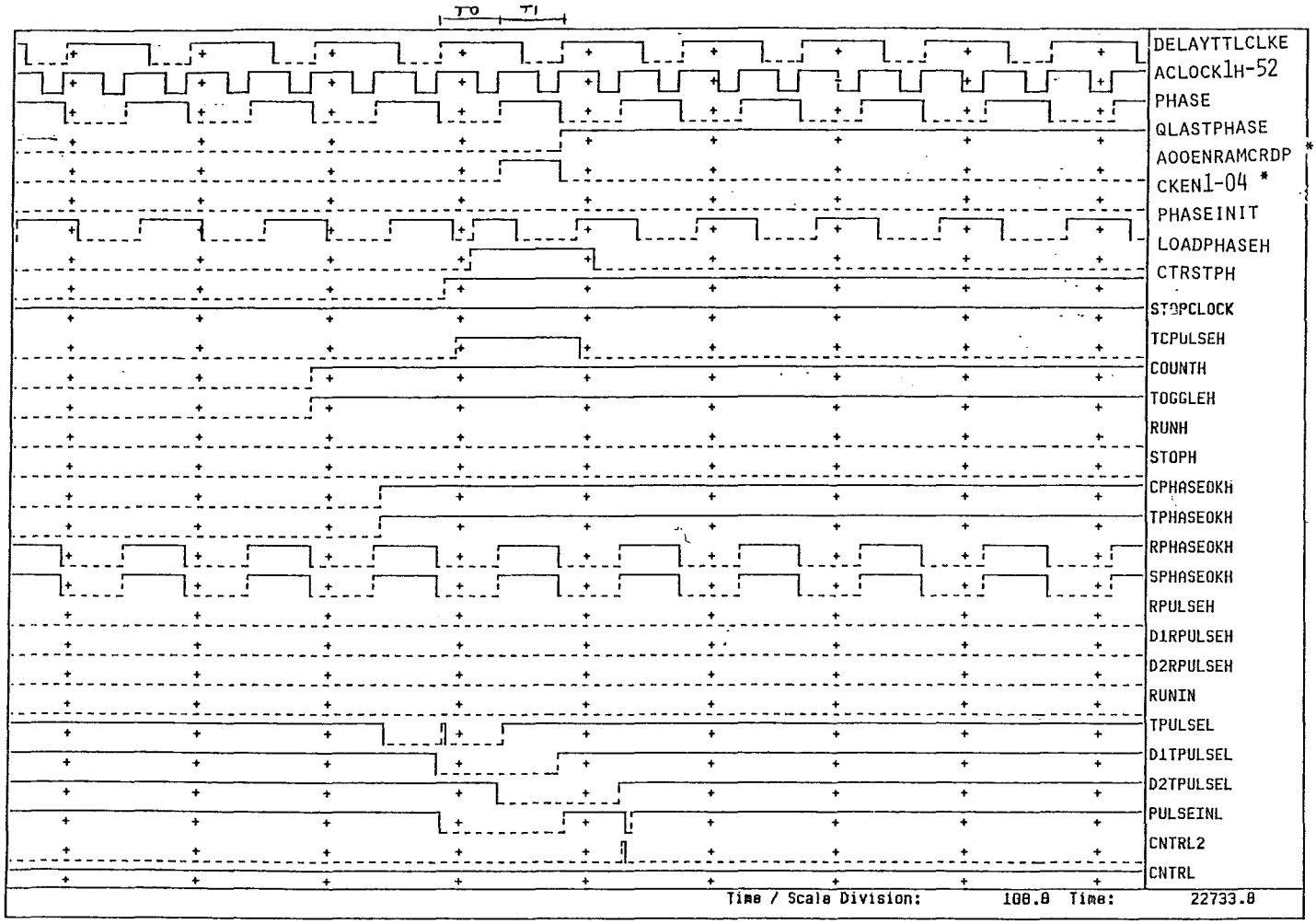


Figure A5 Clock Simulation (Plan A Count 1)

*These two signals, AOOENRAMCRDP and CKEN1-04, are representatives of ECL and TTL clock enable respectively.

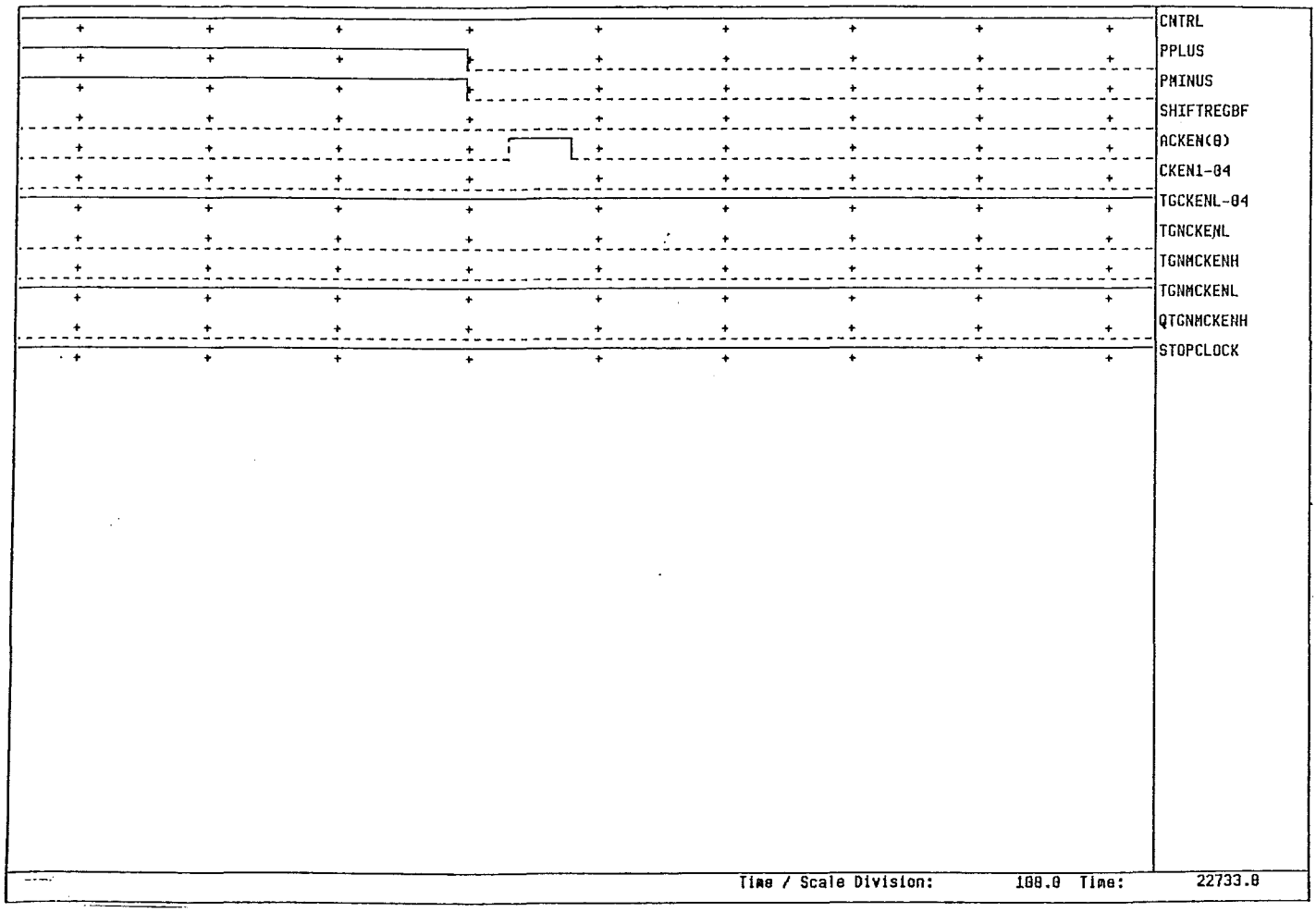


Figure A5 Clock Simulation (Plan A Count 1)

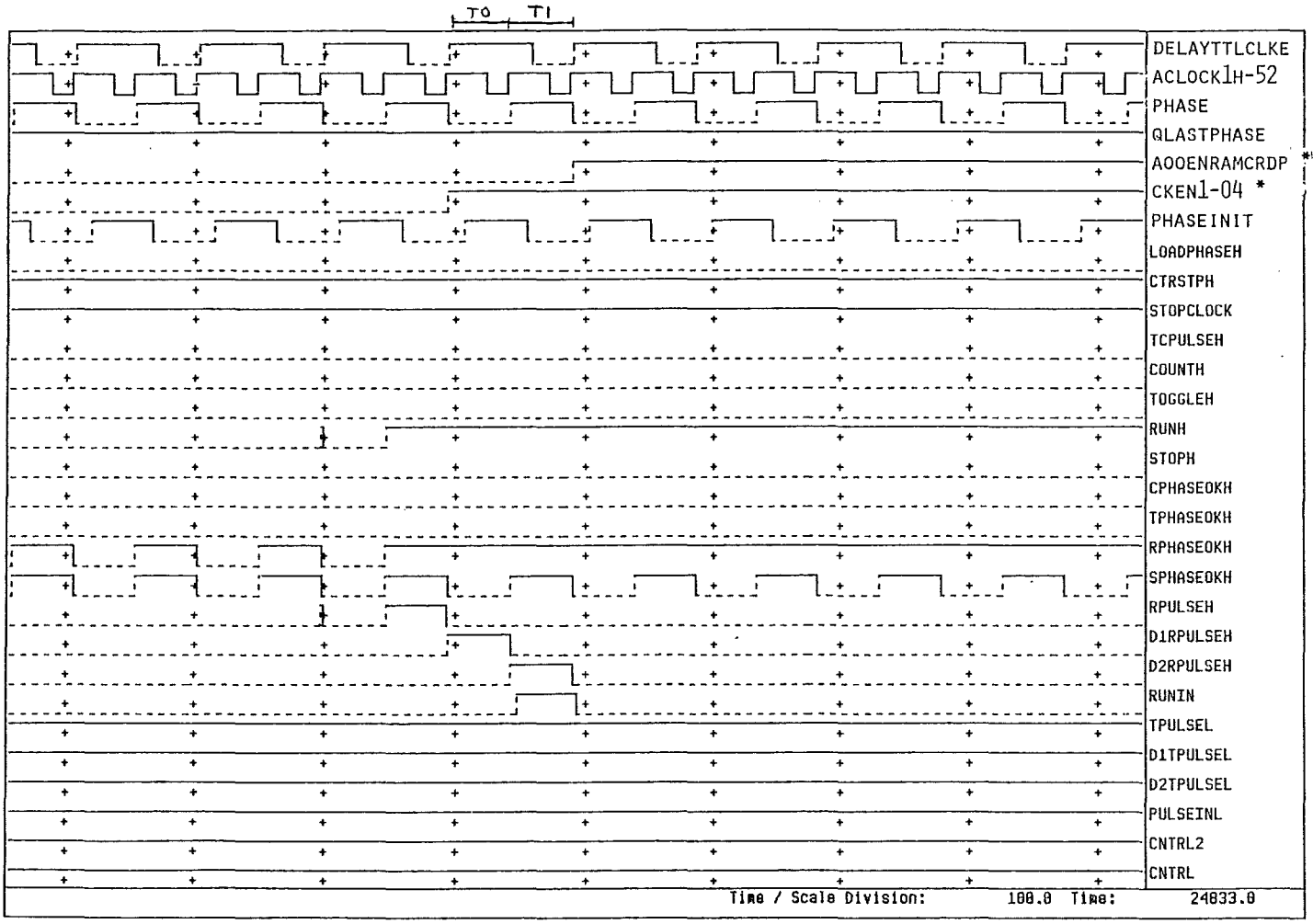


Figure A6 Clock Simulation (Clocks On At T0)

*These two signals, AOOENRAMCRDP and CKEN1-04, are representatives of ECL and TTL clock enable respectively.

