EK-MB020-UD-001

MB20
INTERNAL MEMORY
UNIT DESCRIPTION

digital equipment corporation • marlborough, massachusetts

1st Printing, December 1976

The material in this manual is for informational purposes and is subject to change without notice. Digital Equipment Corporation assumes no responsibility for any errors which may appear in this manual.

Printed in U.S.A.

This document was set on DIGITAL's DECset-8000 computerized typesetting system.

CONTENTS

CONTENTS (Cont)

ILLUSTRATIONS

ILLUSTRATIONS (Cont)

TABLES

PREFACE

The MB20 Internal Memory Unit Description consists of three sections:

Overview
Functional Description
Logic Description

The Overview gives a brief physical description of the memory system and describes its basic operation. MB20 system specifications are provided. The Functional Description gives a detailed description of SBus operation and memory system addressing for the various interleaving modes. It also describes sequence of operation for both a controller and a storage module. Major logic signals are discussed and flowcharts are included. The Logic Description, the most detailed part of the Unit Description, describes the MB20 at the circuit level. Print prefixes are used, providing a direct index into the Field Maintenance Print Set.

SECTION 1
OVERVIEW
KL20

1.1 GENERAL INFORMATION

The MB20 internal memory for the KL10 allows up to four data words to be accessed by a single MBox memory reference. The memory system consists of one to four memory controllers interfaced to the SBus with each controller connecting to and controlling one to four storage modules (Figure 1-1). During interleaved operation, two controllers are addressed at once and cycle together to cause simultaneous storage module operation.

Each storage module (SM) in the MB20 system is a coincident current, ferrite core, 3-wire memory with a basic core cycle time of approximately one μs. SM capacity is 32K (32,768) 37-bit words, with each word consisting of 36 data bits, plus 1 parity bit. Since a system can have a maximum of 16 storage modules, maximum total capacity is 16 × 32K = 512K (524,288) words.

The basic internal memory, contained in the CPU cabinet, consists of two controllers (MC0 and MC1) and associated storage modules. These two controllers are connected in parallel to the MBox via SBus 0. Additional core capacity is provided by the second controller pair (MC2 and MC3) and associated storage modules. These are installed in the I/O cabinet; the controllers connecting to SBus 1, as does the DMA20 Memory Bus Adapter if external memories are connected to supplement the internal memory system. The chart below lists the various MB20 system components. Module utilization is shown in Figure 1-2.

MB20-M (32K × 19-bit core memory section)
- 1-G116 Sense/Inhibit Module
- 1-G236 X-Y Driver Module
- 1-H224-B Stack Module

MB20-E (two storage modules, 64K × 37-bit expansion core memory)
- 4-MB20-M

MB20-G (Controller pair plus two storage modules)
- 1-MB20-E
- 2-M8568 Control Module
- 2-M8565 Timing Module
- 2-M9005 SBus Terminator Module
- 2-H7420 Power Supply
- 4-H744 Power Supply (+5 V)
- 6-H754 Power Supply (+20 V)
- 2-BC20-C SBus Cable
- 1-1213011 Blower Assembly
- 1-7012773 Logic Assembly

NOTE

A fully populated 256K × 37 bit MB20 system consists of one MB20-G, plus three MB20-Es.

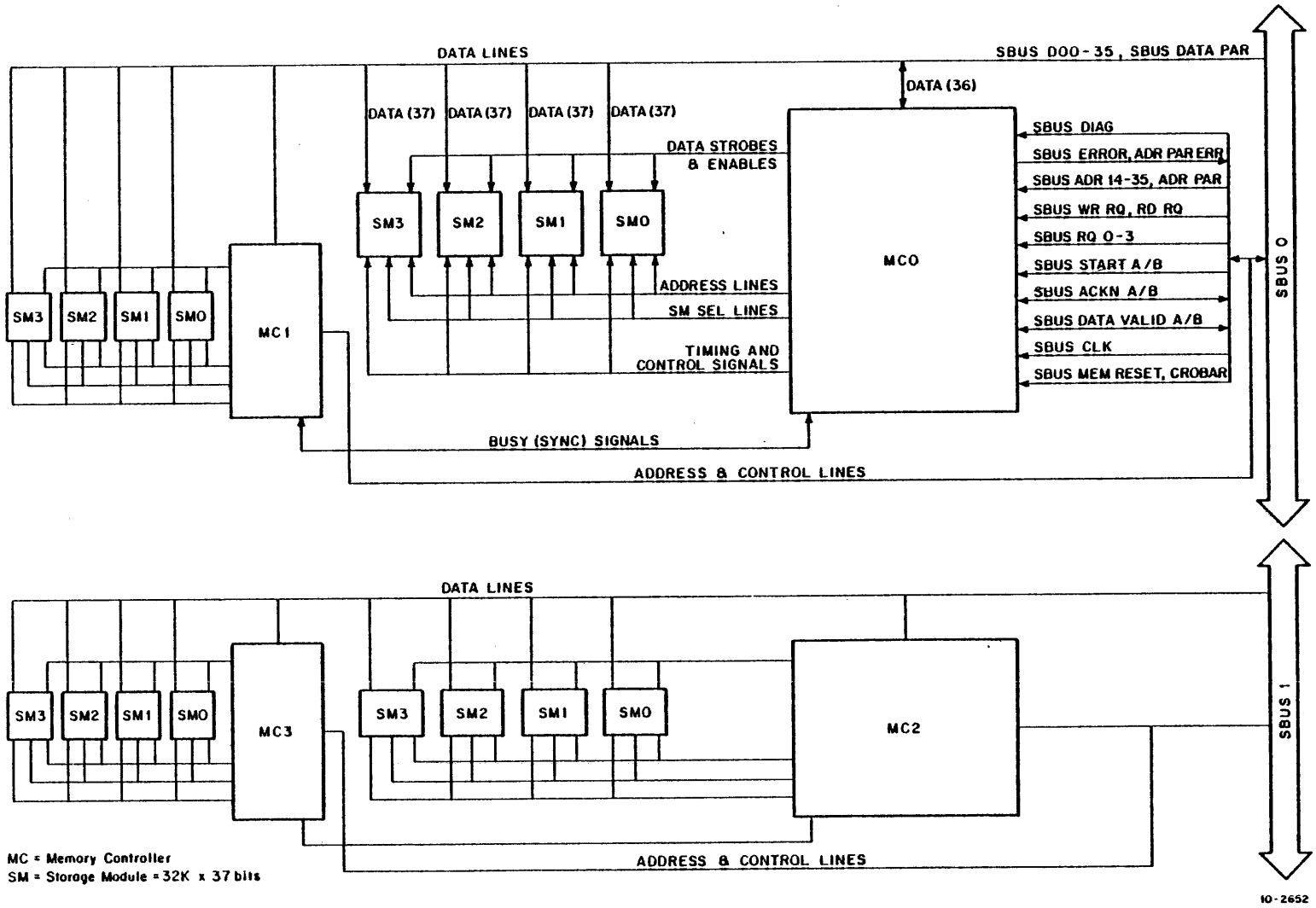


Figure 1-1 MB20 Internal Memory

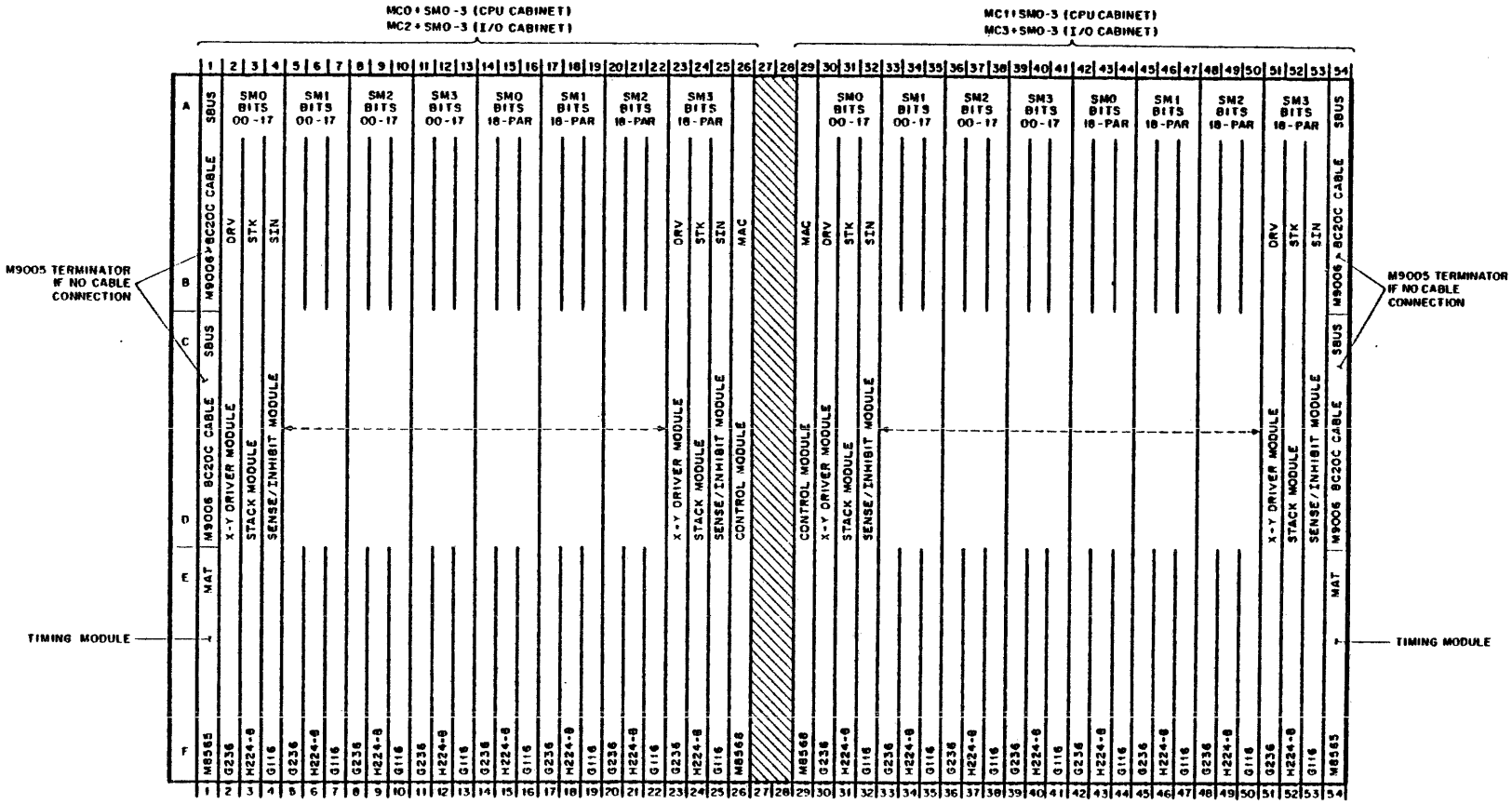


Figure 1-2 MB20 Module Utilization

1.2 BASIC MEMORY OPERATION

The basic memory operations are write, read, read-modify-write, and the diagnostic cycle. During a write operation, data is transferred from the MBox to the MB20 over the SBus and deposited in core under control of the addressed controller(s). During the read operation, data is transferred from core to the SBus under control of the addressed controller(s) and collected by the MBox. The read-modify-write operation transfers data from core (as for the read) for modification by the MBox and then transfers the modified data back to core (as for the write) in one operation. The diagnostic cycle transfers control information from the MBox to the MB20 and then relays status information back to the MBox. A brief description of memory operation follows. A more complete discussion is included in Section 2 where the SBus is described in detail.

1.2.1 Memory Reference

Access to memory is based on a "start/acknowledge" system whereby the MBox asserts an address on the bus, raises a START level, and then waits for an ACKN (acknowledge) response from the addressed controller(s). An acknowledge would be delayed if the addressed controller(s) was still busy from a previous operation. To specify a write or read operation, either the SBUS WR RQ or SBUS RD RQ line is asserted by the MBox along with the START level. Both lines are raised if a read-modify-write operation is to be performed.

Memory references to the MB20 can each access from one to four word locations and they are directed to 4-word groups in core called quad-words. More than one location can be addressed at once because four request lines (RQ 0–3) are included as part of the SBus in addition to a set of conventional address lines (ADR 14–35). Each RQ line acts in conjunction with the ADR lines to specify one word in the quad-word, the address of each word in the group differing only in the value of the two least significant bits (ADR 34 and 35). By encoding the values (00, 01, 10, 11) of the two least significant bits in the four RQ lines and by asserting the appropriate lines, the MBox can specify any or all words (0–3) in the 4-word group. For example, to request words 0 and 2 (addresses 00 and 10), RQ0 and RQ2 are asserted. All RQ lines are raised for a 4-word request.

Although one to four words may be addressed simultaneously on the SBus, data transfer and other communication (including the ACKN for each word) is on a serial basis. The address of the first word to be accessed, the starting address, is specified by the two least significant address lines. This address is encoded in the appropriate RQ line just as any other address in the quad-word that is requested. After the first word, words are cycled in ascending order, modulo 4. For example, a memory reference-requesting four words with a starting address of 01 causes the MB20 to transfer data in the word order 1, 2, 3, 0. A starting address of 00 with request lines RQ 0, 1, and 3 asserted results in words being cycled in the order 0, 1, 3.

1.2.2 Memory Write

For a write operation, the MBox asserts the SBUS ADR lines, places the first word to be written on the data lines, and raises START, WR RQ, and one or more RQ lines. The MB20, in responding to the starting address, generates ACKN, starts a core cycle in the selected storage modules, and strobes the first word from the data lines. If accessing one core location only, the MBox drops START upon receiving ACKN and the SBus operation ends. The addressed controller would remain busy, however, until the core cycle (a core read followed by a core write) ends and the data is written in core.

If more than one word is to be written, the MBox uses the first ACKN signal to place the next word on the data lines. When the MB20 acknowledges the address of this word, generating a second ACKN signal, it again strobes the data lines collecting the word from the SBus. As before, the MBox asserts new data upon receiving ACKN if another word is to be accessed. SBus activity will continue until the address of the last word requested has been acknowledged and the data strobed from the bus. The addressed controller(s) remains busy until the data is written in core.

1.2.3 Memory Read

For a read operation, the MBox asserts the SBUS ADR lines and raises START, together with RD RQ and one or more RQ lines. As for the write operation, the MB20 acknowledges the starting address by generating ACKN and a core cycle is started in the selected storage modules. If accessing one core location, the MBox drops START upon receiving the first ACKN signal. When the data is read from core, the MB20 gates the word onto the data lines and asserts the DATA VALID line, also part of the SBus. DATA VALID is used by the MBox to strobe the word off the data lines, ending the SBus operation. If more than one word has been requested, ACKN and a corresponding DATA VALID are generated for each word until all addresses have been acknowledged and all words have been collected by the MBox. START is negated when the last ACKN is received by the MBox and the SBus operation ends when the last word has been strobed from the data lines. The addressed controller(s) remains busy until data has been restored in core.

1.2.4 Memory Read-Modify-Write

To perform a read-modify-write operation, the MBox asserts the ADR lines, START, RD RQ, WR RQ, and one RQ line. Only one core location may be accessed during the read-modify-write. Upon receiving START, the addressed controller generates ACKN and starts a core cycle. When the data is read from core, the MB20 gates the word on the data lines and generates DATA VALID. The MBox uses this signal to strobe the data from the SBus as in a read operation. Instead of the core cycle continuing (core write follows core read) to restore the data in core (as for a read), the core cycle pauses while the data is manipulated by the MBox. When the data is modified, it is placed on the data lines and the DATA VALID line is asserted, this time by the MBox. The second DATA VALID ends SBus operation and is used by the MB20 to strobe the modified word from the data lines and to initiate the core write cycle. The modified data is then written in memory.

1.2.5 Diagnostic Cycle

The diagnostic cycle is initiated in the CPU by a BLKO PI instruction. Control data is transferred to the addressed controller from the MBox during the first part of the cycle (TO MEMORY) and MB20 status information is returned to the MBox during the second part of the cycle (FROM MEMORY). The controllers (MC0–3) are addressed by their "physical number" (0–3), a hard-wired address.

To start a diagnostic cycle, the MBox places the control information (address limits, interleave mode, clear error, etc.) on the data lines and raises the DIAG line. The MB20 uses DIAG to strobe the data lines and store the control information. When the MBox negates DIAG, it removes the control information from the bus. The MB20 then gates the status information (error flags, address limits, etc.) on the data lines and the MBox collects the information, ending the operation.

1.2.6 Interleaving

Interleaving in the MB20 system is accomplished by allowing the two controllers on each SBus (0 or 1) to operate simultaneously; that is, two controllers can be addressed in one SBus memory reference and each controller can initiate a core cycle in a selected storage module (SM). One controller is enabled to respond to even addresses (RQ 0 and 2) and the other to odd addresses (RQ 1 and 3). One or two SMs may be selected by a controller during one memory reference depending on the mode of operation.

In 2-way interleave mode, one SM can be selected per controller for a total of two active SMs per controller pair. In 4-way interleave mode, two SMs can be selected by a controller allowing four SMs to be cycled in parallel. The 4-way interleave mode, which results in the shortest memory access times, is the normal KL10 operating mode. A no-interleave mode of operation can also be specified where one controller is addressed and one SM is selected. The no-interleave mode (and the 2-way interleave mode) would usually be employed when a system failure precluded the use of the 4-way interleave mode. Interleave operating modes are summarized in Table 1-1.

Table 1-1 Interleave Mode Summary

Interleave Mode	Active Controllers	Selected SMs/Cont	Words/Core Cycle
No Interleave	1	1	1
2-Way Interleave	1–2	1	1–2
4-Way Interleave	1–2	1–2	1–4

The previously described diagnostic cycle allows for program selection of one of the three interleave modes in each controller. It also provides for assigning each controller odd or even status (i.e., whether it is to respond to odd or even addresses) by setting request enable levels (RQ EN 0–3) in the controller at the same time that the interleave mode is assigned. For 2-way and 4-way interleave modes, RQ EN 0 and 2 are set in one controller defining it as even and RQ EN 1 and 3 are set in the other controller defining it as odd. For no-interleave mode, where a single controller must respond to both odd and even addresses, all RQ EN levels (0–3) are set. A controller will appear off-line if no RQ EN levels are set.

NOTE

Results of a memory access are unspecified if the programmer does not load request enables consistent with interleave mode.

1.2.7 Address Boundaries

An address boundary register is incorporated in each MB20 controller to define the fixed portion of available address space represented by the controller and associated storage modules. The register is loaded under program control by means of the SBus diagnostic cycle. The information consists of a memory address, lower address boundary, and upper address boundary. The memory address bits correspond to SBus address bits ADR 14–17 and operate the same way as the address switches mounted on external memory (MG10, etc.); that is, the preset address must match the corresponding address lines in order for the memory to respond. Another condition is that the SBus address must be within certain limits as determined by the memory address acting in conjunction with the lower address boundary and upper address boundary. The lower and upper boundaries correspond to SBus address bits ADR 18–21. The address bits must be equal to or more than the lower limit and equal to or less than the upper limit.

To summarize, successful addressing of the MB20 requires that:

ADR 14–17 = Memory Address
ADR 18–21 $\geqslant$ Lower Address Boundary
ADR 18–21 $\leqslant$ Upper Address Boundary

NOTE

Because a controller pair is addressed in interleaved operation, the corresponding boundary registers in each of the two controllers on an SBus must be set to the same value in 2-way and 4-way interleave mode.

The request enable levels (RQ EN 0–3) are set up in each controller to further specify the particular address (odd, even, all) in the quad-word for which a controller, and only one controller, will respond. Thus, another condition for selecting a controller is:

RQ n = RQ EN n, n = 0–3

As mentioned previously, a controller will appear off-line if no RQ EN levels are set. It will also appear off-line if the upper address boundary is set to a value lower than the lower address boundary, or if the lower address boundary is set greater than the upper address boundary.

1.2.8 Error Checking

A parity bit is written in a core location along with the 36 bits of data. Parity is odd and it is checked by the MBox after the word is read from core and received on the SBus during the read and read-modify-write operations. The MBox also checks for nonexistent memory. If ACKN is not received to indicate the presence of memory 80 μs after the assertion of START, a time-out sequence terminates the operation and prevents a hung condition. Both DATA PARITY ERROR and NONEXISTENT MEMORY are flagged in the MBox and cause an APR interrupt. The MBox also preserves the failing address in its Error Address register (ERA).

Error conditions checked by the MB20 are ADDRESS PARITY ERROR and INCOMPLETE REQUEST. If bad parity is detected on the SBus address and request lines (ADR 14–35, RQ 0–3, RD RQ, WR RQ) when a controller is referenced, normal bus dialogue takes place between the MBox and the MB20, but read/write currents are inhibited by the controller in the referenced storage modules. This causes write data transferred from the MBox not to be deposited in core and zeros will be passed to the MBox as read data. Data parity is also returned as zero, causing the MBox to detect a data parity error for a read or read-modify-write operation when an address parity error occurs. In addition to immediately setting the internal error flag for an address parity error, the controller also raises the SBUS ADR PAR ERR line. This results in the MBox flagging the error condition and causing an APR interrupt. As for data parity and nonexistent memory errors, the failing address is held in the ERA.

If a controller starts an operation and then fails to complete a memory reference after 10.2 μs, indicating a hung condition or incomplete request, a time-out occurs in the controller which sets the internal error flag and clears the controller to its initial state. The SBUS ERROR line is also asserted, causing an APR interrupt in the CPU.

Controller error flags may be read by means of the SBus diagnostic cycle. Error flags set during a previous SBus operation are not cleared when another memory reference is made. Once set, error flags can be cleared only by means of the diagnostic cycle (i.e., BLKO PI).

1.2.9 Diagnostic Features

Loop-around mode is a diagnostic feature of the memory system which allows the data path between the MBox and the MB20 to be checked without actually reading or writing the data in core. The mode is used mainly by the diagnostic programmer in isolating system failures. It is set in a controller by means of the diagnostic cycle. When loop-around mode is set and followed by a write operation, normal SBus operation takes place and the MBox data is strobed into the data register(s) of the selected SM(s). However, SM read/write currents are inhibited by the controller and the data is not written in core. If an SBus read operation is initiated next (controller still in loop-around mode), again SBus operation is normal; that is, the data loaded by the previous write (still in data registers) is strobed onto the SBus and collected by the MBox. However, read/write currents are still inhibited and data is not actually read from core. Thus, loop-around mode tests the data path for both write and read operations independent of core memory selection and circuitry. Loop-around mode is automatically cleared in a controller at the conclusion of the read operation.

Another diagnostic aid is the ability to test SM operation under margin control. Margins are turned on by means of the diagnostic cycle and provision is made to change:

1. The amplitude of X-Y select currents
2. The timing of the sense strobe
3. The sense amplifier threshold voltage.

Margining allows problems to be detected in advance of a hard failure and it can also facilitate troubleshooting by forcing intermittent problems to become constant.

The KL10 system's clocks may be single-stepped under program control. Since the MB20 is sequenced by the SBus clock, the memory system will step with the rest of the system during this type of diagnostic operation. The SM core cycle is not clock-dependent (delay line timing) but slow clocking the MB20 is useful in isolating certain controller problems associated with high speed operation - noise problems, race conditions, etc.

1.3 SPECIFICATIONS

Specifications for the MB20 memory system are listed in Table 1-2.

Table 1-2 MB20 System Specifications

Memory Type	Magnetic core, read/write, coincident current, random access
Organization	Planar, 3-D, 3-wire
Cycle Time (SBus Read Operation)*	1920 ns (4-word CPU reference in 4-way interleave mode 2320 ns (4-word Channel reference in 4-way interleave mode)
Cycle Time (SBus Write Operation)*	1920 ns (4-word CPU or Channel reference in 4-way interleave mode)
Read Access Time*	1040 ns (first word) +240 ns (each additional word in 4-way interleave mode)
Write Access Time*	320 ns (first word) +240 ns (each additional word in 4-way interleave mode)
Voltage Requirements	+5 V ± 5% +20 V ± 5% -5 V ± 5%
Environment Ambient Temperature	60° - 90° F
Relative Humidity	20% - 80% (non-condensing)

*Measured from start of Cache cycle in MBox; MBOX CLK frequency = 25 MHz, SBUS CLK frequency = 25/4 MHz.

SECTION 2
FUNCTIONAL DESCRIPTION

This section contains a system-level description of SBus and MB20 operation. Major functional elements are shown in Figure 2-1.

2.1 SBUS OPERATION

The SBus connects the MA20 Internal Memory, the MB20 Internal Memory, and the DMA20 Memory Bus Adapter to the MBox. The following discussion is concerned only with SBus operation as it relates to MB20 internal memory. Information flow on the SBus is shown in Figure 2-1. Table 2-1 summarizes the functions of the various signals.

2.1.1 MBox and Memory Synchronization

The SBUS CLK INT signal provides a continuous clock train that is used by the MB20 to sequence logic and to synchronize its operation with the MBox. The negative-going edge corresponds to the phase A clock in the MBox and the positive-going edge corresponds to phase B. To maintain synchronization with the MBox, the MB20 uses SBUS CLK INT to generate phase A and B clocks of its own and these are deskewed to coincide exactly with the corresponding MBox clock. Adjustable delay lines are provided in the controller for this purpose. The deskew procedure is detailed on drawing D-BS-MB20-0-INS of the Field Maintenance print set.

NOTE

The MB20 must be deskewed during installation and following the replacement of the M8565 Timing module in a controller. Deskewing is also necessary following the replacement of an SBus cable or an M8519 SBus Translator module in the KL10 CPU.

With the clocks synchronized in the MBox and MB20, and with control bus propagation delays less than the period between clocks, SBus control signals generated on one end of the bus by a particular clock phase can be received at the other end of the bus on the next clock of the same phase without the need for synchronizing logic. For example, SBUS ERROR is transmitted in the controller on phase A and strobed in the MBox one clock period later, also on phase A. Other control signals, such as START, are also linked to a particular phase. With no time lost in synchronizing the bus signals to internal logic, memory access times are held to a minimum.

To further speed SBus operation, two START lines are provided on the bus to allow the MBox to begin an operation on either phase. START A is generated and received on phase A; START B is linked to phase B. Similarly, two ACKN lines (A and B) and two DATA VALID lines (A and B) are employed on the bus to speed operation by shortening controller response time.

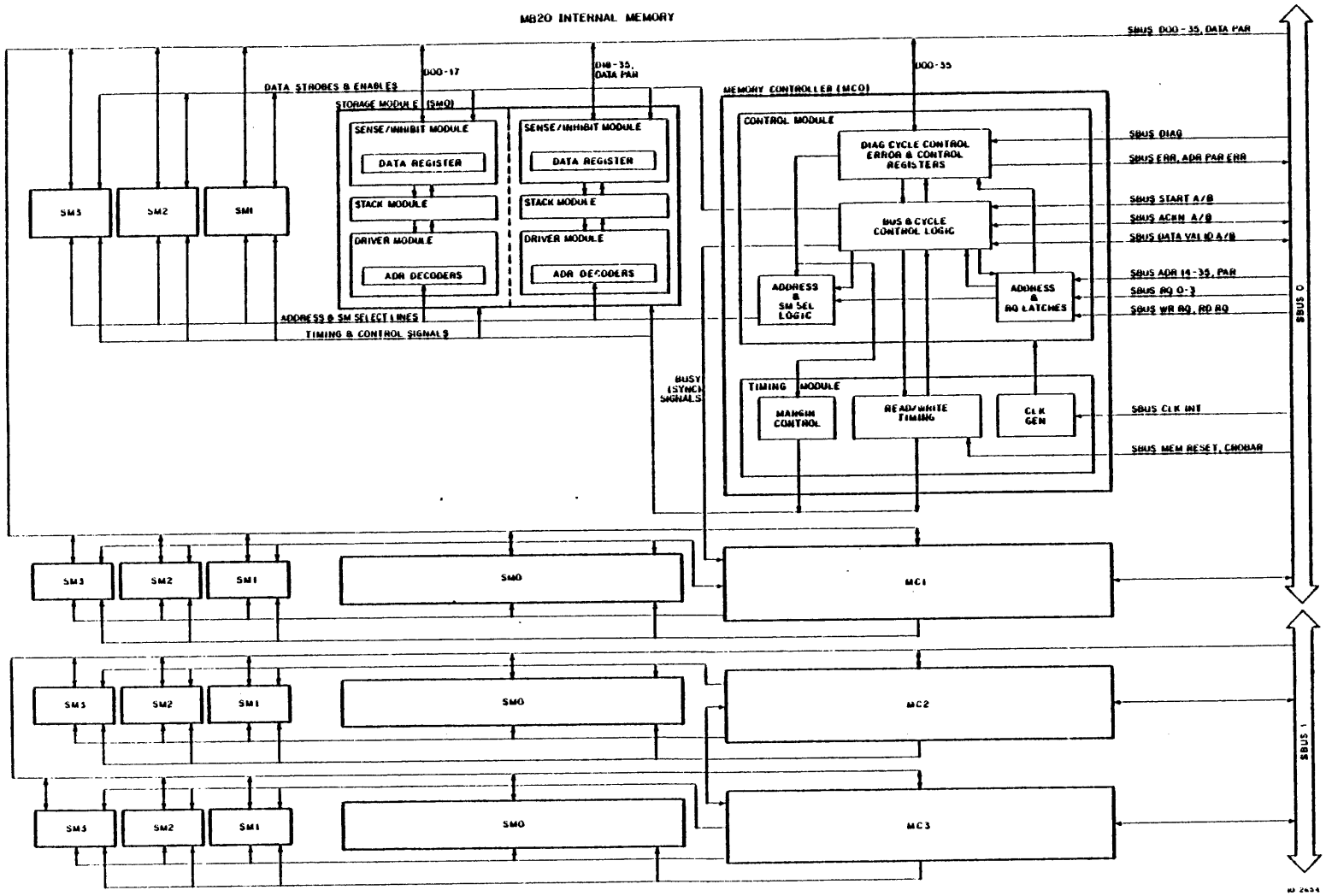


Figure 2-1 MB20 Functional Block Diagram

Table 2-1 SBus Signal Summary

Signal	Direction	Function
ADR 14–33	MBox to MB20	Quad-word address.
ADR 34, 35	MBox to MB20	Starting address. Specifies the first word in the quad-word to be accessed.
ADR PAR	MBox to MB20	Parity bit (odd) for ADR 14–35, RD RQ, WR RQ, and RQ 0–3.
RQ 0–3	MBox to MB20	Word requests. Specify the words in the quad-word to be accessed.
RD RQ	MBox to MB20	Specifies that a read operation is to be performed. Specifies a read-modify-write operation if WR RQ = 1.
WR RQ	MBox to MB20	Specifies that a write operation is to be performed. Specifies a read-modify-write operation if RD RQ = 1.
START A/B	MBox to MB20	Causes execution of the operation specified by RD RQ and WR RQ.
DIAG	MBox to MB20	Causes execution of a diagnostic cycle.
D00–35	Bidirectional	Transfer write data and diagnostic cycle control information to MB20. Transfer read data and diagnostic cycle status information from MB20.
DATA PAR	Bidirectional	Data parity bit (odd) for D00–35 during transfer of read and write data.
ACKN A/B	MB20 to MBox	Address acknowledge.
DATA VALID A/B	Bidirectional	Data strobe – Indicates read or write data asserted on D00–35.
ADR PAR ERR	MB20 to MBox	Indicates an address parity error has been detected.
ERROR	MB20 to MBox	Indicates an incomplete request error has been detected.
CLK INT	MBox to MB20	SBus clock for internal memory.
MEM RESET	MBox to MB20	Clears MB20 to initial state.
CROBAR	MBox to MB20	Clears MB20 to initial state during system power-up and power-down.