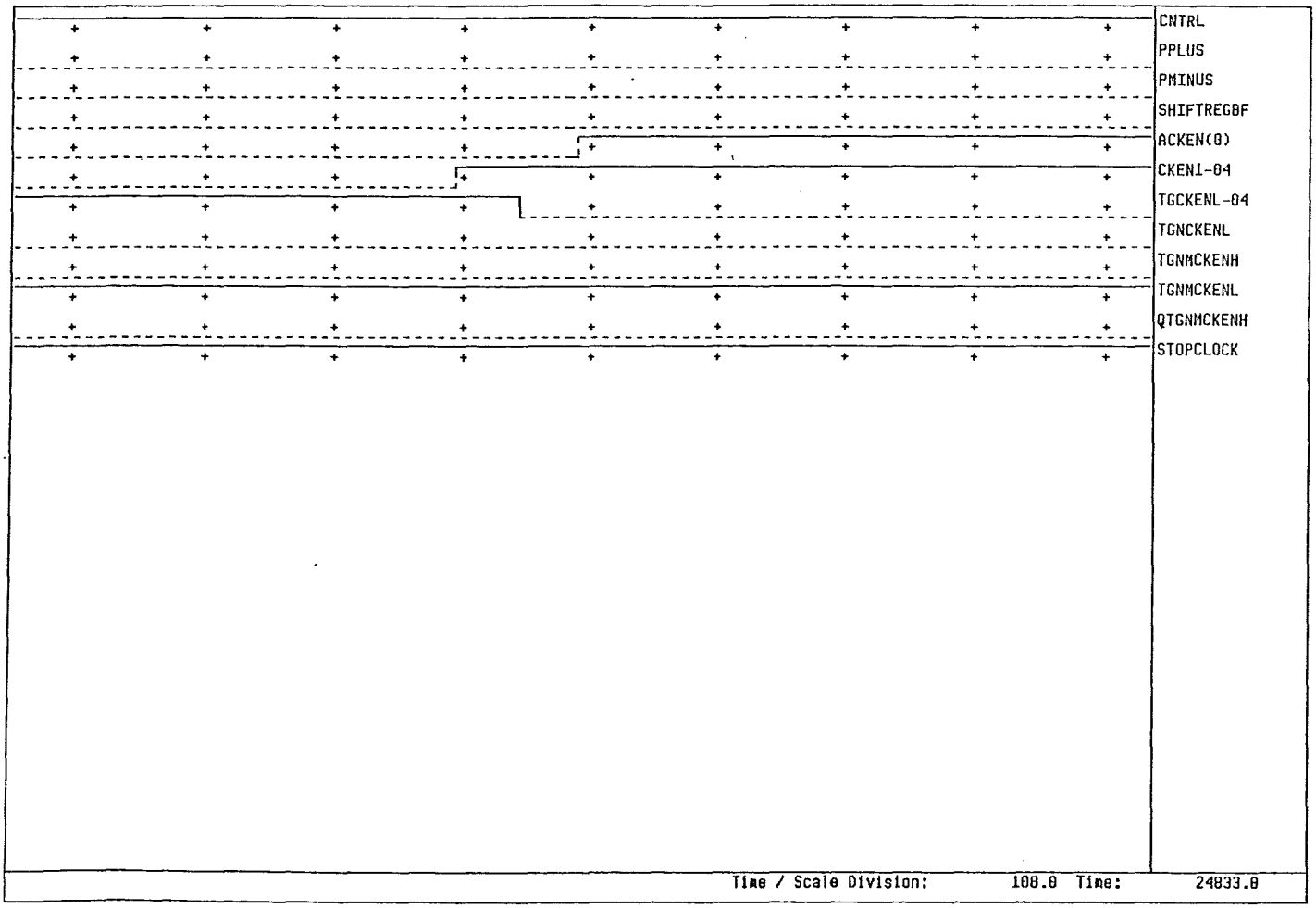


Figure A6 Clock Simulation (Clocks On At T0)

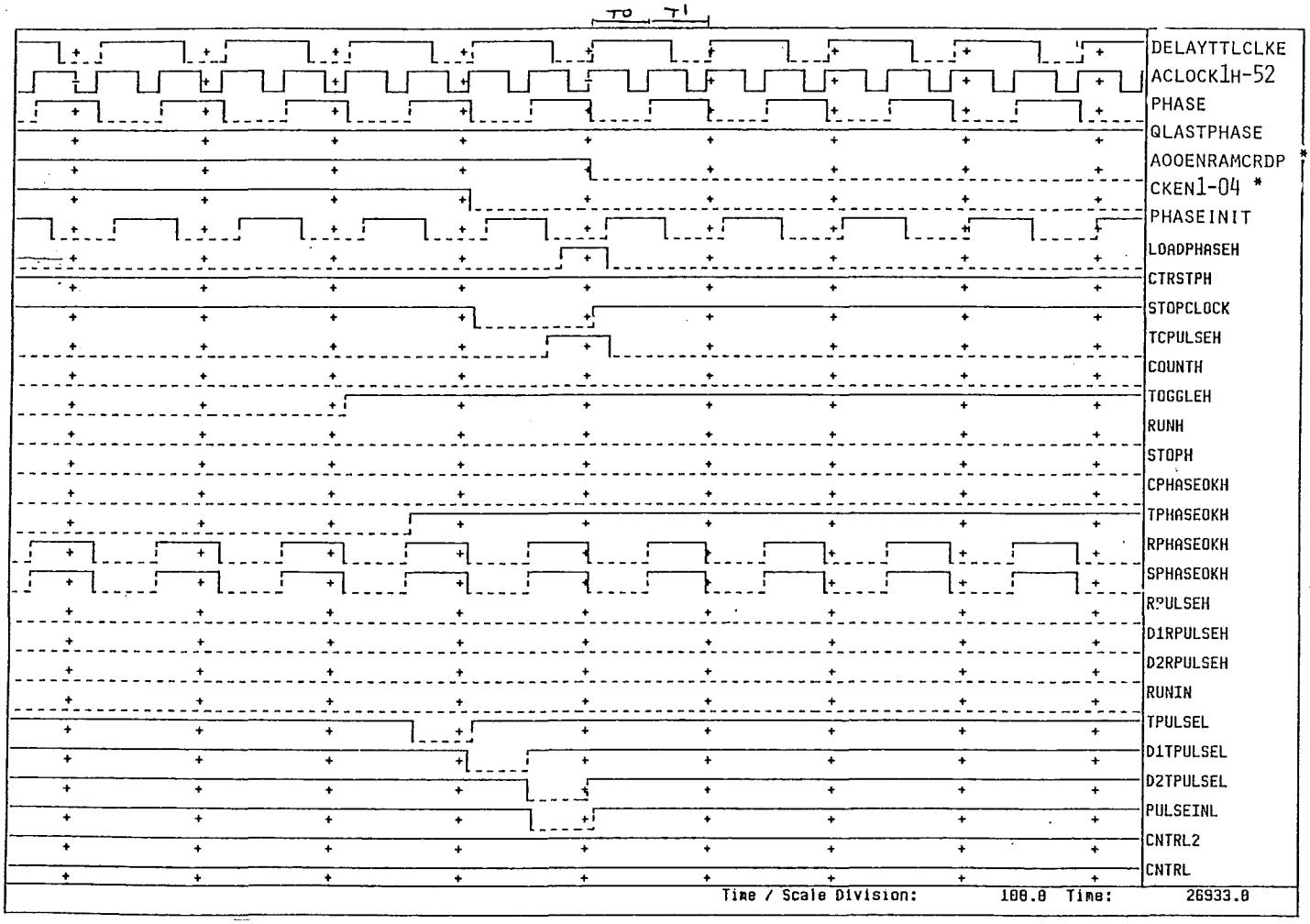


Figure A7 Clock Simulation (Toggle Stopping Clocks)

*These two signals, AOOENRAMCRDP and CKEN1-04, are representatives of ECL and TTL clock enable respectively.

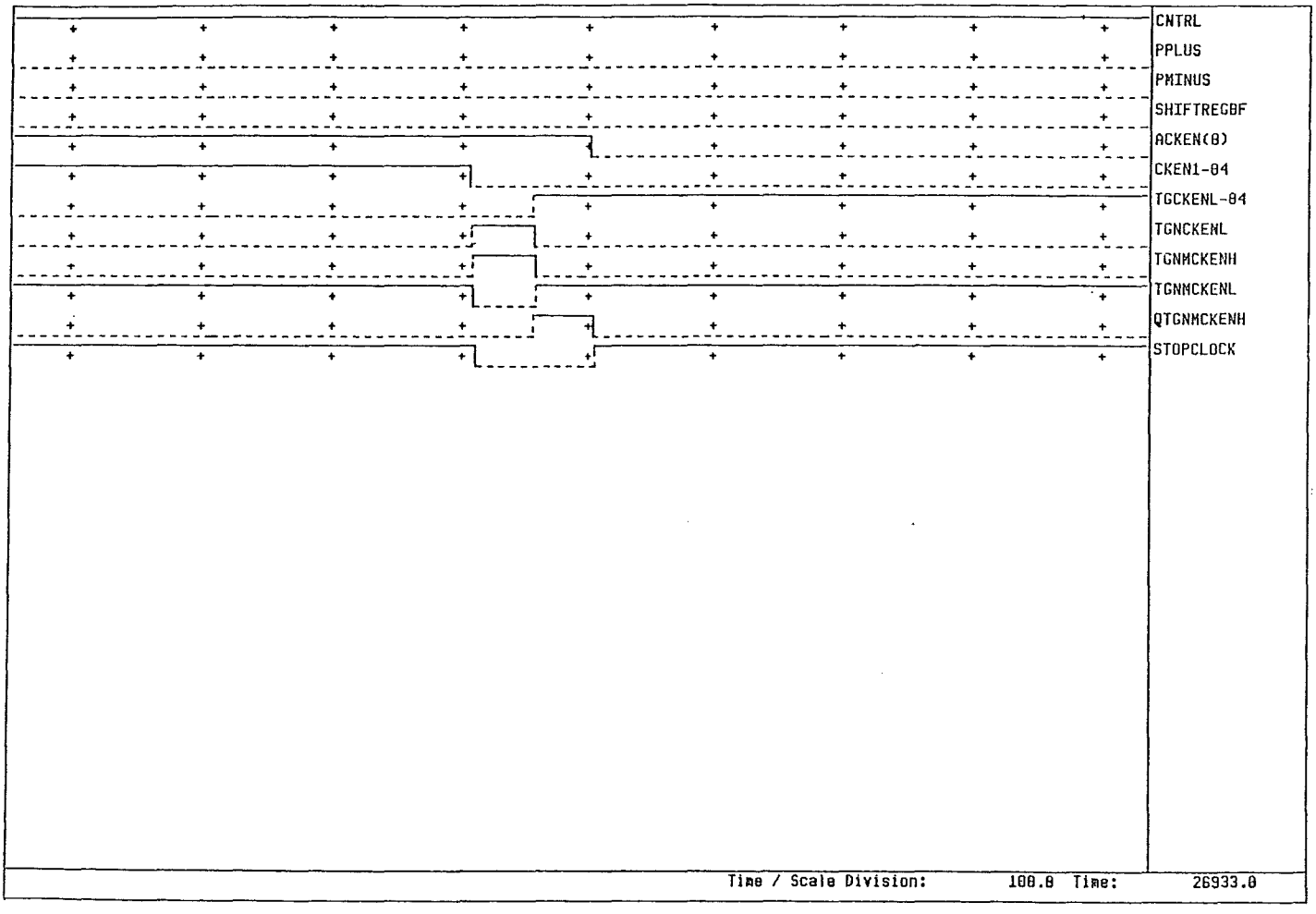


Figure A7 Clock Simulation (Toggle Stopping Clocks)

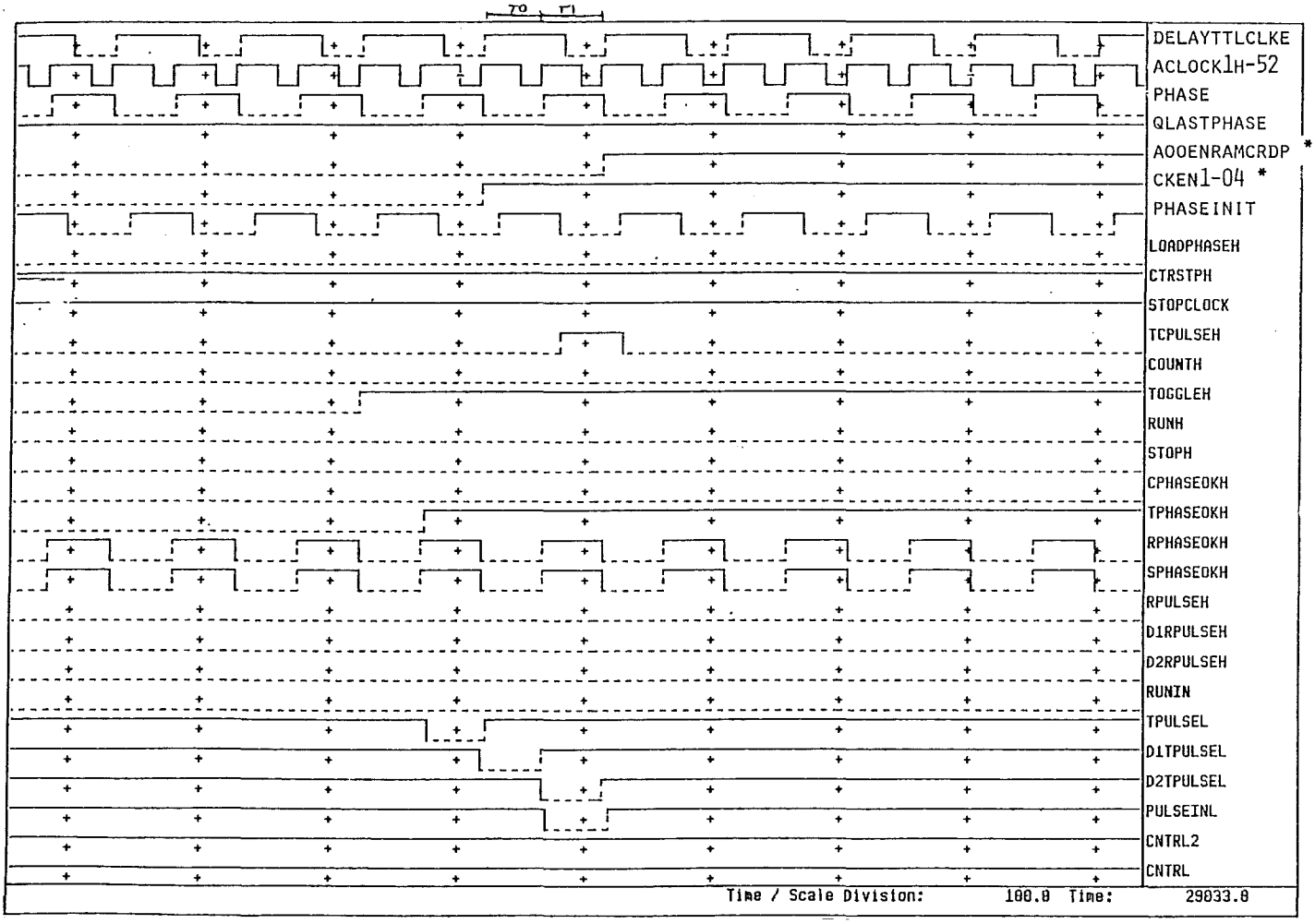


Figure A8 Clock Simulation (Toggle Starting Clocks)

*These two signals, AOOENRAMCRDP and CKEN1-04, are representatives of ECL and TTL clock enable respectively.

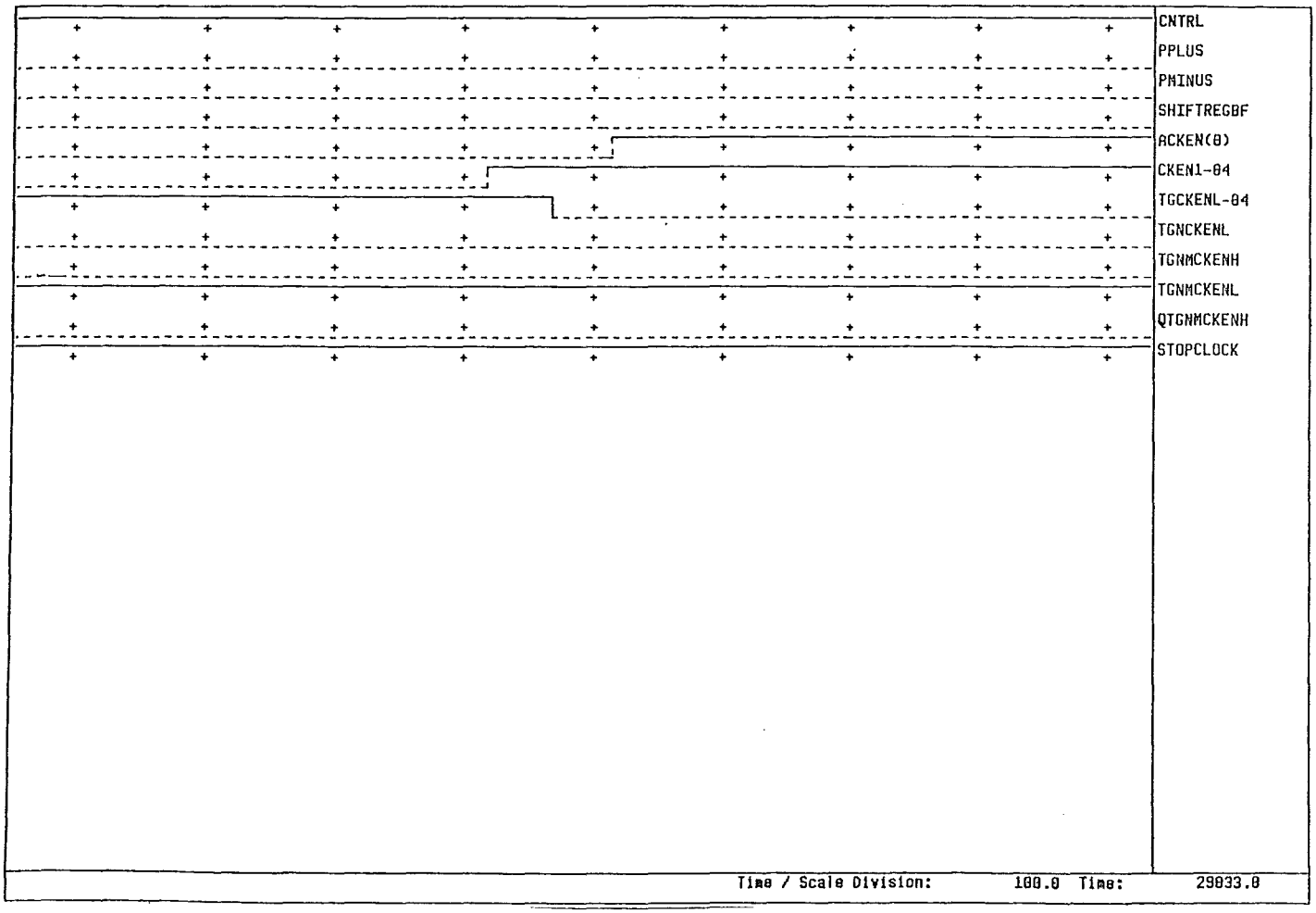


Figure A8 Clock Simulation (Toggle Starting Clocks)

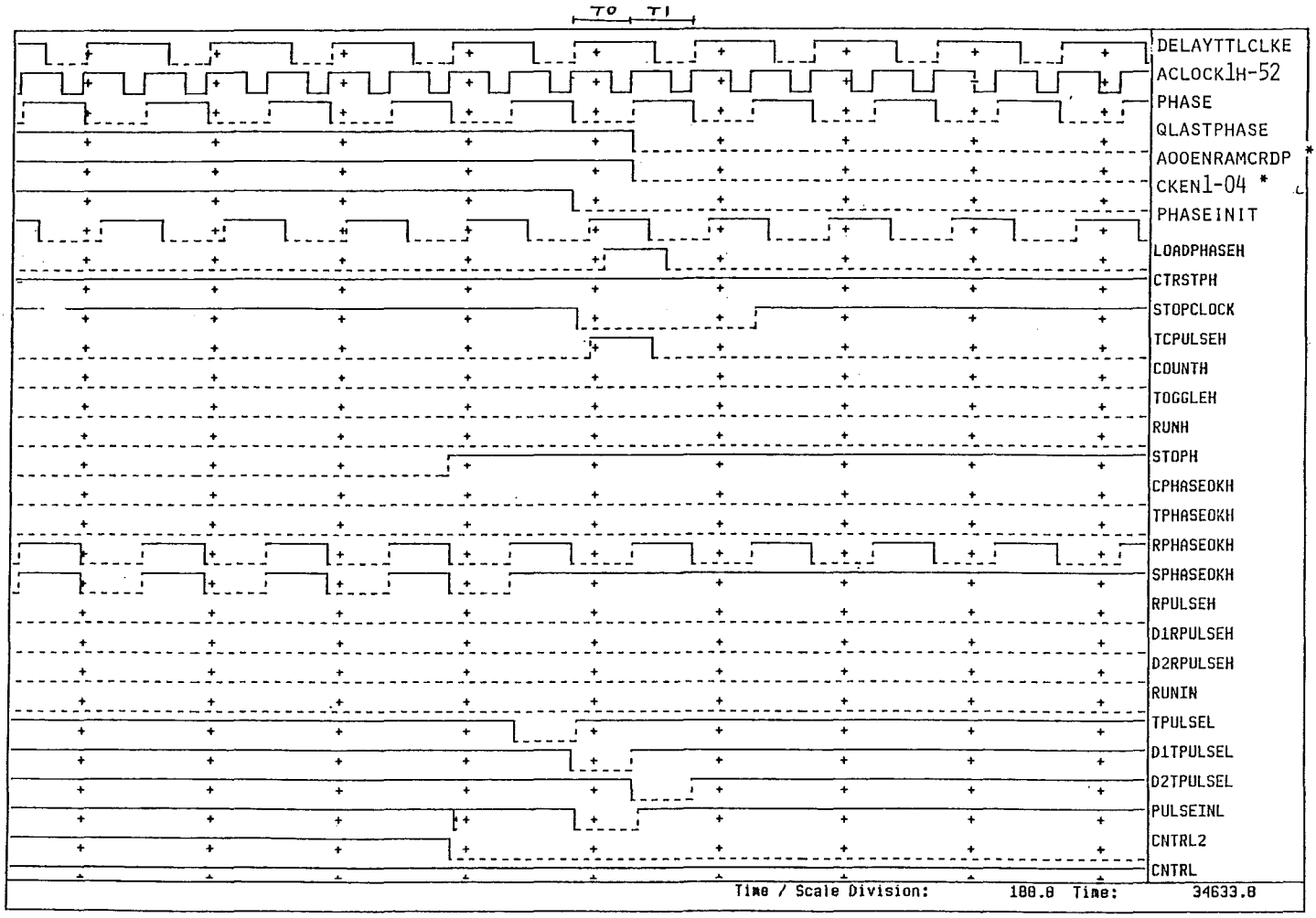


Figure A9 Clock Simulation (Stop Clocks)

*These two signals, AOOENRAMCRDP and CKEN1-04, are representatives of ECL and TTL clock enable respectively.