2.1.2 Diagnostic Cycle

The SBus diagnostic cycle is provided so that the programmer can load control information in a memory controller, and in the same operation, read back status from the same controller. The BLKO PI instruction fetches the control information from the effective address (E) in memory, transfers the control information from the EBox to the MBox, and signals the MBox to execute an SBus diagnostic cycle. The 36 bits of control information are then transferred over the SBus to the controller (TO MEMORY) during the first half of the diagnostic cycle and 36 bits of status information are returned to the MBox (FROM MEMORY) during the second half of the cycle. When the MBox raises a response signal, the EBox collects the status information and transfers the information to memory (E+1) ending the BLKO PI. A timing diagram for the diagnostic cycle is shown in Figure 2-2. The cycle always starts on the clock derived from phase A of SBUS CLK INT and has a duration of four phase A clock intervals.

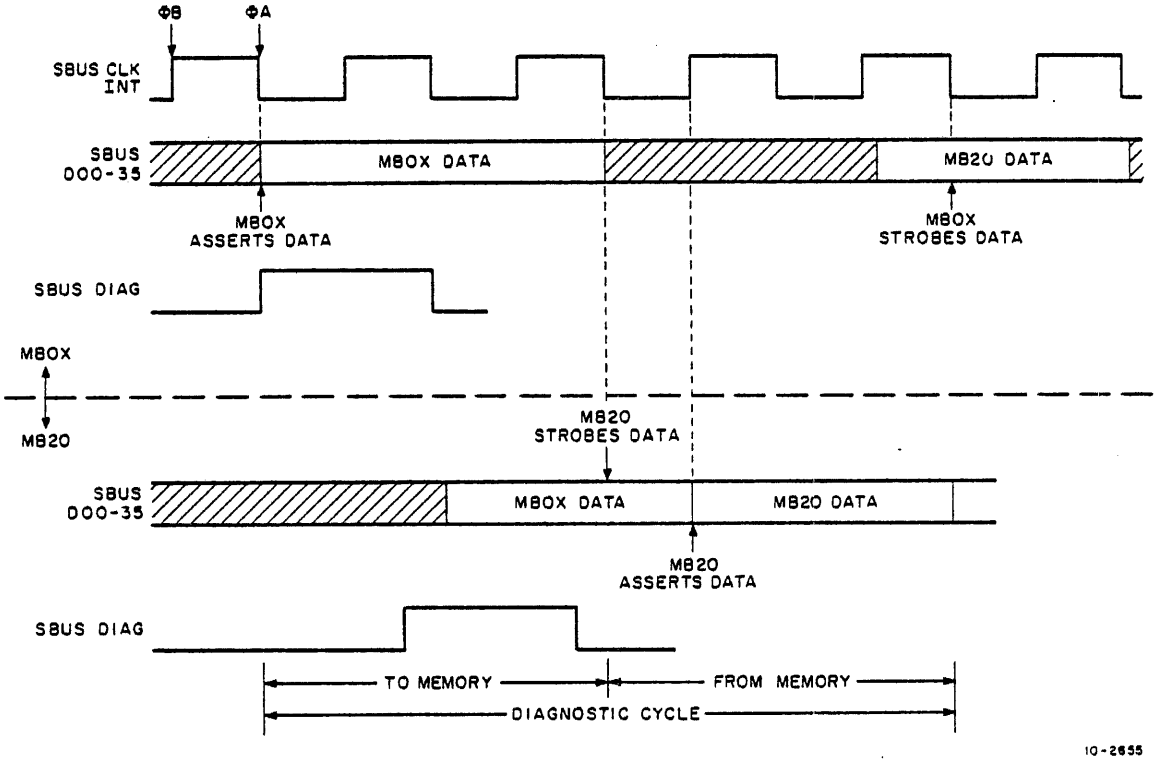


Figure 2-2 SBus Diagnostic Cycle Timing

In the TO MEMORY portion of the cycle, the MBox raises SBUS DIAG and asserts a controller address on data lines D00–04, a function code on data lines D31–35, and control bits on data lines D05–30. The function code, either 0 or 1, specifies the type of control information to be loaded in the addressed controller. This control information is strobed off the data lines on the third phase A clock. In the second or FROM MEMORY portion of the cycle, the controller asserts status information on data lines D00–35 and it is collected by the MBox on the fifth phase A clock. The status information that is collected, like the control information that is loaded, depends on the function code asserted in the first half of the cycle. The controller does not generate a parity bit (SBUS DATA PAR equals zero) for the status information and data parity is not checked by the MBox during the diagnostic cycle. Figure 2-3 shows the control and status information specified by each function code. The diagnostic cycle information is summarized in Table 2-2.

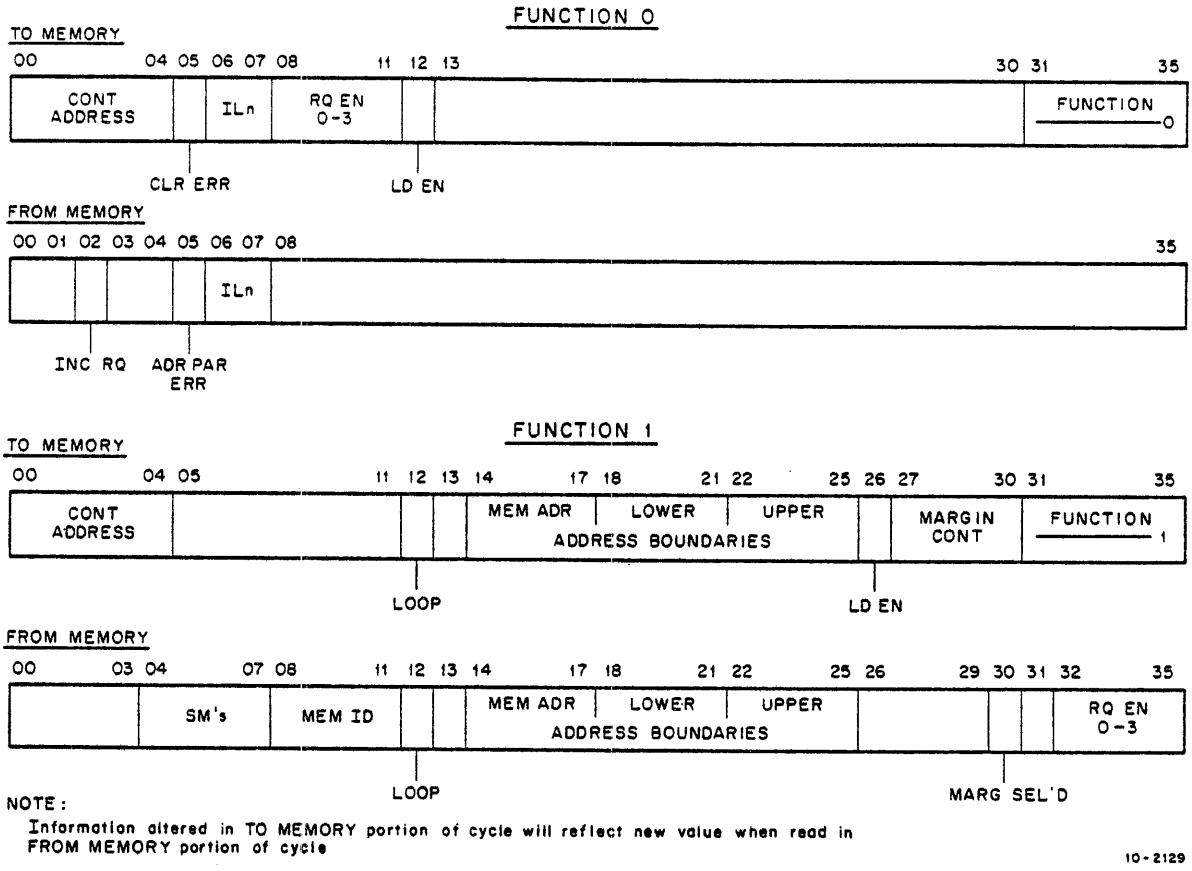


Figure 2-3 MB20 Diagnostic Cycle Data

2.1.3 Word Selection

Each memory reference over the SBus is made to a 4-word block in memory called a quad-word. Four request lines (RQ 0–3) are provided on the bus so that any or all words in the quad-word (words 0–3) can be selected in a single operation. The address of each word differs from the others only in the value of the two least significant bits and each RQ line corresponds to one of the four addresses as shown in Figure 2-4.

To make a quad-word reference, the MBox asserts the appropriate RQ line(s) along with SBUS ADR 14–35. ADR 14–33 (called the quad-word address) is equal to the most significant part of the address for all four words. The two least significant address bits for each word are encoded in the RQ lines. ADR 34 and 35 hold the starting address and point to the first word in the quad-word to be accessed. Word selection is best described by example and Table 2-3 shows ADR and RQ line values for various memory references.

Although one to four words can be selected simultaneously on the SBus, the data lines are only one word wide, requiring that words be transferred serially for multi-word requests. ACKN and DATA VALID signals are also generated serially. This is because they act as data strobes and must correspond in time to the changing data. The words are cycled in ascending order, modulo 4, beginning with the word specified by the starting address. Word order is shown for the examples given in Table 2-3.

Table 2-2 Diagnostic Cycle Data Description

Function 0

TO MEMORY

Bits 00–04					Controller address – Each MB20 controller has a hard-wired address 0–3. MC0 and MC1 connect to SBus 0; MC2 and MC3 to SBus (Figure 1-1).
00	01	02	03	04	
0	0	0	0	0	MC0
0	0	0	0	1	MC1
0	0	0	1	0	MC2
0	0	0	1	1	MC3
0	0	1	0	0	(DMA20 – not an MB20 address)
Bit 05					Clear error – Clears controller's internal error flags INCOMPLETE REQUEST and ADR PAR ERR.
1					
Bits 06, 07					Set interleave mode – Bits are binary-encoded to set interleave mode (Subsection 1.2.6).
06	07				
0	0				(Off-line for DMA20 – not an MB20 operating mode)
0	1				No-interleave
1	0				2-way interleave
1	1				4-way interleave
Bits 08–11					Set Request Enables – Assigns controller odd-even status (Subsection 1.2.6).
08	09	10	11		
RQ EN	RQ EN	RQ EN	RQ EN		
0	1	2	3		
0	0	0	0		Controller off-line
1	0	1	0		Controller even (2-way and 4-way interleave modes)
0	1	0	1		Controller odd (2-way and 4-way interleave modes)
1	1	1	1		Controller odd and even (no-interleave mode)
Bit 12					Load Enable – Enables loading of bits 6–11.
1					(Load and read back)
0					(Read only)

Table 2-2 Diagnostic Cycle Data Description (Cont)

					Function 0
					TO MEMORY
Bits 31–35					
31	32	33	34	35	
-	-	-	-	0	Function 0
					Function 0
					FROM MEMORY
Bit 02					Incomplete request – Indicates controller active for 10.2 μs: hung controller (Subsection 1.2.8).
1					
Bit 05					Address parity error – Indicates bad parity detected for information on SBUS ADR, RQn, RD RQ, and WR RQ lines (Subsection 1.2.8).
1					
Bits 06, 07					Interleave mode – Indicates interleave mode loaded in first half of function 0.
					Function 1
					TO MEMORY
Bits 00–04					Controller address – Refer to Table 2-1, bits 00–04.
Bit 12					Set loop-around mode – Inhibits read/write currents in storage modules. A diagnostic feature for checking data path, independent of core activity (Subsection 1.2.9).
1					
Bits 14–17					Set memory address – Correspond to SBUS ADR 14–17. Must match the SBus address lines if a controller is to respond to a memory reference (Subsection 1.2.7).
Bits 18–21					Set lower address boundary – Correspond to SBUS ADR 18–21. Act in conjunction with memory address (bits 14–17) to specify lower address limit (Subsection 1.2.7).