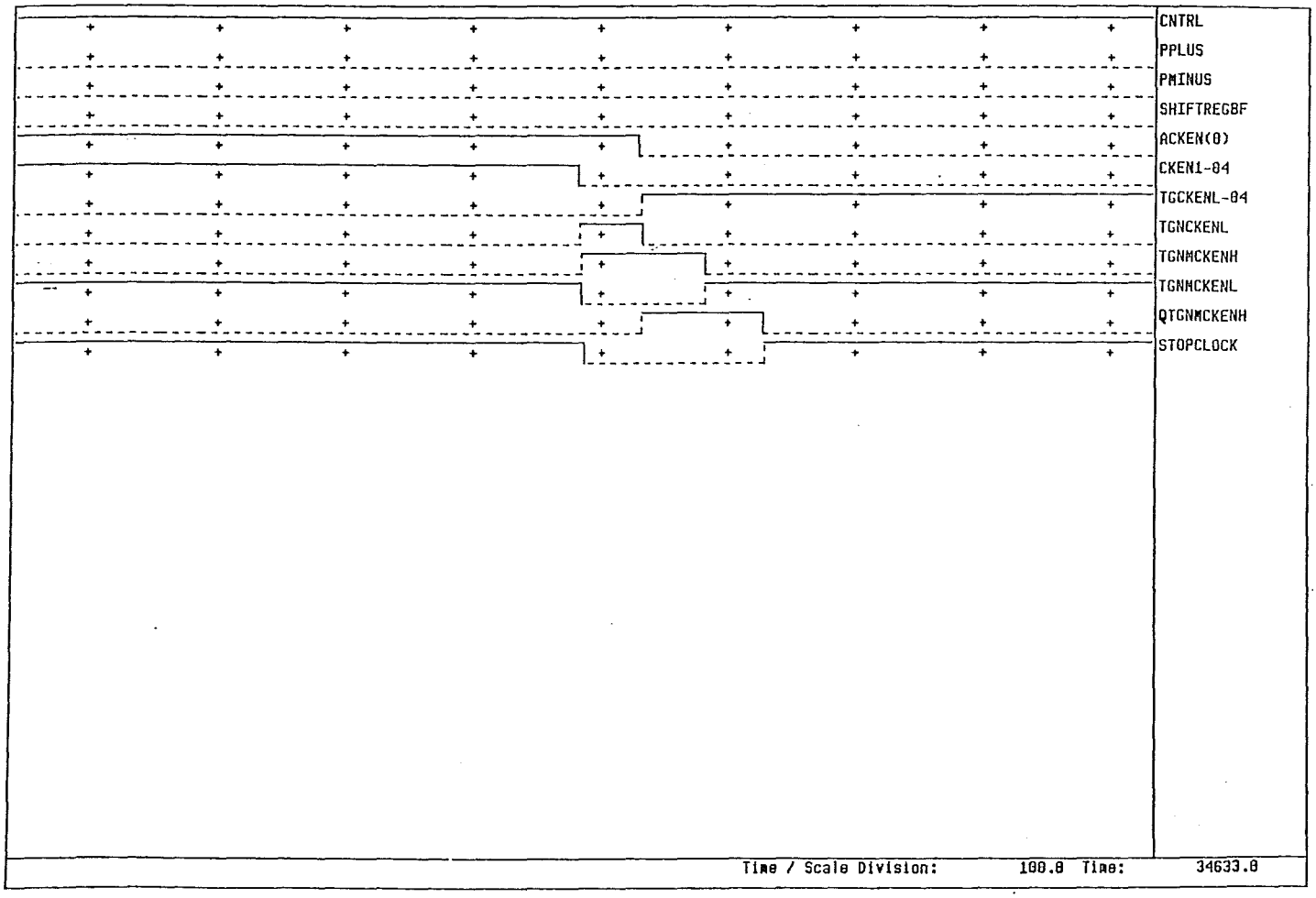


Figure A9 Clock Simulation (Stop Clocks)

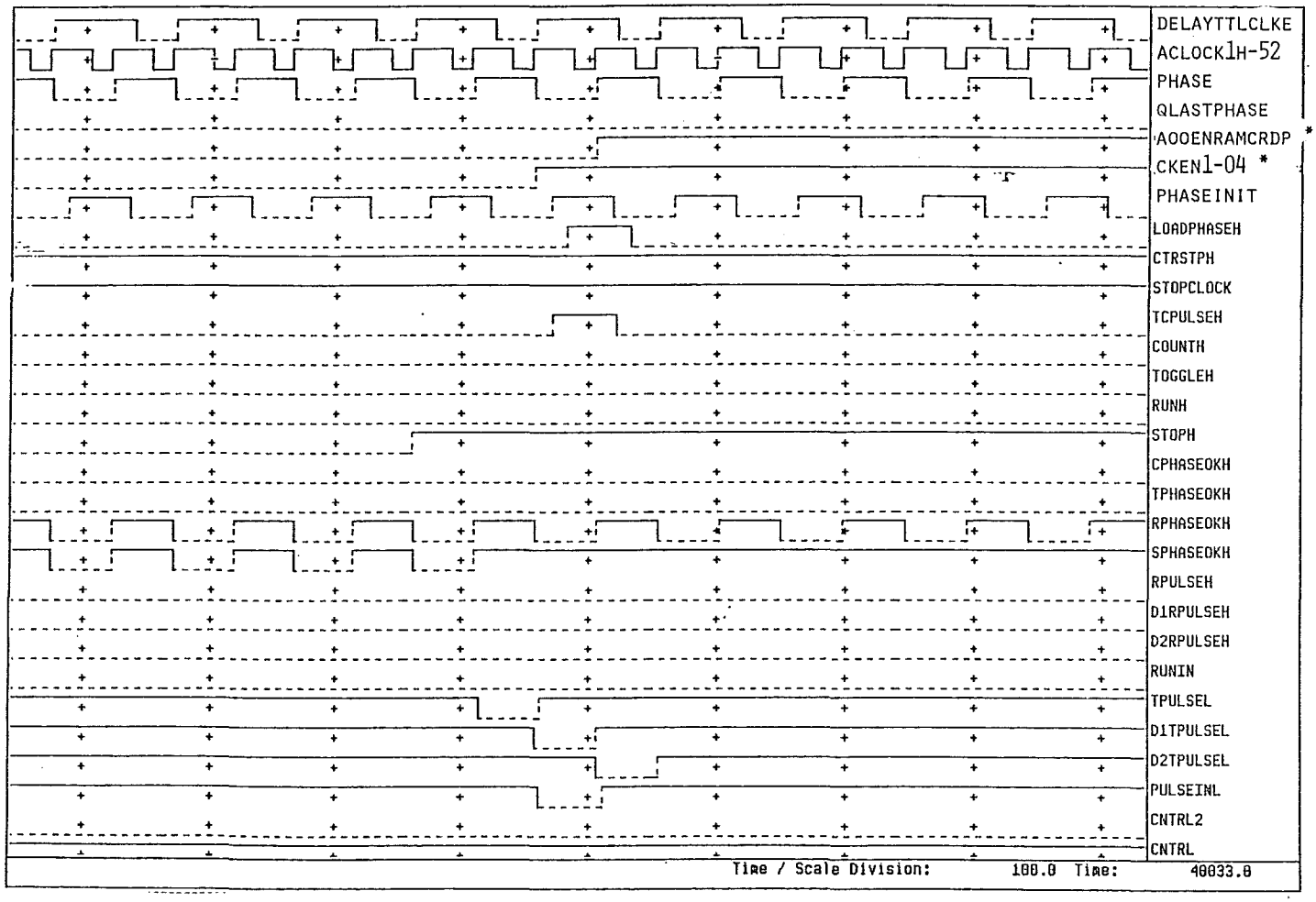


Figure A10 Clock Simulation (Path Check)

*These two signals, AOOENRAMCRDP and CKEN1-04, are representatives of ECL and TTL clock enable respectively.

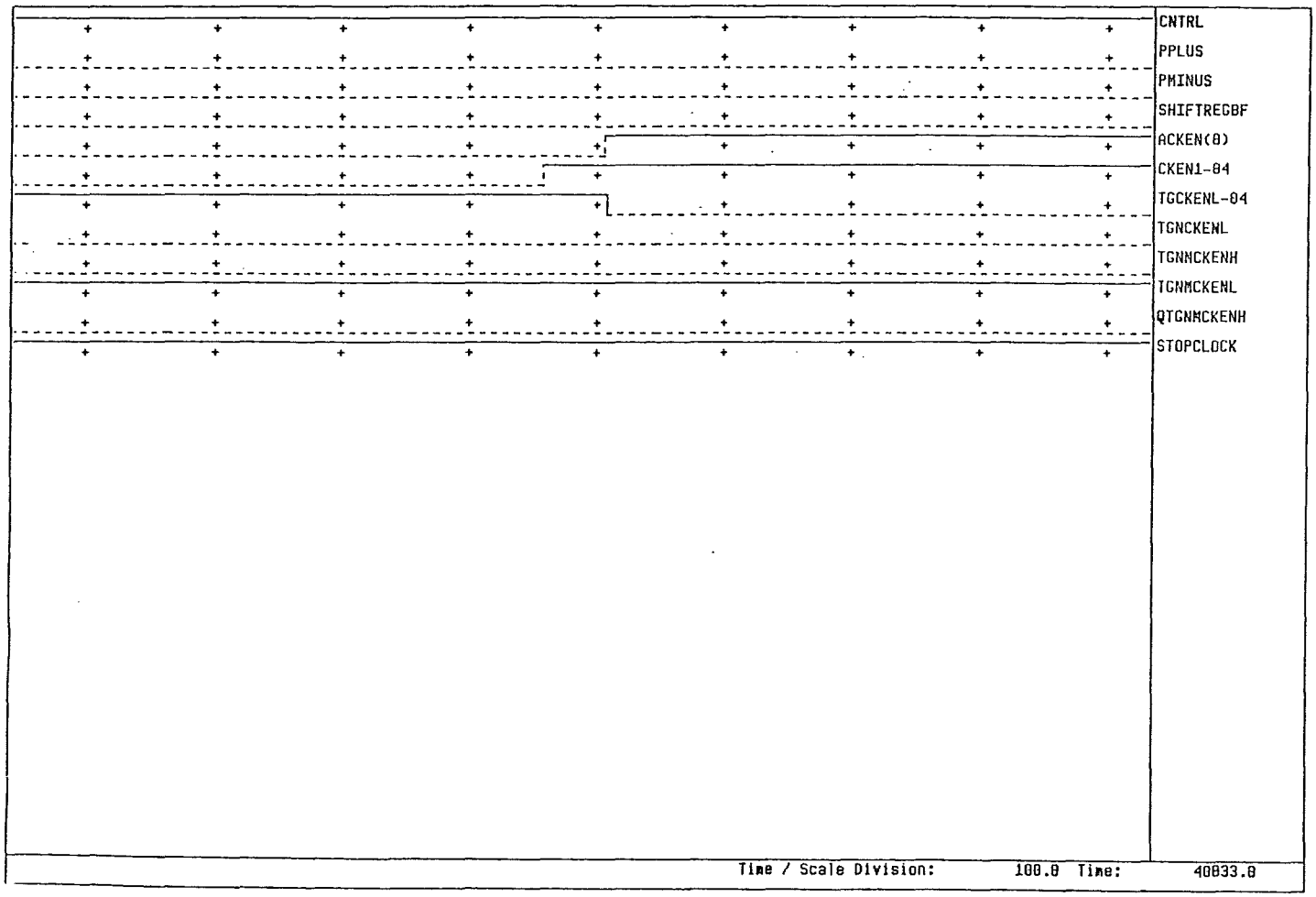
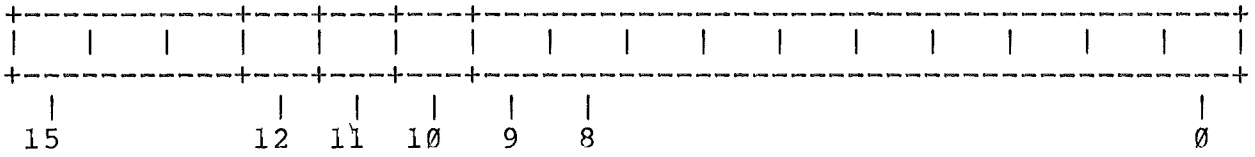


Figure A10 Clock Simulation (Path Check)

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING

The BP a 16-bit register is used as an error reporting field, and is defined in the followinw way:



	FIELD	DESCRIPTION
BITS	15-12	MODULE ADDRESS
BIT	11	ECL NON-FATAL ERROR DETECTED
BIT	10	TTL NON-FATAL ERROR DETECTED
BIT	9	TTL RAM BEING USED
BITS	8-0	ERROR CODE

MODULE NAME	MODULE ADDRESS
80186 PROCESSOR	1
MEMORY INIT.	2
TIMER	3
TEST_BUS	4
DMA	5
MPCC	6
SHIFT	7
IO_TEST	8
NMI_ROUTINE	-

B P R E G I S T E R

The errors are classified as FATAL or NON-FATAL. Fatal errors are defined as those which will prevent the microprocessor from communicating with the MPCCs. Similarly, non-fatal errors are those which have no effect on micro-MPCC communication and can be logged in the ECL ram.