Table 2-2 Diagnostic Cycle Data Description (Cont)

Function 1

TO MEMORY

Bits 22–25					Set upper address boundary – Correspond to SBUS ADR 18–21. Act in conjunction with memory address (bits 14–17) to specify upper address limit (Subsection 1.2.7).
Bit 26					Load Enable – Enable loading of bits 14–25.
1					(Load and read back)
0					(Read only)
Bits 27–30					Set margin control – Turn on margin control as specified below. Only one margin should be turned on at any one time (Subsection 1.2.9).
27	28	29	30		
0	0	0	0		No Op
0	0	0	1		Clear all margins
0	0	1	X		Current margin X = 0 – Low margin
0	1	0	X		Strobe margin X = 1 – High margin
1	0	0	X		Threshold margin
Bits 31–35					Function Code
31	32	33	34	35	
–	–	–	–	1	

Function 1

FROM MEMORY

Bits 04–07				Storage modules connected – These hard-wired bits indicate the number of storage modules connected to a controller.
04	05	06	07	
SM 3	SM 2	SM 1	SM 0	
X	X	X	X	X = 1 – SM connected X = 0 – SM not connected

Table 2-2 Diagnostic Cycle Data Description (Cont)

	Function 1
	FROM MEMORY
Bits 08–11	Memory ID – These hard-wired bits identify memory type.
08 09 10 11	
0 0 1 1	MB20
Bit 12	Loop-around mode – Indicates controller in loop-around mode.
1	
Bits 14–25	Address boundaries – Indicates address boundaries loaded in first half of function 1.
Bit 30	Margins selected – Indicates that current, strobe, or threshold margin control is on.
1	
Bits 32–35	Request enables – Indicates which RQ ENs are set. Loaded in first half of function 0.
32 33 34 35	
RQ EN 0 RQ EN 1 RQ EN 2 RQ EN 3	

2.1.4 Interleaved Operation

Both 2- and 4-way interleaving is accomplished in the MB20 by allowing two controllers to operate in parallel. In 2-way interleave mode, each controller can select and initiate a core cycle in one SM, allowing two core locations to be accessed at once and in one core cycle time. In 4-way interleave mode, each controller can cycle two SMs in parallel and up to four core locations can be accessed in one core cycle time. One controller handles the even addresses in the quad-word, responding to RQ0 and RQ2; while the other controller handles the odd addresses, responding to RQ1 and RQ3. RQ EN levels in each controller determine the addresses for which it will respond. These are set initially via the diagnostic cycle. RQ EN 0 and 2 equal to 1 define the even controller. Setting RQ EN 1 and 3 establishes a controller as odd. The MB20 has a no-interleave mode where only one controller responds to a memory reference and only one SM may be cycled at once. For this mode, just one core location is accessed per core cycle. Because a controller must handle both odd and even addresses in no-interleave mode, all RQ EN levels (0–3) are set.

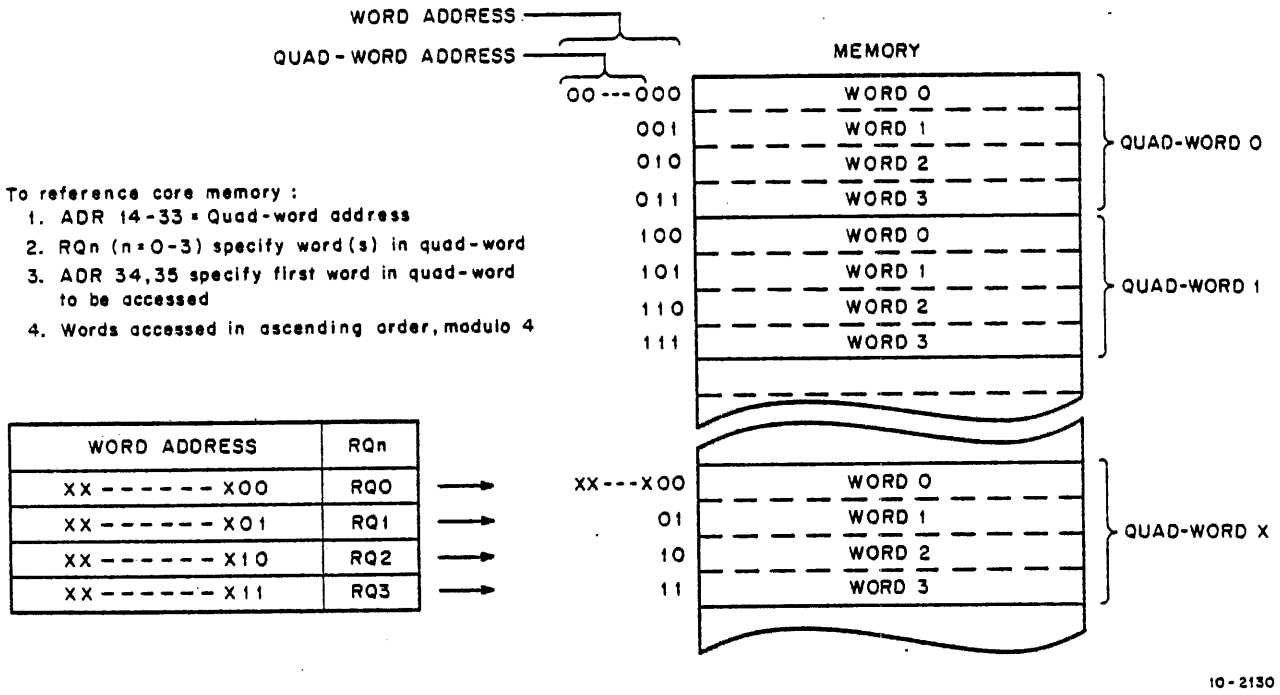


Figure 2-4 Word Selection

Table 2-3 Word Selection-Examples

Example	SBus ADR lines 14 —— 33 *34 *35	SBus RQn 0 1 2 3	Word Order
1 word request, address 100_8	0——01 000 0 0 0	1 0 0 0	0
2 word request, address 134** 135	0——01 011 1 0 0	1 1 0 0	0, 1
3 word request, address 121 122** 123	0——01 010 0 1 0	0 1 1 1	2, 3, 1
4 word request, address 100 101 102 103**	0——01 000 0 1 1	1 1 1 1	3, 0, 1, 2

*ADR 34, 35 = Starting address = S
**First word to be accessed.

Figure 2-5 illustrates memory response for all possible combinations of words requested (based on the starting address) in each interleave mode. The starting address is designated as "S" and can specify any word (0–3) in the quad-word. If S specifies word 1, then S+1 corresponds to word 2 and S+2 corresponds to word 3. Because words are cycled in ascending order, modulo 4, S+3 would specify word 0. The controllers that are active for any memory reference (and the addresses each is responding to) are indicated in the figure. Response to the starting address is by the odd controller if S is odd and by the even controller if S is even. If responding to one address, an active controller initiates one core cycle in one SM. If responding to two addresses (4-way interleave mode), an active controller initiates two parallel core cycles in two SMs. As can be seen in the figure, minimum total access times occur in 4-way interleave mode where two active controllers can respond to four addresses during one core cycle interval (four parallel SM core cycles). This is the normal KL10 operating mode. The 2-way interleave mode and the no-interleave mode of operation are usually not employed unless required due to a system failure.

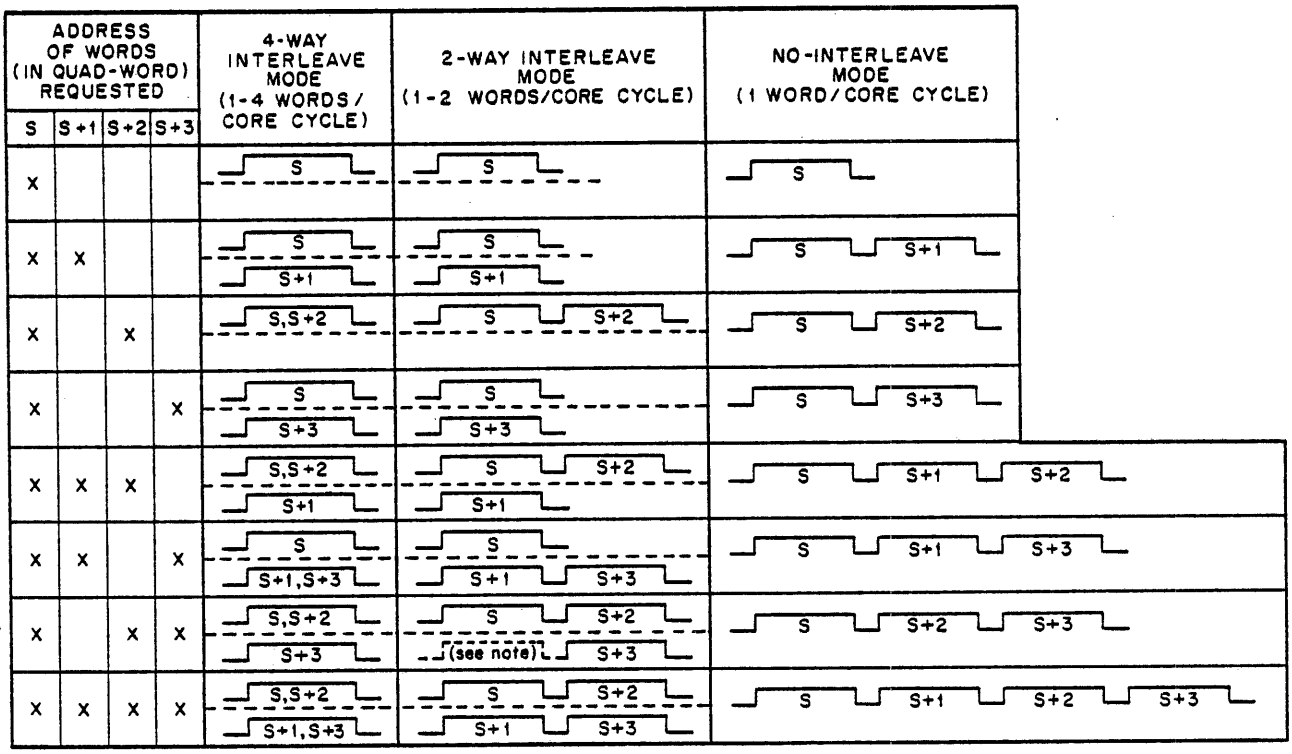
If a storage module failure occurs, a system can continue to be operated in 4-way interleave mode but with a maximum of only two SMs per controller in the failing controller pair configuration. This means that three operative SMs have to be addressed out of the configuration along with the bad SM, greatly reducing total storage capacity; 4-way interleaving is maintained, however. Another option is to switch operation to 2-way interleave mode, in which case (with a SM failure), three working SMs may be operated per controller in the affected controller pair configuration. In this instance, only one working SM is addressed out of the system along with the bad one. However, as shown in Figure 2-5, total access times for memory references involving two odd or two even addresses increase considerably in 2-way interleave mode. This is because a controller responds to only one address per core cycle time and an odd or even controller that is required to handle two addresses must take two cycles. If core capacity is of prime consideration when a SM fails, a controller pair may be switched to no-interleave mode and only the bad SM is addressed out of the system. In this mode, total access times are greatest since a core cycle is required for each word requested. (The no-interleave mode must be implemented whenever one controller in a pair is not in operation.) Rules regarding controller addressing and SM configuration for each interleave mode are given in Subsection 2.2.4.

2.1.5 SBus Write Operation

Timing for the SBus write operation is shown in Figure 2-6. The diagram is specifically for a 4-word request in a 4-way interleave mode, but it illustrates the sequence of SBus signals for any write operation. Figure 2-6 also shows approximate timing for major signals when a 4-word request is made in 2-way and no-interleave modes.

The MBox begins a write operation by asserting the 22 SBus address lines ADR 14–35. It then places the first word to be written in memory on the data lines (SBUS D00–35 and DATA PAR) and asserts SBUS START, SBUS WR RQ, and one or more SBUS RQ 0–3 lines. Address parity is provided on the SBus for the 22 address lines and for all request lines which include RQ 0–3, WR RQ, and RD RQ. (RD RQ equals ZERO for a write operation.) Because the parity computation includes the request levels, the SBUS ADR PAR line is not valid until shortly after the requests are generated and asserted on the bus.

Either one of two START levels may be asserted by the MBox as explained in Subsection 2.1.1. Figure 2-6 shows START A, coinciding with phase A of the SBUS CLOCK, as initiating MB20 operation. The operation is a 4-word request in 4-way interleave mode. When the START A level is received, the two addressed controllers on a bus go active on phase A with each controller initiating a core cycle in the selected storage modules. Also, the controller enabled to handle the starting address (S) responds by generating ACKN on the phase corresponding to the START level received. Both controllers then go busy and the controller acknowledging the starting address strobes the first word plus parity off the SBus data lines into the data register of the selected SM.

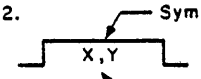


NOTE :
Dummy core cycle is initiated when S, S+2 and S+3 are accessed in 2-way interleave mode.

LEGEND:

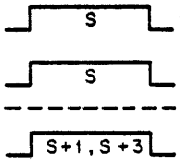
1. S = Starting address. S can be 00,01,10 or 11 corresponding to words 0,1,2 or 3 in quad-word. Addresses are in ascending order, modulo 4. S,S+1,------ 01,10,11,00,01,-------

2. Symbol represents core cycle initiated by a controller.



Designates address of word (or words) in quad-word accessed during core cycle interval.

EXAMPLES:



= One active controller, either odd or even. One word accessed. One SM active.

= Two active controllers, odd and even, operating in parrallel. Three words accessed. One SM active on one controller, two SM's on the other.