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

All non-fatal errors have distinct error codes. The reason for having this distinction is because testing continues after a non-fatal error is encountered. This means that non-fatal errors may be carried from one module to another. On the other hand, whenever a fatal error is encountered it is reported to the LEDs, without destroying bits 11-9 of BP, and testing stops.

Only the first non-fatal error is reported to the LEDs. Subsequent non-fatal errors are simply ignored and the test contiues.

Non-fatal error-codes start from 1FF (hex) down to 1A1 (hex).

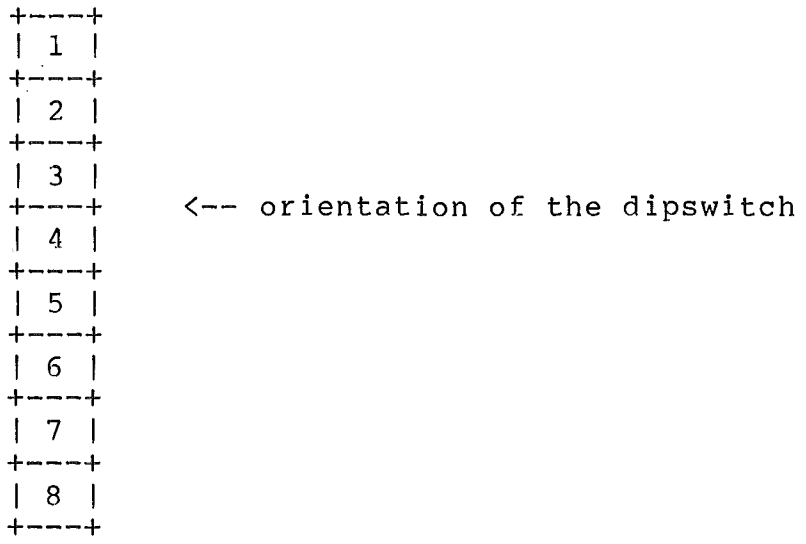
Fatal error-codes are not distinct from one module to another because testing stops as soon as the error is reported to the LEDs. Fatal error within the same module are distinct. Fatal-errors among different modules are distinguished by their module address.

EX., BP=2001 means that a fatal error occured in the memory module BP=1001 means that a fatal error occured in the processor module even though we have the same error code, 001, it has different meaning in each module.

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

D I P S W I T C H S E T T I N G S :

The dipswitch is located, vertically, on the front-end of the SMC card.



The purpose of the dipswitch settings is to provide the debugger with the necessary tools for examining a particular problem in an isolated fashion. However, the debugger cannot use the dipswitch until the NMI return address can be initialized. The break point after which the dipswitch can be used is indicated by turning ALL leds on and off; some dipswitch settings,marked by an asterisk, must adhere to this criterion to ensure proper results.

Dipswitch Settings:	BIT8 - BIT1	
INIT_MODE	EQU	ØØH ;INITIALIZATION
* AX_REG	EQU	Ø1H ;DISPLAY AX REGISTER
* BX_REG	EQU	Ø2H ;DISPLAY BX REGISTER
* CX_REG	EQU	Ø3H ;DISPLAY CX REGISTER
* DX_REG	EQU	Ø4H ;DISPLAY DX REGISTER
* SI_REG	EQU	Ø5H ;DISPLAY SI REGISTER
* DI_REG	EQU	Ø6H ;DISPLAY DI REGISTER
* BP_REG	EQU	Ø7H ;DISPLAY BP REGISTER
* SP_REG	EQU	Ø8H ;DISPLAY SP REGISTER
* DS_REG	EQU	Ø9H ;DISPLAY DS REGISTER
* ES_REG	EQU	ØAH ;DISPLAY ES REGISTER
* SS_REG	EQU	ØBH ;DISPLAY SS REGISTER
* CS_REG	EQU	ØCH ;DISPLAY CS REGISTER

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

```
Dipswitch Settings:          BIT8 - BIT1
-------------------          -----------
  OFF_LINE                     EQU  ØDH ;OFFLINE MODE
  LOOP_PROC                    EQU  ØEH ;LOOP ON PROCESSOR TEST
  LOOP_MEM                     EQU  ØFH ;LOOP ON MEMORY TEST
  LOOP_TIMER                   EQU  1ØH ;LOOP ON TIMER TEST
  LOOP_TEST_BUS                EQU  11H ;LOOP ON TEST-BUS TEST
  LOOP_IO_TEST                 EQU  12H ;LOOP ON IO_TEST
  LOOP_DMA                     EQU  13H ;LOOP ON DMA TEST
  LOOP_MPCC                    EQU  14H ;LOOP ON MPCC TEST
  LOOP_ECL_ARRAY               EQU  15H ;LOOP ON ECL ARRAY TEST
  LOOP_LED_TEST_S              EQU  16H ;START LED/DIPSWITCH TEST
  LOOP_LED_TEST_E              EQU  17H ;END LED/DIPSWITCH TEST
  LED_CK                       EQU  18H ;LED CHECK
  COUNTER_REG_DIPCODE          EQU  19H ;WRITE/READ A's INTO COUNTER
                                        ;REGISTER IN SHIFT ARRAY.
  COUNTER_CTRL_REG_DIPCODE     EQU  1AH ;WRITE/READ 5's INTO COUNTER-
                                        ;CONTROL REG. IN SHIFT ARRAY
  CONTROL_REG_DIPCODE          EQU  1BH ;WRITE/READ F's INTO CONTROL
                                        ;REGISTER IN SHIFT ARRAY.
  MASK_TOGGLE_REG1_DIPCODE     EQU  1CH ;WRITE/READ A's INTO MASK/TOGGLE
                                        ;REGISTER 1 IN SHIFT ARRAY
  MASK_TOGGLE_REG2_DIPCODE     EQU  1DH ;WRITE/READ 5's INTO MASK/TOGGLE
                                        ;REGISTER 2 IN SHIFT ARRAY
  ER_STP_REG_DIPCODE           EQU  1EH ;WRITE/READ F's INTO ERROR
                                        ;STOP REGISTER IN SHIFT ARRAY
  CLOCK_EN1_REG_DIPCODE        EQU  1FH ;READ ONLY,
                                        ;ENABLE 1 REGISTER IN SHFT ARRAY
                                        ;FOR ADDRESS LINES CHECKING ONLY
  CLOCK_EN2_REG_DIPCODE        EQU  2ØH ;READ ONLY,
                                        ;ENABLE 2 REGISTER IN SHFT ARRAY
                                        ;FOR ADDRESS LINES CHECKING ONLY
  RDSRL1_DIPCODE               EQU  21H ;READ ONLY, MPCC1 LOCATION J8F1
                                        ;FOR ADDRESS LINES CHECKING ONLY
  RDSRH1_DIPCODE               EQU  22H ;READ ONLY, MPCC1 LOCATION J8F1
                                        ;FOR ADDRESS LINES CHECKING ONLY
  TDSRL1_DIPCODE               EQU  23H ;WRITE/READ TDSRL1 REGISTER
                                        ;       MPCC1 , LOCATION  J8F1
  TDSRH1_DIPCODE               EQU  24H ;READ ONLY, MPCC1 LOCATION J8F1
                                        ;FOR ADDRESS LINES CHECKING ONLY
  PCSARL1_DIPCODE              EQU  25H ;WRITE/READ PCSARL1 REGISTER
                                        ;       MPCC1 , LOCATION  J8F1
  PCSARH1_DIPCODE              EQU  26H ;WRITE/READ PCSADH1 REGISTER
                                        ;       MPCC1 , LOCATION  J8F1
```

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

Dipswitch Settings: BIT8 - BIT1

```
PCRH1_DIPCODE           EQU  27H ;READ ONLY, MPCC1 LOCATION J8F1
                                 ;FOR ADDRESS LINES CHECKING ONLY
RDSRL2_DIPCODE          EQU  28H ;READ ONLY, MPCC2 LOCATION J0F1
                                 ;FOR ADDRESS LINES CHECKING ONLY
RDSRH2_DIPCODE          EQU  29H ;READ ONLY, MPCC2 LOCATION J0F1
                                 ;FOR ADDRESS LINES CHECKING ONLY
TDSRL2_DIPCODE          EQU  2AH ;WRITE/READ TDSRL2 REGISTER
                                 ;       MPCC2 , LOCATION  J0F1
TDSRH2_DIPCODE          EQU  2BH ;READ ONLY, MPCC2 LOCATION J0F1
                                 ;FOR ADDRESS LINES CHECKING ONLY
PCSARL2_DIPCODE         EQU  2CH ;WRITE/READ PCSADL2 REGISTER
                                 ;       MPCC2 , LOCATION  J0F1
PCSARH2_DIPCODE         EQU  2DH ;WRITE/READ PCSARH2 REGISTER
                                 ;       MPCC2 , LOCATION  J0F1
PCRH2_DIPCODE           EQU  2EH ;READ ONLY, MPCC2 LOCATION J0F1
                                 ;FOR ADDRESS LINES CHECKING ONLY
LCS_ADR0_DIPCODE        EQU  2FH ;WRITE/READ A's , ADDRESS ZERO
MCS1_ADR0_DIPCODE       EQU  30H ;WRITE/READ A's , ADDRESS ZERO
MCS2_ADR0_DIPCODE       EQU  31H ;WRITE/READ A's , ADDRESS ZERO
EXT_REG_DIPCODE         EQU  32H ;CHECK ECL EXTERNAL REGISTER
REG_DUMP                EQU  33H ;DUMP 80186 REGISTERS IN MEMORY
TSTBUS_DRIVER           EQU  34H ;CHECK TESTBUS DRIVERS

LCS_ADR0_0LOOP          EQU  35H ;WRITE/READ 0's , ADDRESS ZERO
MCS1_ADR0_0LOOP         EQU  36H ;WRITE/READ 0's , ADDRESS ZERO
MCS2_ADR0_0LOOP         EQU  37H ;WRITE/READ 0's , ADDRESS ZERO

LCS_ADR0_5LOOP          EQU  38H ;WRITE/READ 5's , ADDRESS ZERO
MCS1_ADR0_5LOOP         EQU  39H ;WRITE/READ 5's , ADDRESS ZERO
MCS2_ADR0_5LOOP         EQU  3AH ;WRITE/READ 5's , ADDRESS ZERO

LCS_ADR0_FLOOP          EQU  3BH ;WRITE/READ F's , ADDRESS ZERO
MCS1_ADR0_FLOOP         EQU  3CH ;WRITE/READ F's , ADDRESS ZERO
MCS2_ADR0_FLOOP         EQU  3DH ;WRITE/READ F's , ADDRESS ZERO

LCS_ADR2_0LOOP          EQU  3EH ;WRITE/READ 0's , ADDRESS TWO
MCS1_ADR2_0LOOP         EQU  3FH ;WRITE/READ 0's , ADDRESS TWO
MCS2_ADR2_0LOOP         EQU  40H ;WRITE/READ 0's , ADDRESS TWO

LCS_ADR2_5LOOP          EQU  41H ;WRITE/READ 5's , ADDRESS TWO
MCS1_ADR2_5LOOP         EQU  42H ;WRITE/READ 5's , ADDRESS TWO
MCS2_ADR2_5LOOP         EQU  43H ;WRITE/READ 5's , ADDRESS TWO
```

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

Dipswitch Settings: BIT8 - BIT1

```
LCS_ADR2_ALOOP          EQU  44H ;WRITE/READ A's , ADDRESS TWO
MCS1_ADR2_ALOOP         EQU  45H ;WRITE/READ A's , ADDRESS TWO
MCS2_ADR2_ALOOP         EQU  46H ;WRITE/READ A's , ADDRESS TWO

LCS_ADR2_FLOOP          EQU  47H ;WRITE/READ F's , ADDRESS TWO
MCS1_ADR2_FLOOP         EQU  48H ;WRITE/READ F's , ADDRESS TWO
MCS2_ADR2_FLOOP         EQU  49H ;WRITE/READ F's , ADDRESS TWO

EXT_REG_FS              EQU  50H ;WR/RD F's , EXTERNAL REGISTER
EXT_REG_0S              EQU  51H ;WR/RD 0'0 , EXTERNAL REGISTER
LOOP_CMD                EQU  52H ;WRITE IOBUFFCMD REGISTER
LOOP_STAT               EQU  53H ;READ IOBUFFSTAT REGISTER
LOOPSETNMI              EQU  54H ;LOOP ON NMI SET

LOOP_TY                 EQU  5CH ;RUN A's PATTERN ON TY CHAIN
LOOP_TTL_CLKEN          EQU  5DH ;SET TTL CLOCK ENABLE IN CONTROL
                                 ;   REGISTER INSIDE SHIFT ARRAY.
```

P R O G R A M M A B L E F U N C T I O N S:

```
LOOP_LCS1_PGM           EQU  57H ;WRITE/READ LCS1 REGION
LOOP_LCS2_PGM           EQU  58H ;WRITE/READ LCS2 REGION
LOOP_MCS1_PGM           EQU  59H ;WRITE/READ MCS  REGION
LOOP_ECL_PGM            EQU  5AH ;WRITE/READ ECL  REGION
LOOP_ECL_PGM            EQU  5AH ;WRITE/READ ECL  REGION
LOOP_UCS_PGM            EQU  5BH ;READ UCS  REGION
```

MEMORY REGION	BASE		RANGE	
LCS1	0	to	0FFFFH	
LCS2	1000H	to	1FFFFH	
MCS	2000H	to	2FFFH	; MCS = MCS1
ECL	3000H	to	31FFH	
UCS	0F000H	to	0FFFFFH	

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

Dipswitch Bit Definition:

Bit-8 = 1: RESET 80186 and start executing diagnostics from the beginning.
Bit-8 = 0: Serve NMI by executing DIPNEW module of the diagnostics.

HOW TO USE PROGRAMMABLE LOOPS:

step #1 : Turn bit-8 of the dipswitch.
step #2 : Select the desired programmable function by setting dipswitch bit to the function code; example

```
LOOP_ECL_PGM   EQU   5AH

dipswitch   bit-4 thru bit-1 = A
            bit-7 thru bit-5 = 5
```

step #3: Press PUSH BUTTON.

```
%*****************************************************************
%                      W A R N I N G                              *
%*****************************************************************
%*****     DO NOT TOUCH PUSH BUTTON THROUGHOUT THE REMAINDER      *
%*****     OF THE TEST.  IN CASE OF ANY DOUBT REPEAT THE PROCESS  *
%*****     PROCESS STARTING FROM STEP #2.                         *
%*****************************************************************
```

step #4: Set bit-4 thru bit-1 of the dipswitch to indicate the most significant digit of the offset. Reverse bit-7 to latch information. The data will be echoed on the LEDs.

% *** NOTE: BIT-7 ACTS AS A LATCHING SWITCH AND MUST BE REVERSED IN
% *** ONE DIRECTION ONLY.

step #5: Repeat step #4 for the remaining digits of the offset field.
step #6: Repeat step #4 for the remaining digits of the data field.

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

P R O C E S S O R

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 1001	PROCESSOR	PARITY FLAG ERROR	FATAL
BP = 1002	PROCESSOR	CARRY FLAG ERROR	FATAL
BP = 1003	PROCESSOR	CARRY FLAG ERROR	FATAL
BP = 1004	PROCESSOR	CARRY FLAG ERROR	FATAL
BP = 1005	PROCESSOR	CARRY FLAG ERROR	FATAL
BP = 1006	PROCESSOR	CARRY FLAG ERROR	FATAL
BP = 1007	PROCESSOR	CARRY FLAG ERROR	FATAL
BP = 1008	PROCESSOR	CARRY FLAG ERROR	FATAL
BP = 1009	PROCESSOR	CARRY FLAG ERROR	FATAL
BP = 100A	PROCESSOR	CARRY FLAG ERROR	FATAL
BP = 100B	PROCESSOR	CARRY FLAG ERROR	FATAL
BP = 100C	PROCESSOR	CARRY FLAG ERROR	FATAL
BP = 100D	PROCESSOR	CARRY FLAG ERROR	FATAL
BP = 100E	PROCESSOR	CARRY FLAG ERROR	FATAL
BP = 100F	PROCESSOR	CARRY FLAG ERROR	FATAL
BP = 1010	PROCESSOR	CARRY FLAG ERROR	FATAL
BP = 1011	PROCESSOR	CARRY FLAG ERROR	FATAL
BP = 1012	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 1013	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 1014	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 1015	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 1016	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 1017	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 1018	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 1019	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 101A	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 101B	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 101C	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 101D	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 101E	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 101F	PROCESSOR	ZERO FLAG ERROR	FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

P R O C E S S O R
(continued)

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 1020	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 1021	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 1022	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 1023	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 1024	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 1025	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 1026	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 1027	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 1028	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 1029	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 102A	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 102B	PROCESSOR	ZERO FLAG ERROR	FATAL
BP = 102C	PROCESSOR	SIGN FLAG ERROR	FATAL
BP = 102D	PROCESSOR	SIGN FLAG ERROR	FATAL
BP = 102E	PROCESSOR	SIGN FLAG ERROR	FATAL
BP = 102F	PROCESSOR	OVERFLOW FLAG ERROR	FATAL
BP = 1030	PROCESSOR	OVERFLOW FLAG ERROR	FATAL
BP = 1031	PROCESSOR	OVERFLOW FLAG ERROR	FATAL
BP = 1032	PROCESSOR	OVERFLOW FLAG ERROR	FATAL
BP = 1033	PROCESSOR	PARITY FLAG ERROR	FATAL
BP = 1034	PROCESSOR	PARITY FLAG ERROR	FATAL
BP = 1035	PROCESSOR	PARITY FLAG ERROR	FATAL
BP = 1036	PROCESSOR	CARRY FLAG ERROR	FATAL
BP = 1037	PROCESSOR	ZERO FLAG ERROR	FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

M E M O R Y

Region	Board Location
PROM	D0 : B9D2
PROM	D1 : C7D2
PROM	D2 : E3D2
PROM	D3 : B1D2

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 21FF	MEMORY	NMI-ENABLE STUCK LOW	NON-FATAL
BP = 2002	MEMORY	TTL PROM, D0 IS FAULTY	FATAL
BP = 2003	MEMORY	TTL PROM, D1 IS FAULTY	FATAL
BP = 2004	MEMORY	TTL PROM, D2 IS FAULTY	FATAL
BP = 2005	MEMORY	TTL PROM, D3 IS FAULTY	FATAL
BP = 2006	MEMORY	TTL PROM, D1 D0 ARE FAULTY	FATAL
BP = 2007	MEMORY	TTL PROM, D2 D0 ARE FAULTY	FATAL
BP = 2008	MEMORY	TTL PROM, D3 D0 ARE FAULTY	FATAL
BP = 2009	MEMORY	TTL PROM, D2 D1 D0 FAULTY	FATAL
BP = 200A	MEMORY	TTL PROM, D3 D1 D0 FAULTY	FATAL
BP = 200B	MEMORY	TTL PROM, D3 D2 D0 FAULTY	FATAL
BP = 200C	MEMORY	TTL PROM, D1 D2 ARE FAULTY	FATAL
BP = 200D	MEMORY	TTL PROM, D3 D1 ARE FAULTY	FATAL
BP = 200E	MEMORY	TTL PROM, D3 D2 D1 FAULTY	FATAL
BP = 200F	MEMORY	TTL PROM, D3 D2 ARE FAULTY	FATAL
BP = 2010	MEMORY	TTL PROM, D3 D2 D1 D0 BAD	FATAL
BP = 21FE	MEMORY	NMI ENABLE STUCK HIGH	NON-FATAL
BP = 21FD	MEMORY	UNABLE TO ACQUIRE I/O BUFFER	NON-FATAL
BP = 21FC	MEMORY	UNABLE TO ACQUIRE I/O BUFFER	NON-FATAL
BP = 21FB	MEMORY	UNABLE TO ACQUIRE I/O BUFFER	NON-FATAL
BP = 21FA	MEMORY	NMI ENABLE STUCK LOW	NON-FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

M E M O R Y
(continued)

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 21F9	MEMORY	UNABLE TO ACQUIRE I/O BUFFER	NON-FATAL
BP = 21F8	MEMORY	UNABLE TO ACQUIRE I/O BUFFER	NON-FATAL
BP = 21F7	MEMORY	UNABLE TO ACQUIRE I/O BUFFER	NON-FATAL
BP = 21F6	MEMORY	UNABLE TO ACQUIRE I/O BUFFER	NON-FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

M E M O R Y
(continued)

```
%          ***********************************************
%          *  TTL LCS REGION IN DIVIDED INTO TWO EQUAL   *
%          *  PARTS (LCSL & LCSH) EACH IS 64K BYTES.     *
%          *                                             *
%          *  ALL ERRORS WHICH CORRESPOND TO TTL RAM     *
%          *  HAVE BIT #9 OF BP SET.  THIS BIT INDICATES *
%          *  THAT TTL RAM IS BEING USED.                *
%          *                                             *
%          *  C H E C K   C O N T R O L   L I N E S      *
%          *                                             *
%          ***********************************************
```

Region	Board Location
LCS	D0 : B9B7
LCS	D1 : C7B7
LCS	D2 : E3B7
LCS	D3 : B1B7

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 2212	MEMORY Ø's	LCSL-REGION, LOWER BYTE OF LEAST SIG. WORD	FATAL
BP = 2213	MEMORY	LCSL, UPPER BYTE OF LEAST SIGNIFICANT WORD D1 BAD	FATAL
BP = 2214	MEMORY	LCSL, LOWER BYTE OF MOST SIGNIFICANT WORD D2 BAD	FATAL
BP = 2215	MEMORY	LCSL, UPPER BYTE OF MOST SIGNIFICANT WORD D3 BAD	FATAL
BP = 2216	MEMORY	LCSL, BOTH CHIPS HOLDING LEAST SIG. WORD D0 D1 BAD	FATAL
BP = 2217	MEMORY	LCSL, BOTH CHIPS HOLDING MOST SIG. WORD D2 D3 BAD	FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

M E M O R Y
(continued)

Region	Board Location	Region	Board Location
MCS	D0 : C8A7	MCS	D4 : A8A7
MCS	D1 : D3A7	MCS	D5 : B3A7
MCS	D2 : D8A7	MCS	D6 : B8A7
MCS	D3 : E3A7	MCS	D7 : C3A7

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 23B0	MEMORY 0's	MCS REGION, D0 IS FAULTY	NON-FATAL
BP = 23AF	MEMORY	MCS REGION, D1 IS FAULTY	NON-FATAL
BP = 23AE	MEMORY	MCS REGION, D2 IS FAULTY	NON-FATAL
BP = 23AD	MEMORY	MCS REGION, D3 IS FAULTY	NON-FATAL
BP = 23AC	MEMORY	MCS REGION, D1 D0 ARE BAD	NON-FATAL
BP = 23AB	MEMORY	MCS REGION, D2 D0 ARE BAD	NON-FATAL
BP = 23AA	MEMORY	MCS REGION, D3 D0 ARE BAD	NON-FATAL
BP = 23A9	MEMORY	MCS REGION, D2 D1 ARE BAD	NON-FATAL
BP = 23A8	MEMORY	MCS REGION, D3 D1 ARE BAD	NON-FATAL
BP = 23A7	MEMORY	MCS REGION, D3 D2 ARE BAD	NON-FATAL
BP = 23A6	MEMORY	MCS REGION, D2 D1 D0 BAD	NON-FATAL
BP = 23A5	MEMORY	MCS REGION, D3 D1 D0 BAD	NON-FATAL
BP = 23A4	MEMORY	MCS REGION, D3 D2 D0 BAD	NON-FATAL
BP = 23A3	MEMORY	MCS REGION, D3 D2 D1 BAD	NON-FATAL
BP = 23A2	MEMORY	MCS REGION, D3 D2 D1 D0	NON-FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