10 - 2131

Figure 2-5 MB20 Memory Response in 4-Way, 2-Way, and No-Interleave Modes

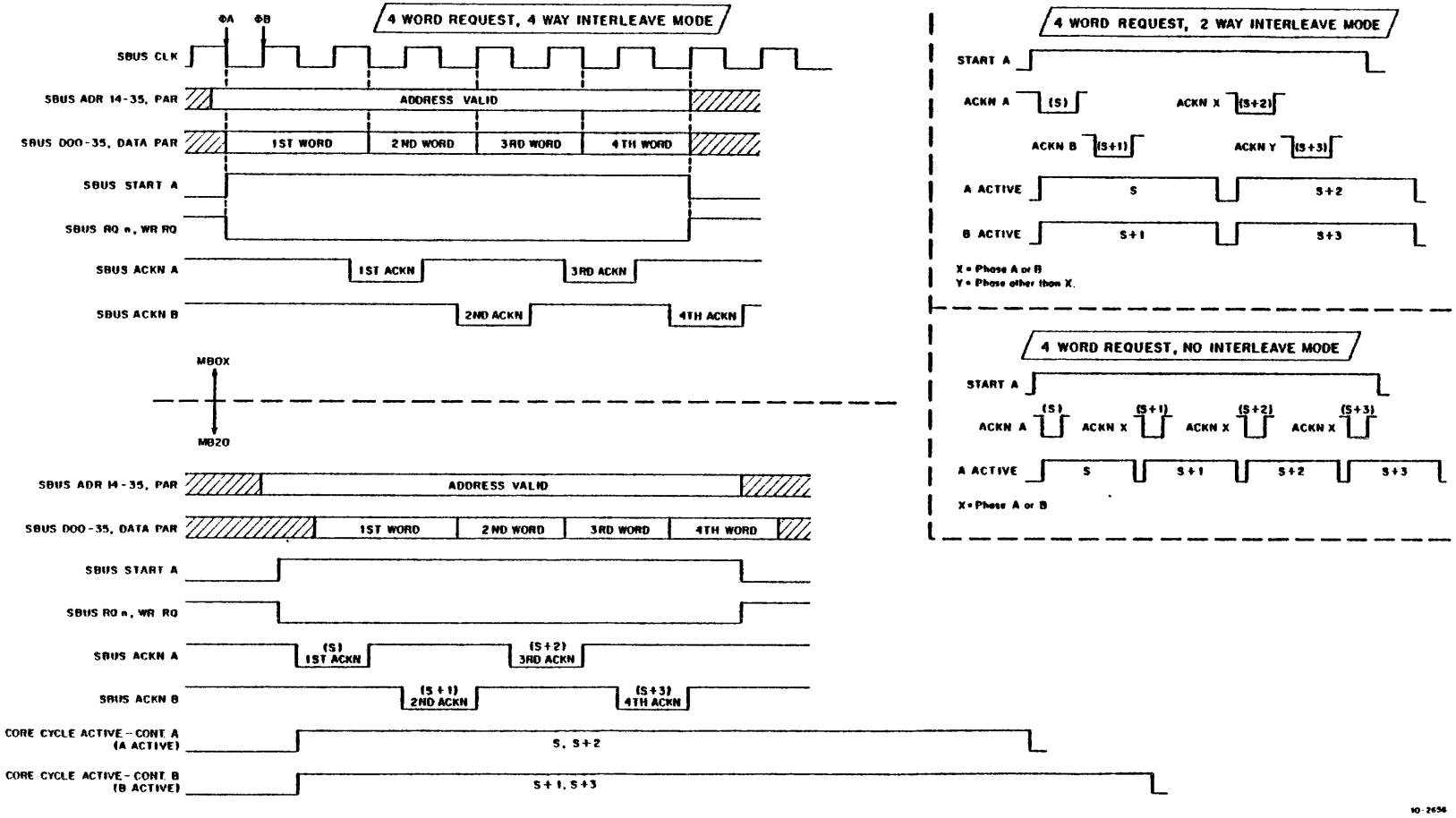


Figure 2-6 SBus Write Timing Diagram (MB20)

When the first ACKN is received by the MBox, the second of the four words to be written is asserted on the data lines. Because the address (S+1) of the next word is odd if the first was even and vice versa, the other controller in the pair must access the second word and it responds by generating a second ACKN on the SBus. Again the data lines are strobed and again the next word is asserted by the MBox when ACKN is received. Since the operation is 4-way interleaved and all four words are collected and deposited in core during one core cycle time, ACKN signals for addresses S+2 and S+3 immediately follow and the last two words are strobed from the data lines. As for the first two ACKN signals, the third and fourth are generated alternately by the controllers since one address is odd and the other even. The MBox disconnects, dropping the START level and ending the operation, when the last ACKN is received. The controllers remain busy, however, until the core cycle is complete. If the MBox directs another reference to a controller or controller pair for which a busy condition exists, the START level is ignored until the previous operation ends.

If the 4-word reference described above had been directed to controllers operating in 2-way or no-interleave mode, two or four successive core cycles would be required. As a result, relative timing between ACKN signals (and the resulting bus action) would be different as shown in the upper-right portion of Figure 2-6. The basic sequence of bus signals remains the same in any mode, however, with addresses acknowledged and words written in core in the order S, S+1, S+2, and S+3.

ACKN signals are asserted on the bus for one SBus clock period. The first ACKN generated in any mode of operation is always on the starting phase. In an operation requiring successive core cycles (2-way and no-interleave modes), the first ACKN in core cycle intervals following the initial one can be generated on either phase, no matter what the asserted START level is. Also, in any one core cycle interval, the first and third ACKN signals are asserted on one phase and the second and fourth are asserted on the other phase.

2.1.6 SBus Read Operation

Timing for the SBus read operation is shown in Figure 2-7. The diagram is specifically for a 4-word request in 4-way interleave mode, but it illustrates the sequence of SBus signals for any read operation. Figure 2-7 also shows approximate timing for major signals when a 4-word request is made in 2-way and no-interleave modes.

The MBox begins a read operation by asserting the SBus address lines ADR 14–35. It then raises either START A or B, (Subsection 2.1.1), together with SBUS RD RQ and one or more SBUS RQ 0–3 lines. Odd parity is computed by the MBox for the ADR, WR RQ (WR RQ equals ZERO for a read operation), RD RQ, and RQ 0–3 lines. The parity line SBUS ADR PAR is valid shortly after the START line is asserted.

Figure 2-7 shows START A initiating the read operation in the MB20. The operation is a 4-word request in 4-way interleave mode. When the START A level is received, the addressed controllers start on phase A with each initiating a core cycle in the selected SMs. Also, the controller enabled to handle the first word acknowledges the starting address by generating ACKN. Both controllers then go busy and the addresses of the remaining three words are acknowledged in succession, as described for the write operation (Subsection 2.1.5).

As the four words of data are read from core into the SM data registers, the controllers alternately gate the words onto the SBus data lines and generate a DATA VALID signal corresponding to each word. As seen in Figure 2-7, words are placed on the bus in the same order as the addresses are acknowledged. The MBox uses the DATA VALID signals to strobe the data lines. The SBus operation ends when the last word has been collected. As for the SBus write operation, the MBox drops START and negates the other bus control lines when the last ACKN is received. The controller pair remains busy until all DATA VALID signals have been generated and the core cycles have ended. If another memory reference is directed to a controller or controller pair for which a busy condition exists, the START level will be ignored until the previously initiated core cycles have ended.

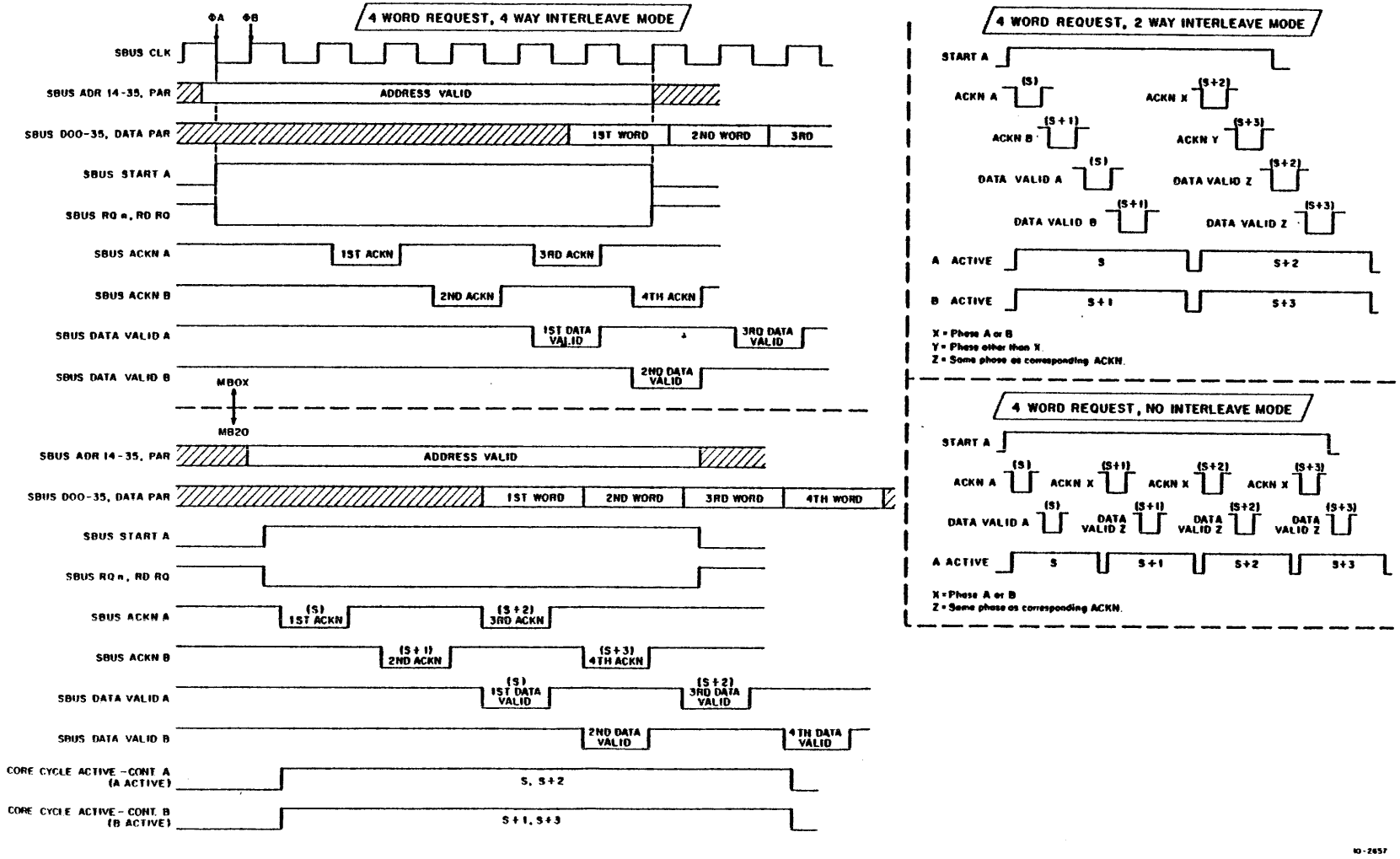


Figure 2-7 SBus Read Timing Diagram (MB20)

For a 4-word reference, two successive core cycles are required in 2-way interleave mode and four successive core cycles are required in 4-way interleave mode. Although the timing between control signals differs (Figure 2-7, upper-right), word order and the sequence of bus signals is the same.

ACKN and DATA VALID signals are asserted on the bus for one SBus clock period. The clock phase on which ACKN signals are generated is the same as for the write operation. A DATA VALID signal is generated on the same phase as the ACKN signal for the same word.

2.1.7 SBus Read-Modify-Write Operation (RMW)

Timing for the SBus read-modify-write operation is shown in Figure 2-8. The MBox begins the operation by asserting the SBus address lines ADR 14–35. It then raises either SBUS START A or SBUS START B (Subsection 2.1.1) and the request lines SBUS RD RQ, SBUS WR RQ, and one SBUS RQ line. Parity is generated by the MBox for the address and request lines and SBUS ADR PAR becomes valid shortly after these lines are asserted on the bus.

Only one core location can be accessed by the RMW operation and only one controller will respond in any interleave mode. When START is received by the addressed controller, the controller acknowledges the SBus address (by generating ACKN on the starting phase) and starts a core cycle in the selected SM. The controller then goes busy and data is read from core into the SM data register as in a normal read operation. The contents of the data register are then gated onto the SBus data lines. Also, the DATA VALID line is asserted to signal the MBox that data is on the bus. The MBox then collects the data word, but instead of the normal read continuing in the MB20 (read data restored in core), the core cycle pauses and the core write operation is delayed until the MBox modifies the data and sends it over the SBus.

DATA VALID is asserted for the second time in the operation (this time by the MBox) when the modified data is placed on the data lines. When DATA VALID is received by the MB20, the contents of the data lines are strobed into the SM data register. The core write cycle is then allowed to take place and the modified data is deposited in core. The controller remains busy until the end of the core cycle. If another memory reference is directed to the active controller, the START level is ignored until the core cycle ends.

ACKN and DATA VALID signals are asserted on the bus for one SBus clock period. The ACKN and the first DATA VALID are generated by the MB20 on the starting phase. The second DATA VALID can be asserted by the MBox on either phase to end SBus operation.

2.1.8 Special Data Modes

If a memory reference is made to an MB20 controller set in loop-around mode (Subsection 1.2.9) or to a controller that has detected an address parity error (Subsection 1.2.8), normal SBus dialogue takes place but read/write currents are inhibited in the selected SMs. This means that write data is transferred normally from the MBox to the SM data registers but it is not written in core. The reason for this is to prevent the wrong core location from being overwritten when an address parity error occurs. Also, it allows data to be stored in the MB20, independent of core selection and read/write electronics, when in a loop-around mode.

When an SBus write operation is followed by an SBus read operation in a loop-around mode (read/write currents still inhibited), the normal clearing of the SM data registers is prevented at the beginning of the core cycle and the data stored during the previous write operation is passed back to the MBox in the usual fashion. Because the data is not cycled through core, diagnostic programmers can use this mode of operation to isolate failures in the data path.

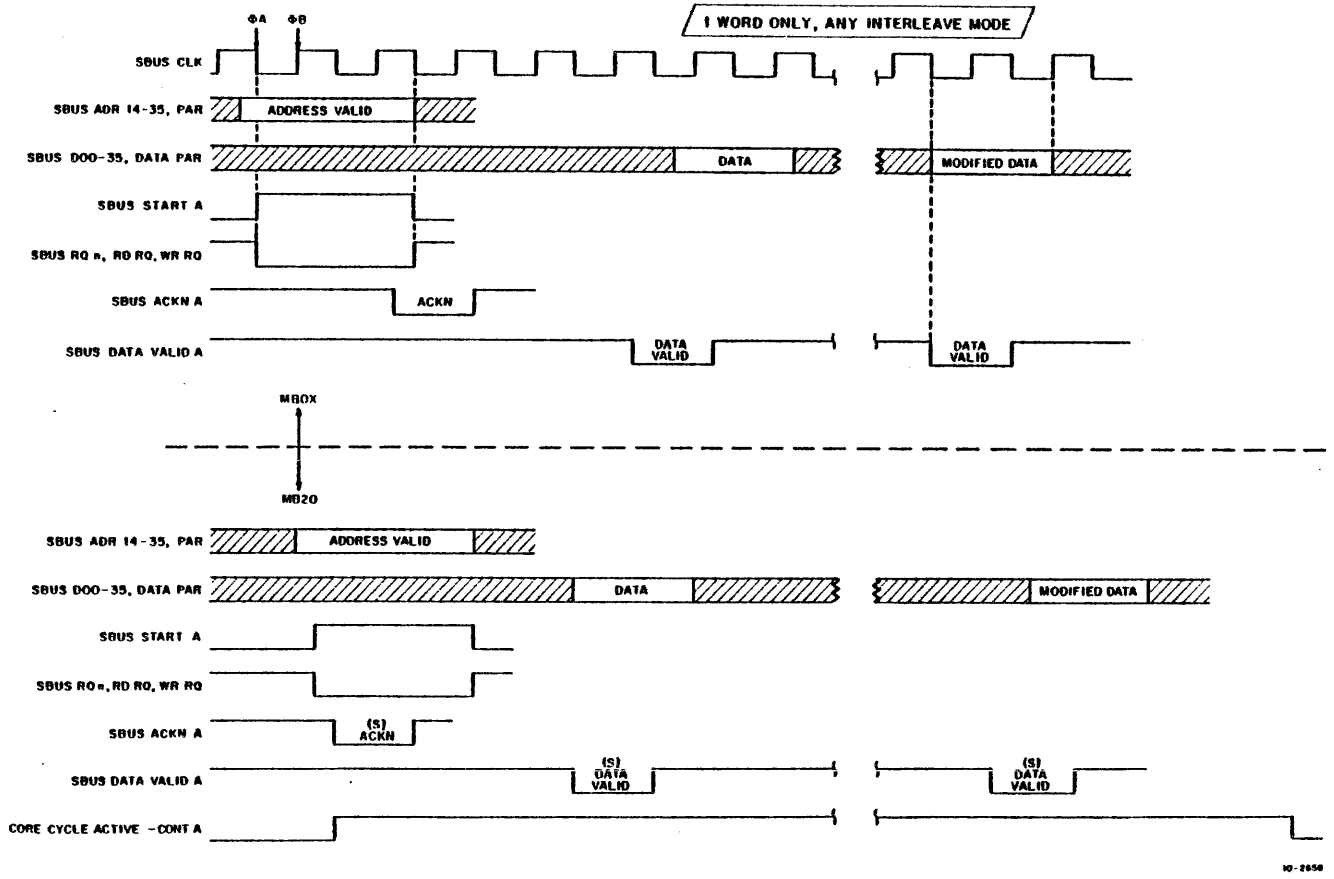


Figure 2-8 SBus RMW Timing Diagram (MB20)

When an address parity error has been detected during a read operation and loop-around mode is not set, the SM data registers normally clear at the beginning of the core cycle and (with no read/write currents) 0s are passed back to the MBox. Because the data parity bit is also 0, the MBox detects a data parity error for this case. In loop-around mode, an address parity error has no effect. The normal clearing of the data registers is inhibited during the read operation and the previously written data is returned to the MBox even though an address parity error occurred.