M E M O R Y
(continued)

```
% ************************************************************************
%     MEMORY REGION MCS2 IS THE ECL RAM. ECL RAM ERROR ARE NON-FATAL
%     AND BIT #9 OF BP-REG IS SET.
%     - NON-FATAL ERRORS ARE DISTINCT FROM ONE ANOTHER THROUGHOUT THE
%       DIAGNOSTICS TEST.
%     - NON-FATAL ERROR-CODES START FROM 1FFH DOWNWARD.
%
%       CONSEQUENTLY, ALL ECL RAM NON-FATAL ERROR-CODES START WITH
%       29xxH, WHERE THE ACTUAL CODE IS 1xxH.
% ************************************************************************
```

Region	Board Location
ECL	D0 : L3F8
ECL	D1 : M1F8
ECL	D2 : M9F8
ECL	D3 : N7F8

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 205B	MEMORY	COULDN'T ENABLE SMC TTL-CLK	FATAL
BP = 205D	MEMORY 0's	ECL RAM, D0 IS FAULTY	FATAL
BP = 205E	MEMORY	ECL RAM, D1 IS FAULTY	FATAL
BP = 205F	MEMORY	ECL RAM, D2 IS FAULTY	FATAL
BP = 205C	MEMORY	ECL RAM, D3 IS FAULTY	FATAL
BP = 2060	MEMORY	ECL RAM, D1 D0 IS FAULTY	FATAL
BP = 2061	MEMORY	ECL RAM, D2 D0 IS FAULTY	FATAL
BP = 2062	MEMORY	ECL RAM, D3 D0 IS FAULTY	FATAL
BP = 2063	MEMORY	ECL RAM, D2 D1 IS FAULTY	FATAL
BP = 2064	MEMORY	ECL RAM, D3 D1 IS FAULTY	FATAL
BP = 2065	MEMORY	ECL RAM, D3 D2 IS FAULTY	FATAL
BP = 2066	MEMORY	ECL RAM, D2 D1 D0 FAULTY	FATAL
BP = 2067	MEMORY	ECL RAM, D3 D1 D0 FAULTY	FATAL
BP = 2068	MEMORY	ECL RAM, D3 D2 D0 FAULTY	FATAL
BP = 2069	MEMORY	ECL RAM, D3 D2 D1 FAULTY	FATAL
BP = 206A	MEMORY	ECL RAM, D3 D2 D1 D0	FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

M E M O R Y
(continued)

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 2227	MEMORY	LCS-MCS CHIP SELECT	FATAL
BP = 2228	MEMORY	LCS-MCS2 (ECL) CHIP SELECT	FATAL
BP = 2229	MEMORY	MCS -MCS2 (ECL) CHIP SELECT	FATAL
BP = 222A	MEMORY	SBE FLAG STUCK HIGH	FATAL
BP = 222B	MEMORY	MBE FLAG STUCK HIGH	FATAL
BP = 222C	MEMORY	SBE & MBE FLAGS STUCK HIGH	FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

M E M O R Y
(continued)

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 222E	MEMORY 5's	LCSL UPPER BYTE OF MOST SIG. WORD OR ITS CTRL, D3	FATAL
BP = 222F	MEMORY	LCSL UPPER BYTE OF LEAST SIG. WORD OR ITS CTRL, D1	FATAL
BP = 2230	MEMORY	LCSL LOWER BYTE OF MOST SIG. WORD OR ITS CTRL, D2	FATAL
BP = 2231	MEMORY	LCSL LOWER BYTE OF LEAST SIG. WORD OR ITS CTRL, D0	FATAL
BP = 2232	MEMORY	LCSL BOTH CHIPS HOLD'N LEAST SIG. WORD, D2 D3	FATAL
BP = 2233	MEMORY	LCSL BOTH CHIPS HOLD'N MOST SIG. WORD , D0 D1	FATAL
BP = 2234	MEMORY	LCSL ADDRESS LINES PROBLEM ON MOST SIG. WORD	FATAL
BP = 2235	MEMORY	LCSH ADDRESS LINES PROBLEM ON BOTH CHIPS	FATAL
BP = 23A1	MEMORY 5's	MCS ..... D3 IS FAULTY	NON-FATAL
BP = 23A0	MEMORY	MCS ..... D0 IS FAULTY	NON-FATAL
BP = 239F	MEMORY	MCS ..... D2 IS FAULTY	NON-FATAL
BP = 239E	MEMORY	MCS ..... D1 IS FAULTY	NON-FATAL
BP = 239D	MEMORY	MCS .... D0,D1 ARE FAULTY	NON-FATAL
BP = 239C	MEMORY	MCS .... D0,D2 ARE FAULTY	NON-FATAL
BP = 239B	MEMORY	MCS .... D0,D3 ARE FAULTY	NON-FATAL
BP = 239A	MEMORY	MCS .... D1,D2 ARE FAULTY	NON-FATAL
BP = 2399	MEMORY	MCS .... D1,D3 ARE FAULTY	NON-FATAL
BP = 2398	MEMORY	MCS .... D2,D3 ARE FAULTY	NON-FATAL
BP = 2397	MEMORY	MCS , D0,D1,D2 ARE FAULTY	NON-FATAL
BP = 2396	MEMORY	MCS , D0,D1,D3 ARE FAULTY	NON-FATAL
BP = 2395	MEMORY	MCS , D0,D2,D3 ARE FAULTY	NON-FATAL
BP = 2394	MEMORY	MCS , D1,D2,D3 ARE FAULTY	NON-FATAL
BP = 2393	MEMORY	MCS , D0,D1,D2,D3 FAULTY	NON-FATAL
BP = 2392	MEMORY	MCS , ADDRESS LINES	NON-FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

M E M O R Y
(continued)

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 206B	MEMORY ,5's	ECL RAM, D3 IS FAULTY	FATAL
BP = 206C	MEMORY	ECL RAM, D0 IS FAULTY	FATAL
BP = 206D	MEMORY	ECL RAM, D1 IS FAULTY	FATAL
BP = 206E	MEMORY	ECL RAM, D2 IS FAULTY	FATAL
BP = 206F	MEMORY	ECL RAM, D0,D1 ARE FAULTY	FATAL
BP = 2070	MEMORY	ECL RAM, D0,D2 ARE FAULTY	FATAL
BP = 2071	MEMORY	ECL RAM, D0,D3 ARE FAULTY	FATAL
BP = 2072	MEMORY	ECL RAM, D1,D2 ARE FAULTY	FATAL
BP = 2073	MEMORY	ECL RAM, D1,D3 ARE FAULTY	FATAL
BP = 2074	MEMORY	ECL RAM, D2,D3 ARE FAULTY	FATAL
BP = 2075	MEMORY	ECL RAM, D2 D1 D0 FAULTY	FATAL
BP = 2076	MEMORY	ECL RAM, D3 D1 D0 FAULTY	FATAL
BP = 2077	MEMORY	ECL RAM, D3 D2 D0 FAULTY	FATAL
BP = 2078	MEMORY	ECL RAM, D3 D2 D1 FAULTY	FATAL
BP = 2079	MEMORY	ECL RAM, D3 D2 D1 D0	FATAL
BP = 2089	MEMORY	ECL RAM, ADDRESS PROBLEM	FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

M E M O R Y
(continued)

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 2246	MEMORY Adr	LCSL UPPER BYTE OF MOST SIG. WORD OR ITS CTRL	FATAL
BP = 2247	MEMORY	LCSL UPPER BYTE OF LEAST SIG. WORD OR ITS CTRL	FATAL
BP = 2248	MEMORY	LCSL LOWER BYTE OF MOST SIG. WORD OR ITS CTRL	FATAL
BP = 2249	MEMORY	LCSL LOWER BYTE OF LEAST SIG. WORD OR ITS CTRL	FATAL
BP = 224A	MEMORY	LCSL BOTH CHIPS HOLD'N LEAST SIG. WORD	FATAL
BP = 224B	MEMORY	LCSL BOTH CHIPS HOLD'N MOST SIG. WORD	FATAL

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 2391	MEMORY, Adr	MCS ..... D3 FAULTY	NON-FATAL
BP = 2390	MEMORY	MCS ..... D0 FAULTY	NON-FATAL
BP = 238F	MEMORY	MCS ..... D1 FAULTY	NON-FATAL
BP = 238E	MEMORY	MCS ..... D2 FAULTY	NON-FATAL
BP = 238D	MEMORY	MCS ..... D0,D1 FAULTY	NON-FATAL
BP = 238C	MEMORY	MCS ..... D0,D2 FAULTY	NON-FATAL
BP = 238B	MEMORY	MCS ..... D0,D3 FAULTY	NON-FATAL
BP = 238A	MEMORY	MCS ..... D1,D2 FAULTY	NON-FATAL
BP = 2389	MEMORY	MCS ..... D1,D3 FAULTY	NON-FATAL
BP = 2388	MEMORY	MCS ..... D2,D3 FAULTY	NON-FATAL
BP = 2387	MEMORY	MCS ..... D0,D1,D2 FAULTY	NON-FATAL
BP = 2386	MEMORY	MCS ..... D0,D1,D3 FAULTY	NON-FATAL
BP = 2385	MEMORY	MCS ..... D0,D2,D3 FAULTY	NON-FATAL
BP = 2384	MEMORY	MCS ..... D1,D2,D3 FAULTY	NON-FATAL
BP = 2383	MEMORY	MCS .. D0,D1,D2,D3 FAULTY	NON-FATAL
BP = 225B	MEMORY	COULDN'T ENABLE SMC-TTL CLOCK ENABLE	FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

M E M O R Y
(continued)

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 207A	MEMORY, Adr	ECL RAM, D3 IS FAULTY	FATAL
BP = 207B	MEMORY	ECL RAM, D0 IS FAULTY	FATAL
BP = 207C	MEMORY	ECL RAM, D1 IS FAULTY	FATAL
BP = 207D	MEMORY	ECL RAM, D2 IS FAULTY	FATAL
BP = 207E	MEMORY	ECL RAM, D0,D1 ARE FAULTY	FATAL
BP = 207F	MEMORY	ECL RAM, D0,D2 ARE FAULTY	FATAL
BP = 2080	MEMORY	ECL RAM, D0,D3 ARE FAULTY	FATAL
BP = 2081	MEMORY	ECL RAM, D1,D2 ARE FAULTY	FATAL
BP = 2082	MEMORY	ECL RAM, D1,D3 ARE FAULTY	FATAL
BP = 2083	MEMORY	ECL RAM, D2,D3 ARE FAULTY	FATAL
BP = 2084	MEMORY	ECL RAM, D2 D1 D0 FAULTY	FATAL
BP = 2085	MEMORY	ECL RAM, D3 D1 D0 FAULTY	FATAL
BP = 2086	MEMORY	ECL RAM, D3 D2 D0 FAULTY	FATAL
BP = 2087	MEMORY	ECL RAM, D3 D2 D1 FAULTY	FATAL
BP = 2088	MEMORY	ECL RAM, D3 D2 D1 D0	FATAL