2.2 MEMORY ADDRESSING

A discussion of MB20 memory addressing for each interleave mode follows. Quad-word distribution and other details are illustrated in Figures 2-9, 2-10, and 2-11.

2.2.1 Four-Way Interleave Mode

In 4-way interleave mode (Subsection 2.1.4), the two controllers on either SBus 0 or 1 operate in parallel. Either controller in a pair can be designated as even, the other as odd. The even controller handles words 0 and 2 in the quad-word and the odd controller handles words 1 and 3.

The left-half of Figure 2-9 shows word distribution among the SMs connected to a controller pair in 4-way interleave mode. Each word in the quad-word is contained in a different SM, with SM0 and SM1 on both controllers being selected when SBUS ADR 18 = 0. As shown, SM0 and SM1 on the even controller contain words 0 and 2; on the odd controller, they contain words 1 and 3. As addresses increase in value and SBUS ADR 18 = 1, the same basic word pattern prevails, but words are contained in SM2 and SM3. ADR 18 and the RQ lines are hard-wire decoded by the controller to effect this word distribution; the decoding is summarized in tabular form in Figure 2-9. Also shown is how the ADR lines are gated to select a specific core location.

For any one memory reference in 4-way interleave mode, the ADR lines address the same core location in all selected SMs. To select all possible locations, 15 bits of address (ADR 19–33) are used as SMs have a 32K word capacity. (ADR 34 and 35 are not part of the core address because they are encoded in the RQ lines and used for SM selection.) In directing the ADR lines from the SBus to the address decoders in each SM, the controller gates out the two most significant bits of SBus core address in place of ADR 34 and 35. Thus, ADR 19 and 20 become the two least significant bits of actual core address as seen by the SMs. All other SBus address bits are gated to the corresponding decoder inputs in the SMs.

If a memory reference is made with all SBus core address bits equal to 0, location 0 will be accessed in all selected SMs. If the MBox then references consecutive quad-words from this point in 4-way interleave mode, the core addresses gated to the SMs increase in value but have the same two least significant bits (00). This is because of the bit swapping described in the preceding paragraphs. Thus, every fourth location is selected in an SM. As the SBus address increments, all locations ending in 00 are selected, then all locations ending in 01, then 10, and finally 11, after which all SM locations have been selected.

Figure 2-9 shows an example of how address boundary registers could be set in the controllers when in 4-way interleave mode. As required, both controllers in a pair have the same boundary settings. The upper and lower limits are those for a fully configured system with four SMs per controller.

With eight SMs operating on a bus, it has been shown that only one set of four SMs are referenced in any one SBus operation in 4-way interleave mode. These are either SM0 and SM1 or SM2 and SM3 on both controllers. Consequently, if an SM fails, one of the two sets of four SMs may continue to be operated if the address boundary registers are changed so that only the operational set is selected. A disadvantage to this procedure is that three operating SMs are addressed out of the system along with the bad one. This greatly reduces system memory capacity but 4-way interleaving is maintained. As an example of the above, refer to Figure 2-9 and assume SM3 failed on one of the controllers on SBus 1. This precludes the use of SM2 and SM3 on both controllers and they are deselected by changing the upper address boundary registers from 0111 to 0111. The address space of the controller pair is now defined so that it will respond to addresses for which SBUS ADR 18 always equals 0 and only SM0 and SM1 will be selected.

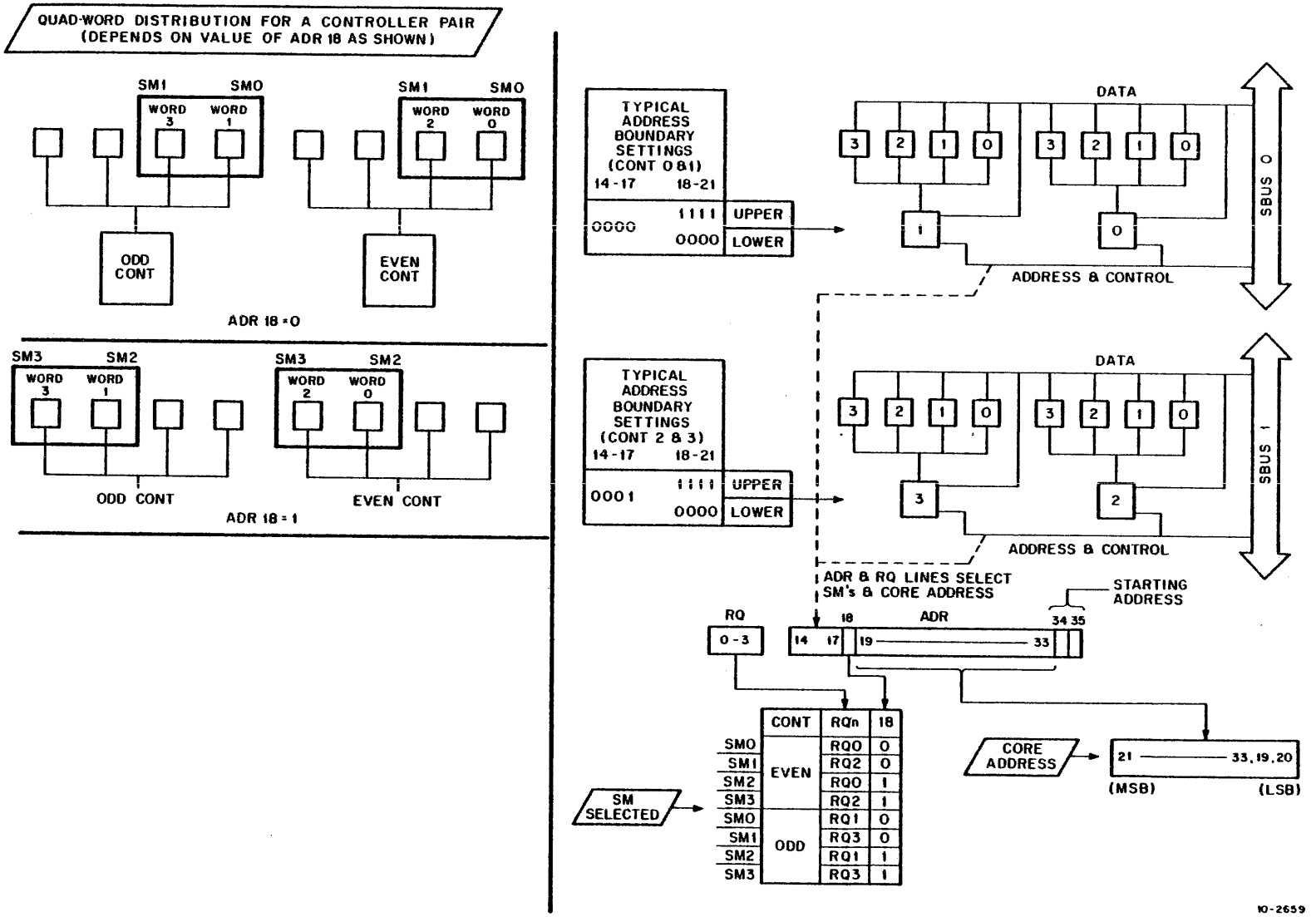


Figure 2-9 Memory Selection, 4-Way Interleave Mode

2.2.2 Two-Way Interleave Mode

In 2-way interleave mode (as in 4-way), two controllers on either SBus 0 or 1 operate in parallel and one controller on a bus is designated as even, the other as odd. Quad-word distribution within the SMs differs, however. As shown in Figure 2-10, the four words are distributed among two SMs, not four. Core locations for words 0 and 2 are selected from an SM on the even controller and the locations for words 1 and 3 are selected from the corresponding SM on the odd controller. The particular SMs selected depends on the values of SBUS ADR 18 and 19. For example, the quad-word is selected from SM0 on each controller when these address bits equal 00. SM1, SM2, and SM3 are selected in a similar fashion when the address bits equal 01, 10, and 11. How a controller decodes the RQ and ADR lines to select an SM is shown in Figure 2-10. Also shown is the gating of the ADR lines to select core locations.

As for 4-way interleave mode, the most significant bits of core address gated by the controller to the SMs are ADR 21–33. ADR 35 is not gated directly as part of the SM core address since its value is implicit in RQ0 and RQ2 (35=0) or RQ1 and RQ3 (35=1) and these levels are hard-wire decoded by the controller in SM selection. Instead, ADR 20 is gated (in place of ADR 35) to provide the least significant bit of core address. The core address bit corresponding to ADR 34 is generated by the controller as follows.

Two different core locations must be selected in the same SM in 2-way interleave mode and (to select one of them) the address bit corresponding to ADR 34 is set equal to ADR 34 when a controller is selecting the starting address (S); it is set equal to the updated value of ADR 34 (updated by controller logic – Subsection 2.3.6) when selecting the updated or modified starting address (S'). The modified starting address is the first address accessed in the second core cycle interval. To generate the address bit when a controller is selecting a word address other than S or S', ADR 34 cannot be used directly. Instead, the address bit has to be calculated depending on which SM locations (two possible) have been requested, and if both, which is to be addressed in the current core cycle. The calculation is based on the stored word requests (RQn, n = 0–3).

The core address generated for words 0 and 1 is the same in both storage modules. This is also true for words 2 and 3. Also, if words 0 and 1 have address X, then words 2 and 3 have address X + 2. If successive quad-words are accessed starting with an SBus core address equal to 0, locations 0 (words 0 and 1) and 2 (words 2 and 3) in the selected SMs are addressed first, then locations 4 and 6, followed by increasing even addresses for each quad-word until all even locations in the first set of SMs have been selected. At this point, the most significant bit of SBus core address increments to 1. Because of the bit swapping described previously, odd-numbered locations are now selected in the same set of SMs starting with locations 1 (words 0 and 1), and 3 (words 2 and 3). Successive quad-word references then address all the odd-numbered locations. With every SM location having been selected, the addressing sequence repeats as the incrementing SBus address selects locations 0 and 2 in another set of SMs.

Figure 2-10 shows how address boundary registers are set up in 2-way interleave mode. Note that in the example given, the settings are identical to those for the same configuration in 4-way interleave mode (Figure 2-9). Any valid 4-way interleaved configuration may be operated in 2-way interleave mode without having to change the address boundaries.

It has been shown that the SBus address in 2-way interleave mode selects the quad-word from corresponding SMs on the interleaved controllers. This should be kept in mind when a system is reconfigured to operate in 2-way interleave mode after an SM failure takes place. As an example of this, consider a fully configured system operating in 4-way interleave mode and assume SM3 fails on one controller on SBus 0. The bad SM must be deselected by changing the address boundaries for the controllers on SBus 0. If the upper limit was 1111, as in Figures 2-8 and 2-9, it would be changed to 1011. This defines an address space whereby ADR 18 and 19 cannot have a value of 11 (only 00, 01, and 10) and the inoperative storage module (SM3) can never be selected. However, due to 2-way interleave mode addressing, this also deselects SM3 on the other controller and an operative SM is addressed out of the system, further reducing storage capacity. If capacity is a major factor, switching to 2-way interleave mode is still a better alternative to remaining in 4-way interleave mode after an SM failure. This requires deselecting three operative SMs as explained in Subsection 2.2.1.

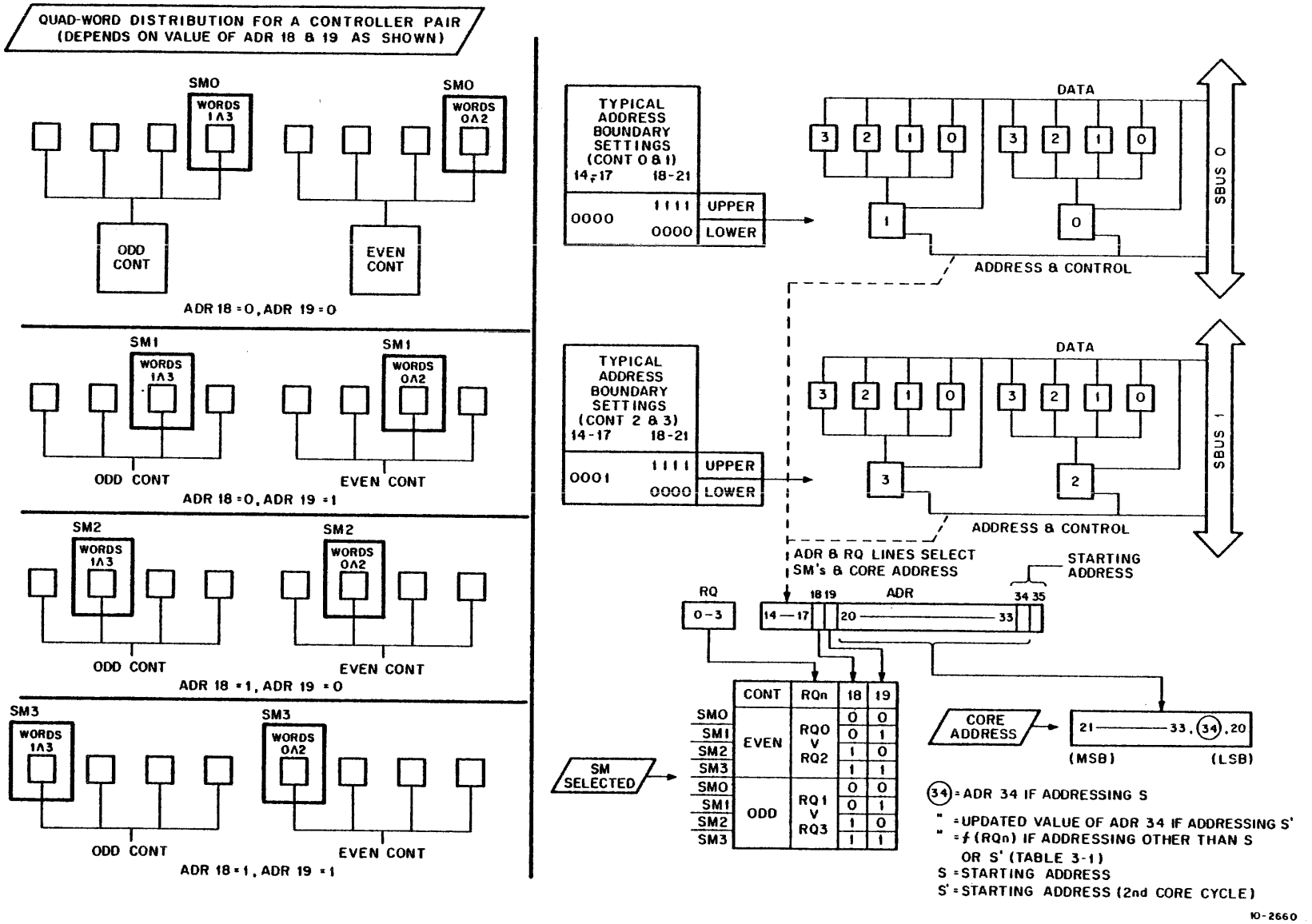


Figure 2-10 Memory Selection, 2-Way Interleave Mode

2.2.3 No-Interleave Mode

In no-interleave mode, only one controller in a system is selected by an SBus address. Furthermore, the quad-word is contained in only one SM on a controller with words being distributed as shown in Figure 2-11. The SM selected depends on SBus address lines ADR 19 and 20. The binary value of the address bits corresponds to the SM selected; that is, 00 picks SM0, 01 picks SM1, etc.

As for any interleave mode, ADR 21–33 are gated directly to the corresponding SM address decoder inputs to provide all but the two least significant bits of core address. The two least significant bits are equal to ADR 34 and 35 for the first core cycle, when the controller is accessing the starting address, and are equal to the updated values of ADR 34 and 35, the modified starting address (Subsection 2.3.6), for core cycles following the first. The address bits are modified for each core cycle depending on the stored word requests (RQn).

A quad-word occupies four consecutive SM core locations in no-interleave mode. If consecutive quad-words are referenced starting with an SBus core address of 0s, locations 0–3 in the selected SM are addressed first, followed by locations 4–7, with the core addresses increasing in value as the SBus address increases in value. When all locations in one SM have been addressed, another SM is selected by the incrementing SBus address and the sequence repeats starting at locations 0–3.

Figure 2-11 shows typical address boundary register settings for a fully configured system. Unlike 4-way and 2-way interleave modes of operation, no-interleave mode requires that each controller (not a pair) be assigned an exclusive portion of the available address space.

If an SM fails, operation may continue in (or be switched to) no-interleave mode without operative SMs being addressed out of the system when the bad SM is deselected. This is because words in a quad-word are not selected from two or four SMs as in interleaved operation. To show how the address boundary registers are changed to deselect an SM in no-interleave mode, refer to Figure 2-11 and assume that SM2 fails on controller 0. Since ADR 19 and 20 effect SM selection, the boundaries are changed so that these address bits cannot have a value of 10 (selects SM2) and still be within the address space assigned to controller 0. Changing the limits from "0000 and 0111" to "0110 and 1011" accomplishes this. It can be seen, however, that the new boundaries for controller 0 infringe on the address space assigned previously to controller 1. Consequently, for this particular reconfiguration, the boundaries in controller 1 (and controllers 2 and 3) must also be changed.

2.2.4 Rules for Memory System Configuration

The following rules apply to MB20 addressing and system configuration:

1. An interleaved controller pair must be connected to the same SBus, either 0 or 1.

2. Parallel operation in 4-way and 2-way interleave modes requires that two controllers be included in the same block of assigned address space (address boundary registers set to the same values). In no-interleave mode, each controller is assigned an exclusive portion of the available address space.

3. Storage modules must occupy contiguous blocks of core within the controller's assigned address space.

4. Each controller in an interleaved pair must have the same number of storage modules. In 4-way interleave mode, both must have either SM0 and SM1, SM2 and SM3, or SM0–3. In 2-way interleave mode, both must have corresponding storage modules; SM0 on both, SM1 on both, etc. A controller can have any number of storage modules (1–4) in no-interleave mode.

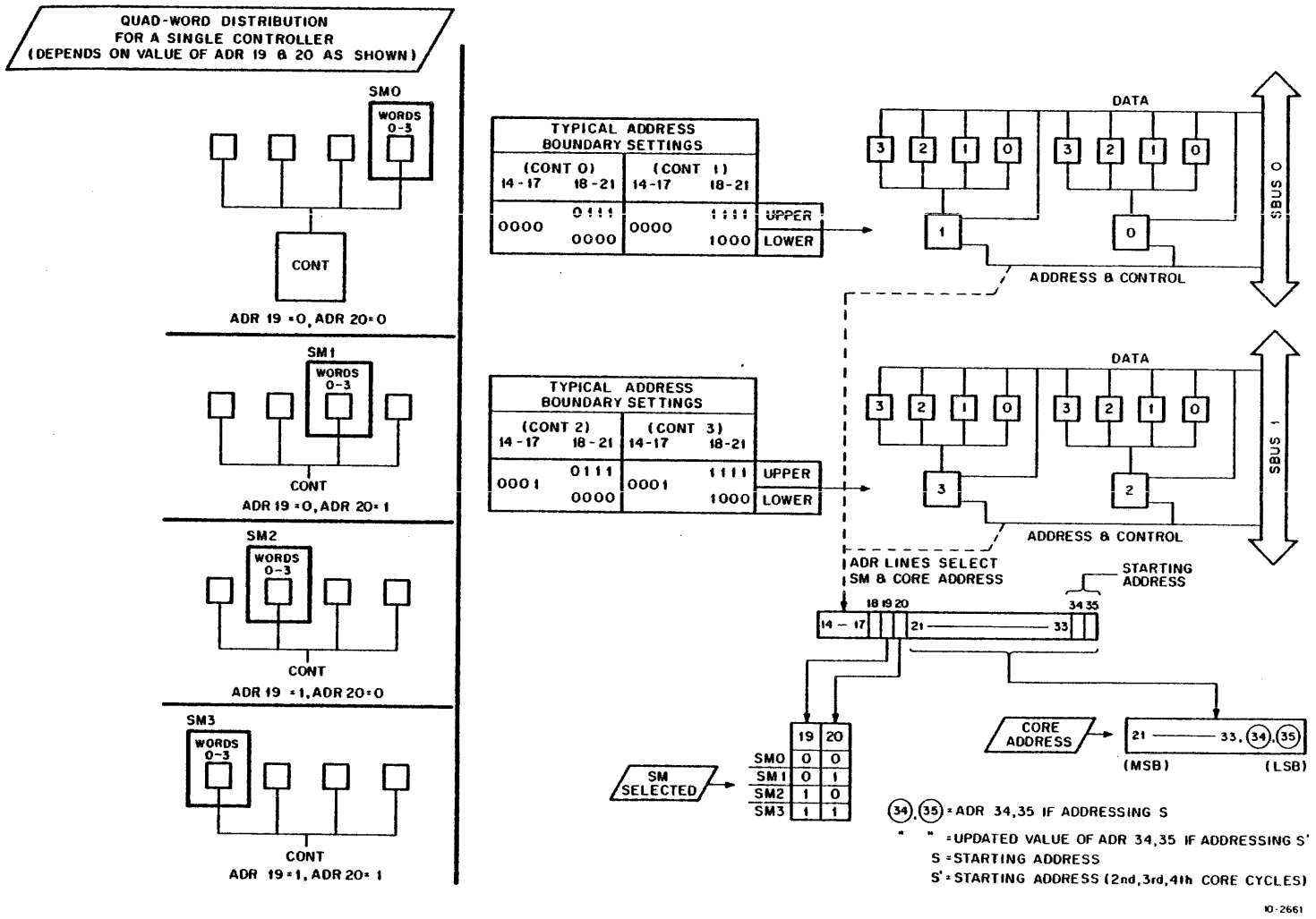


Figure 2-11 Memory Selection, No-Interleave Mode

2.3 BASIC CONTROLLER OPERATION

The MB20 controller consists of a control module and a timing module. A description of basic operation follows, with emphasis on sequence of operation during a memory reference. Major control and timing signals are discussed and basic logic flow is shown in Figure 2-12. A more detailed description of the controller is given in Section 3.

2.3.1 Start-Up

SBUS START is generated in the MBox coincident with one phase of SBUS CLK. A controller will start on the next clock of the same phase provided all of the following conditions are true. Otherwise, the START level is ignored.

1. The SBus address lines must have a value within the limits defined by the controller's address boundary registers. This condition is termed ADR MATCH in Figure 2-12.

2. The controller must be enabled to access the words requested. The logic level RQ EN will be asserted if one or more word requests are enabled.

3. The controller must not be busy from a previous memory reference. Also, the other controller on the same SBus must not be busy if the operation is interleaved (both odd and even word requests).

The first two conditions for start-up are discussed in Subsection 1.2.7. The busy condition, diagrammed in Figure 2-12, allows a memory reference directed to one inactive controller to start even though the other controller on the bus is still busy from the previous reference. This increases word transfer rates over the SBus when successive operations reference first one controller and then the other. An example of this is an operation requesting word 1 in the quad-word followed by an operation requesting word 0. The even controller is allowed to start while the odd controller is still busy with the request for word 1.

The busy condition also specifies that if both controllers are to be active in the same memory reference (odd and even word requests in 4-way and 2-way interleave modes), both controllers must be inactive before the operation can begin. In other words, a controller pair must start together in an interleaved operation. This is necessary because controllers are synchronized when operating in parallel with address counters and other logic stepping together as described in Subsection 2.3.2.

With reference to the flow diagram (Figure 2-12), the following occurs when a controller is started.

1. A core cycle is initiated in the SMs selected by the ADR and RQ lines. (Memory addressing is discussed in Subsection 2.2.) The core cycle is controlled by a succession of control and timing signals generated by the controller's timing module. The signals for the core read cycle are generated first. If an address parity error occurs or if loop-around mode is set, the timing logic sequences but timing signals are not gated to the selected SMs. This is to inhibit SM read/write currents for the reasons given in Subsection 2.1.8. Read/write currents are also inhibited for a special case occurring in 2-way interleave mode when three words are requested having addresses S, S + 2, S + 3. The reason for this is that address S + 3 must be accessed during the second core cycle interval (Figure 2-5), but the controller handling this one address must start and step with the other controller during the first core cycle to maintain synchronization during the interleaved operation. To maintain synchronization, it inhibits read/write currents and steps through a dummy cycle while the other controller handles address S.

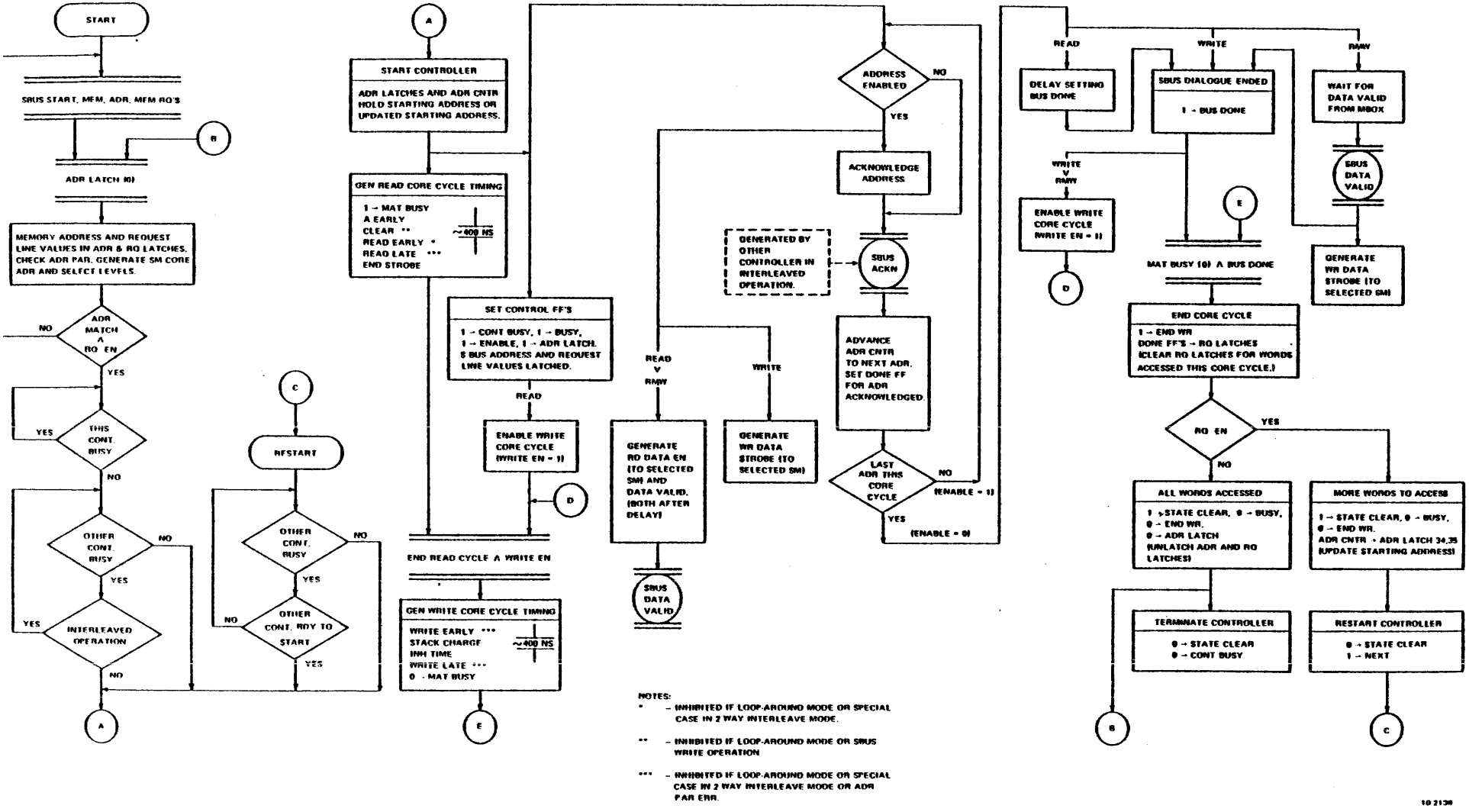


Figure 2-12 MB20 Controller, Sequence of Operation

2. The starting address is acknowledged if the controller is enabled to access the first word. Generation of ACKN is discussed in Subsection 2.3.2.

3. Flip-flops are set to define the controller as busy (CONT BUSY), to indicate a core cycle is active (BUSY), and to allow generation of ACKN signals (ENABLE). A control flip-flop (ADR LATCH) is also set to latch the address and request latch circuits. The latch circuits store the SBus address, the word requests, and the read/write requests.