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 2089	MEMORY	ZERO FLAG ERROR, 80186	FATAL
BP = 208A	MEMORY	ZERO FLAG ERROR, 80186	FATAL
BP = 208B	MEMORY	ZERO FLAG ERROR, 80186	FATAL
BP = 208C	MEMORY	ZERO FLAG ERROR, 80186	FATAL
BP = 208D	MEMORY	ZERO FLAG ERROR, 80186	FATAL
BP = 208E	MEMORY	ZERO FLAG ERROR, 80186	FATAL
BP = 208F	MEMORY	ZERO FLAG ERROR, 80186	FATAL
BP = 2090	MEMORY	ZERO FLAG ERROR, 80186	FATAL
BP = 2091	MEMORY	ZERO FLAG ERROR, 80186	FATAL
BP = 2092	MEMORY	ZERO FLAG ERROR, 80186	FATAL
BP = 2093	MEMORY	ZERO FLAG ERROR, 80186	FATAL
BP = 2094	MEMORY	ZERO FLAG ERROR, 80186	FATAL
BP = 2095	MEMORY	ZERO FLAG ERROR, 80186	FATAL
BP = 2096	MEMORY	SIGN FLAG ERROR, 80186	FATAL
BP = 2097	MEMORY	SIGN FLAG ERROR, 80186	FATAL
BP = 2098	MEMORY	OVERFLOW FLAG ERROR, 80186	FATAL
BP = 2099	MEMORY	PARITY FLAG ERROR, 80186	FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

T I M E R

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 3001	TIMER	COULDN'T DISABLE TIMER0	FATAL
BP = 3002	TIMER	COULDN'T DISABLE TIMER1	FATAL
BP = 3003	TIMER	COULDN'T DISABLE TIMER2	FATAL
BP = 3004	TIMER	COULDN'T ENABLE TIMER0	FATAL
BP = 3005	TIMER	COULDN'T ENABLE TIMER1	FATAL
BP = 3006	TIMER	COULDN'T ENABLE TIMER2	FATAL
BP = 3007	TIMER		FATAL
BP = 3008	TIMER		FATAL

T E S T B U S

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 45F5	TSTBUS	FRAMING ERROR	NON-FATAL
BP = 45F4	TSTBUS	PARITY ERROR	NON-FATAL
BP = 45F3	TSTBUS	PARITY & FRAMING ERROR	NON-FATAL
BP = 45F2	TSTBUS	PARITY & OVERRUN ERROR	NON-FATAL
BP = 45F1	TSTBUS	PARITY,OVERRUN & FRAMING	NON-FATAL
BP = 45F0	TSTBUS	OVERRUN ERROR	NON-FATAL
BP = 45EF	TSTBUS	OVERRUN & FRAMING ERROR	NON-FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

D M A

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 55EE	DMA 0	TRANSFERED DATA IS BAD	NON-FATAL
BP = 55ED	DMA0	DID NOT RECEIVE END OF INTERRUPT	NON-FATAL
BP = 55EC	DMA 1	TRANSFERED DATA IS BAD	NON-FATAL
BP = 55EB	DMA 1	DID NOT RECEIVE E.O.M. INT	NON-FATAL
BP = 55EA	DMA 0	TRANSFERED DATA WAS BAD & DIDN'T RECEIVE E.O.M. INT.	NON-FATAL
BP = 55E9	DMA 1	TRANSFERED DATA WAS BAD	NON-FATAL
BP = 55E8	DMA 1	TRANSFERED DATA WAS BAD & DIDN'T RECEIVE E.O.M. INT.	NON-FATAL
BP = 5001	DMA 0,1	DATA TRANSFERED IS BAD	FATAL
BP = 5002	DMA 0,1	DIDN'T RECEIVE E.O.M. INT.	FATAL
BP = 5003	DMA 0,1	TRANSFERED DATA BAD, DMA0,1 DIDN'T RCV F.O.M. INT. DMA0	FATAL
BP = 5004	DMA 0,1	TRANSFERED DATA BAD, DMA0,1 DIDN'T RCV INT. DMA 0,1	FATAL
BP = 5005	DMA 0,1	TRANSFERED DATA BAD, DMA 1 DIDN'T RCV INT. DMA 0,1	FATAL
BP = 5006	DMA 0,1	DMA 0 DIDN'T XFER GOOD DATA DMA 1 DIDN'T RCV INT.	FATAL
BP = 5007	DMA 0,1	DMA 0 DIDN'T XFER GOOD DATA DMA 0 DIDN'T RCV INT. DMA 1 DIDN'T RCV INT.	FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

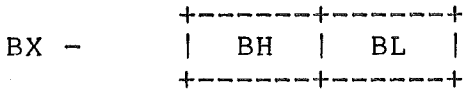
D M A
(Continued)

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 5008	DMA 0,1	DMA 0 DIDN'T RCV INT. DMA 1 DIDN'T RCV INT.	FATAL
BP = 55E7	DMA 1	DIDN'T RCV END OF DMA INT.	NON-FATAL
BP = 55E6	DMA 0	DIDN'T XFER DATA PROPERLY.	NON-FATAL
BP = 55E5	DMA 0	DIDN'T XFER DATA PROPERLY. & DIDN'T RCV INT.	NON-FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

M P C C

The data register - BX holds the error codes for the MPCCs.



BH holds error information for MPCC1
BL holds error information for MPCC0

The least significant digit in BH, BL may have the following codes,

0 : Good data
E : Did not receive data
F : Bad data

The most significant digit in BH, BL may have the following codes,

0 : Good data
F : Did not receive end of message interrupt, E.O.M int.

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 6001	MPCC 0,1	BOTH MPDCCs ARE FAILING	FATAL
BP = 6002	MPCC 0,1	BOTH MPDCCs ARE FAILING	FATAL
BP = 6003	MPCC 0,1	BOTH MPDCCs ARE FAILING	FATAL
BP = 6004	MPCC 0,1	BOTH MPDCCs ARE FAILING	FATAL
BP = 6005	MPCC 0,1	BOTH MPDCCs ARE FAILING	FATAL
BP = 63E4	MPCC 1	DIDN'T RCV E.O.M. INT.	NON-FATAL
BP = 63E3	MPCC 1	MPCC 0 IS OK, MPCC1 IS BAD	NON-FATAL
BP = 63E2	MPCC 0	DIDN'T RCV E.O.M. INT.	NON-FATAL
BP = 63E1	MPCC 1	DIDN'T RCV E.O.M. INT. or DATA RECEIVED IS BAD	NON-FATAL
BP = 63E0	MPCC 1	DIDN'T RCV E.O.M. INT.	NON-FATAL
BP = 63DF	MPCC 0	DATA RECEIVED IS BAD	NON-FATAL
BP = 63DE	MPCC 0	DIDN'T RECEIVE DATA or EOM	NON-FATAL
BP = 63DD	MPCC 0	DIDN'T RCV E.O.M. INT. & DIDN'T RECEIVE DATA	NON-FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

S H I F T

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 71DC	SHIFT	COUNTER DIDN'T COUNT TO Ø	NON-FATAL
BP = 71DB	SHIFT	COUNTER DIDN'T COUNT TO Ø HALT TESTING, OFFLINE MODE	NON-FATAL
BP = 71DA	SHIFT	TEST WORD DIDN'T PROPAGATE	NON-FATAL
BP = 71D9	SHIFT	TEST WORD DIDN'T PROPAGATE HALT TESTING, LOOP/OFFLINE	NON-FATAL
BP = 71D8	SHIFT	COUNTER DIDN'T COUNT TO Ø	NON-FATAL
BP = 71D7	SHIFT	COUNTER DIDN'T COUNT TO Ø HALT TESTING, LOOP/OFFLINE	NON-FATAL
BP = 71D6	SHIFT	MASK-TOGGLE 1 PROBLEM	NON-FATAL
BP = 71D5	SHIFT	MASK-TOGGLE 1 PROBLEM & HALT TESTING, LOOP/OFFLINE	NON-FATAL
BP = 71D4	SHIFT	MASK-TOGGLE 2 PROBLEM	NON-FATAL
BP = 71D3	SHIFT	MASK-TOGGLE 2 PROBLEM & HALT TESTING, LOOP/OFFLINE	NON-FATAL
BP = 71D2	SHIFT	CONTROL REGISTER PROBLEM	NON-FATAL
BP = 71D1	SHIFT	CONTROL REGISTER PROBLEM HALT TESTING, LOOP/OFFLINE	NON-FATAL
BP = 71DØ	SHIFT	COUNTER CONTROL REG PROBLEM	NON-FATAL
BP = 71CF	SHIFT	COUNTER CONTROL REG PROBLEM HALT TESTING, LOOP/OFFLINE	NON-FATAL
BP = 71CE	SHIFT	COUNTER CONTROL REG PROBLEM	NON-FATAL
BP = 71CD	SHIFT	COUNTER CONTROL REG PROBLEM HALT TESTING, LOOP/OFFLINE	NON-FATAL

I / O T E S T

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = 81CC	I/O	COULDN'T OWN BUFFER	NON-FATAL
BP = 81CB	I/O	COULDN'T RELEASE BUFFER	NON-FATAL
BP = 81CA	I/O	COUNTER IS NOT ZERO	NON-FATAL
BP = 81C9	I/O	SIGNATURE WORD DIDN'T CHK	NON-FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

N M I R O U T I N E
(no module address here)

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = -5C8	NMI	SBE IN TTL PROM - HALT OFFLINE MODE, MEMORY MODULE	NON-FATAL
BP = -5C7	NMI	SBE IN TTL PROM - MEMORY MODULE	NON-FATAL
BP = -5C6	NMI	SBE IN LCSH, RD/WR ADR OFFLINE MODE, MEMORY MODULE	NON-FATAL
BP = -5C5	NMI	OFFLINE MODE, MEMORY MODULE H A L T	NON-FATAL
BP = -0B1	NMI	MBE IN LCS REGION DMA/MPCC	FATAL
BP = -0B2	NMI	MBE IN HIGHER LEVEL SFTWR	FATAL
BP = -5C4	NMI	SBE IN HIGHER LEVEL SFTWR	NON-FATAL
BP = -5C3	NMI	SBE IN HIGHER LEVEL SFTWR OFFLINE MODE - HALT	NON-FATAL
BP = -5C2	NMI	SBE IN MCS , RD/WR of ADR	NON-FATAL
BP = -5C1	NMI	SBE IN MCS , RD/WR of ADR OFFLINE MODE - HALT	NON-FATAL
BP = -5C0	NMI	SBE IN MCS , RD/WR of 0's	NON-FATAL
BP = -5BF	NMI	SBE IN MCS , RD/WR of 0's OFFLINE MODE - HALT	NON-FATAL
BP = -5BE	NMI	SBE IN MCS , RD/WR of A's	NON-FATAL
BP = -5BD	NMI	SBE IN MCS , RD/WR of A's OFFLINE MODE - HALT	NON-FATAL
BP = -5BC	NMI	SBE IN LCSH, RD/WR of 0's	NON-FATAL
BP = -5BB	NMI	SBE IN LCSH, RD/WR of 0's OFFLINE MODE - HALT	NON-FATAL
BP = -5BA	NMI	SBE IN LCSH, RD/WR of A's	NON-FATAL
BP = -5B9	NMI	SBE IN LCSH, RD/WR of A's OFFLINE MODE - HALT	NON-FATAL
BP = -5B8	NMI	SBE IN LCSL, RD/WR of 0's	NON-FATAL
BP = -5B7	NMI	SBE IN LCSL, RD/WR of 0's OFFLINE MODE - HALT	NON-FATAL

APPENDIX B SMC DIAGNOSTICS ERROR REPORTING (Continued)

N M I R O U T I N E
(no module address here)
(Continued)

ERROR_CODE	MODULE	ERROR MESSAGE	TYPE
BP = -5B6	NMI	SBE IN LCSL, RD/WR A's	NON-FATAL
BP = -5B5	NMI	SBE IN LCSL, OFFLINE HALT	NON-FATAL
BP = -5B4	NMI	SBE IN LCSL, RD/WR ADR	NON-FATAL
BP = -5B3	NMI	SBE IN LCSL, OFFLINE HALT	NON-FATAL
BP = -5B2	NMI	SBE IN LCSL, DMA or MPCC	NON-FATAL
BP = -5B1	NMI	SBE IN LCSL, DMA or MPCC OFFLINE MODE - HALT	NON-FATAL