2.3.2 Address Acknowledge

Logic flow for the generation of ACKN is shown in Figure 2-12. When a controller is started, the starting address held in the two low order address latches is loaded in an address counter. The starting address is also compared with the previously set RQ EN levels and ACKN is immediately transmitted on the SBus if the address is enabled. If not enabled, the operation must be interleaved and the address will be enabled in the other controller on the bus. In either case, the starting address is acknowledged by one of the controllers.

When SBUS ACKN is generated, it is received by both controllers on the bus as well as the MBox. An active controller uses the signal to advance the address counter to the address of the next word to be accessed depending on the stored word requests. The counter logic increments the address in ascending order, modulo 4, and determines the order in which the words requested at the beginning of the operation are accessed. With two controllers active in an interleaved operation, both address counters increment together and step through the same addresses as each ACKN is transmitted.

As occurred for the starting address, the incremented address is compared with the RQ EN levels and ACKN is generated for each enabled address. ACKN is transmitted and the address counter advances until the control logic determines that all addresses have been acknowledged for the current core cycle interval. The ENABLE flip-flop is cleared at this time preventing ACKN from being generated. This causes the counter to stop incrementing and the ACKN control logic ceases to cycle. With reference to Figure 2-5, it can be seen that the address counters in both controllers would increment four times during the one core cycle interval when four addresses are requested and acknowledged in 4-way interleave mode. In contrast, a one-word request results in only the starting address being acknowledged and the counter advancing just one time during the core cycle. At the end of a core cycle, the address counter is left pointing to the next address to be accessed if another core cycle is to follow.

Four controller flip-flops (DONE 0–3, corresponding to words 0–3) are used to keep track of which words have been acknowledged during an operation. The flip-flops are used at the end of a core cycle to clear the latch circuits that store the corresponding word requests. The controller will terminate if all enabled word requests are cleared at this time. If not, the controller restarts and another core cycle takes place. Termination and restart are discussed in Subsection 2.3.6.

2.3.3 SBus Write

As each address is acknowledged during an SBus write operation, a data strobe signal is generated by the control module and directed to the selected SM. The signal gates the word to be written off the SBus data lines and into the SM data register. There are four data strobes, Bn CLK (n = 0–3), one for each storage module (SM 0–3).

When all addresses have been acknowledged and ENABLE goes to 0, SBus dialogue has ended for the write operation (for the current core cycle) and the controller sets BUS DONE. Because all data words have been strobed into SM data registers, BUS DONE is allowed to assert WRITE EN. This signal is directed to the timing module to cause the timing signals necessary for the write portion of the core cycle to take place provided the read portion has ended. The read timing signals are initiated at start-up (Subsection 2.3.2). When the write timing signals have been generated to end the core cycle, the timing module busy signal goes false which sets END WR in the control module. The controller will then terminate or start another core cycle as explained in Subsection 2.3.6.

2.3.4 SBus Read

Delay logic is activated as each address is acknowledged during an SBus read operation. After the fixed delay, during which time the data word is read from core into the selected SM data register, a read data enable signal is generated by the control module to gate the data register contents onto the SBus data lines. Also, a DATA VALID is generated at the same time to signal the MBox that data is on the lines. There are four read data enable signals, Bn DO EN (n = 0–3), one corresponding to each storage module (SM 0–3).

Because data is restored in core for the SBus read operation, WRITE EN is set at the beginning of the operation when timing for the read portion of the core cycle is started. With no new data to be written in core, this allows core write cycle timing to immediately follow core read cycle timing. The data just read is then written back in the selected core location.

BUS DONE is set after all addresses have been acknowledged for the current core cycle (ENABLE cleared) and just prior to transmission of the last DATA VALID on the SBus. With SBus dialogue near completion, the END WR flip-flop then sets (when the core write cycle ends or if it has already ended) and the controller terminates or restarts as discussed in Subsection 2.3.6.

2.3.5 SBus Read-Modify-Write (RMW)

Only one word is requested in an SBus RMW operation. Operation is similar to the SBus read operation in that a read data enable signal and a DATA VALID signal are generated after a fixed delay when the word address is acknowledged. WRITE EN is not set at start-up, however, and BUS DONE is not set when the first DATA VALID is generated. This is because core write cycle timing cannot be enabled and SBus dialogue cannot be flagged as having ended until the MBox modifies the data word read from core and transmits a second DATA VALID signal signaling that the modified data is on the data lines. When this occurs, BUS DONE sets and asserts WRITE EN. One of the four write data strobes, Bn CLK (n = 0–3), is also generated to load the data word in the selected SM's data register. When write core cycle timing signals have ended, END WR is set and the controller terminates as described in Subsection 2.3.6.

2.3.6 Termination And Restart

As discussed previously, the END WR flip-flop sets when the SM core cycle and associated SBus dialogue have ended. END WR clears the two least significant bits of stored SBus address, ADR LATCH 34 and 35. It also clears one or more of the stored word requests in latch circuits RQ 0–3, depending on which words were accessed during the core cycle interval. It does this by gating the four flip-flop outputs DONE 0–3 (one of which is set as each address is acknowledged during the core cycle – Subsection 2.3.2) directly to the latch circuits. An asserted DONE flip-flop clears the corresponding latch.

If all the stored word requests enabled by the controller are cleared when END WR sets, all words have been accessed and the controller can terminate. If enabled word requests are still held in the latches, more words are to be accessed and another core cycle must be started. The state of RQ EN determines which of the above conditions are true. RQ EN is asserted and required at start-up (Subsection 2.3.1) when the word requests are first loaded in the latches and compared to the preset enable levels. It remains asserted after the latches are cleared at the end of a core cycle, as long as one stored word request compares with the corresponding enable level. As a result, it will equal 1 when more words are to be accessed and it will go to 0 at the end of the last core cycle. The number of core cycles required for a memory reference range from one (1-word request) to four (4-word request in no-interleave mode).

With END WR asserted at the end of a core cycle, STATE CLEAR goes true clearing BUS DONE and BUSY. If RQ EN equals ONE, indicating another core cycle is to follow, STATE CLEAR also gates the contents of the address counter into ADR LATCH 34 and 35. The address counter holds the address of the next word to be accessed. NEXT is then asserted and the controller will restart, provided

the other controller on the bus is not busy. If the other controller is busy, restart is delayed until it goes inactive or until it also generates a NEXT signal. This ensures that both controllers will restart together if both are initiating another core cycle.

If RQ EN equals 0 when BUSY clears at the end of the core cycle, the controller terminates by first unlatching all address and request latch circuits and then by clearing CONT BUSY. It is now ready to respond to another MBox memory reference provided the conditions for start-up are met (Subsection 2.3.1).

2.3.7 Core Cycle Timing

SM timing and control signals generated by the timing module are listed below. As shown in Figure 2-12, core read cycle timing is initiated at start-up and core write cycle timing starts when enabled by the controller and the core read cycle timing has ended.

A EARLY	Activates core read select circuits and turns on Y read select current.
CLEAR	Clears data register preparatory to data being direct-set in register when read from core.
RD RQ	Asserted for SBus read or RMW operation. Enables generation of read strobe so that data read from core will be loaded in data register.
RD EARLY	Times read currents. Trailing edge turns off X and Y select current.
RD LATE	Turns on X read select currents. Trailing edge deactivates read select circuits.
END STROBE	Trailing edge ends sense amplifier strobe signal during core read.
WR EARLY	Core write select circuits activated during duration of this signal.
STK CHARGE	Complement of WRITE EARLY. Enables stack charge circuit which reverse-biases unselected diodes in select matrix.
INH TIME	Turns on inhibit currents.
WR LATE	Turns on X and Y write select currents.

2.4 BASIC STORAGE MODULE OPERATION

An MB20 Storage module is divided into two sections, each section providing 32K 19-bit words of core memory. Only 18 bits of storage are utilized in one section, the full 19 bits in the other. The first section stores bits 00–17 of the data word and the second section stores bits 18–35 plus the parity bit. The two sections together form an SM capable of storing 32K 37-bit words. Each section consists of three modules (total of six for SM). These include a stack module, an X-Y driver module, and a sense/inhibit module.

The SM core cycle consists of a core read operation followed by a core write operation. The core cycle is sequenced by the succession of control and timing signals generated by the timing module. A description of basic SM organization and operation follows. Sequence of operation is shown in Figure 2-19. A more detailed description of the storage module is given in Section 3.

2.4.1 Core Array

Each SM section contains a ferrite core memory consisting of 19 memory mats arranged in a planar configuration with each mat containing 32,768 ferrite cores arranged in a 256 × 128 array. A mat represents a single bit position in the data word and the planar configuration provides a total of 32,768 19-bit word locations. Each ferrite core can assume a stable magnetic state corresponding to either a binary 1 or 0. Even if power is removed from an SM, the core elements retain their magnetic state until changed by a memory reference after power is restored.

Each core is threaded by three wires to provide a means for core selection and switching. The three wires include two select lines (X and Y) and a sense/inhibit line. Figure 2-13 illustrates typical wiring for a small portion of one core mat. The X select lines pass through all cores in each horizontal row and Y select lines thread all cores in each vertical row. There are as many X and Y select lines as there are horizontal and vertical rows on a mat. Thus, in the 256 × 128 core array for the MB20, there are 256 X lines and 128 Y lines.

Each select line passes through all core mats in the memory. This is illustrated in Figure 2-14 for a smaller 16-word × 4-bit planar memory. Note that each of the X and Y lines pass through corresponding cores on all four mats. Wiring is similar for the MB20 where 19 mats are threaded by the X-Y lines.

The sense/inhibit lines are wound in parallel with the Y select lines (Figure 2-13). There are two lines per mat (bit position) with each threading half of the cores. The function of the sense/inhibit line and the select lines are discussed in Subsections 2.4.2 and 2.4.3.

2.4.2 Basic Core Write

As described in Subsection 2.4.1, X and Y select lines thread each ferrite core. A core will change its magnetic state if the resultant magnetic field caused by the currents through these lines is sufficient. This is illustrated in Figure 2-15 which shows a typical hysteresis loop for a ferrite core. The effect of select line current is as follows.

Assuming a core is initially in the 0 state, normal write current applied in either the X or Y select line causes a flux change in the core corresponding to a move from point 0 to point 1 on the hysteresis loop. The core remains in this state as long as the drive current continues to flow. The core returns to the 0 state where drive current ends. This is because the current in one select line is not great enough to change the core's magnetic state. The current in either an X or Y line is called a half-select current.

When X and Y lines are both driven, the reinforcing magnetic field caused by the coincident half-select write currents causes a flux change in the core corresponding to a move from the 0 state to point 2 on the loop. For this case, the direction of flux changes in the core, and when drive current ends, the core moves from point 2 to point 3 but remains in the 1 state. It is this action that allows the X-Y wiring arrangement to select a core location. One X and one Y line are driven during a memory reference and the coincident currents select one core on each mat. There are a total of 37 mats used in the SM and the 37 cores selected by the driven X and Y lines comprise the core location. Another combination of X and Y lines would select another 37 cores or another core location.

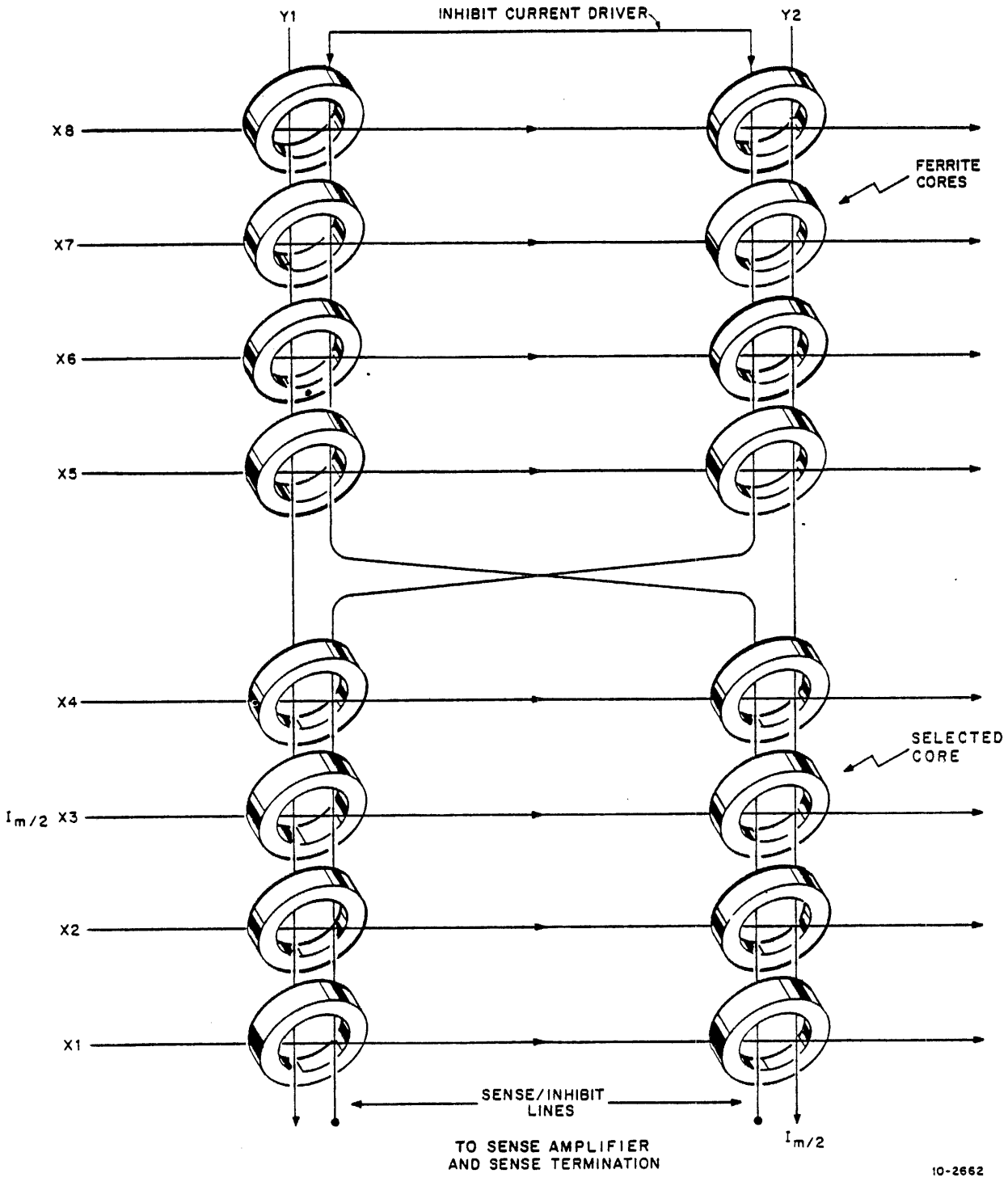


Figure 2-13 Three-Wire Memory Configuration

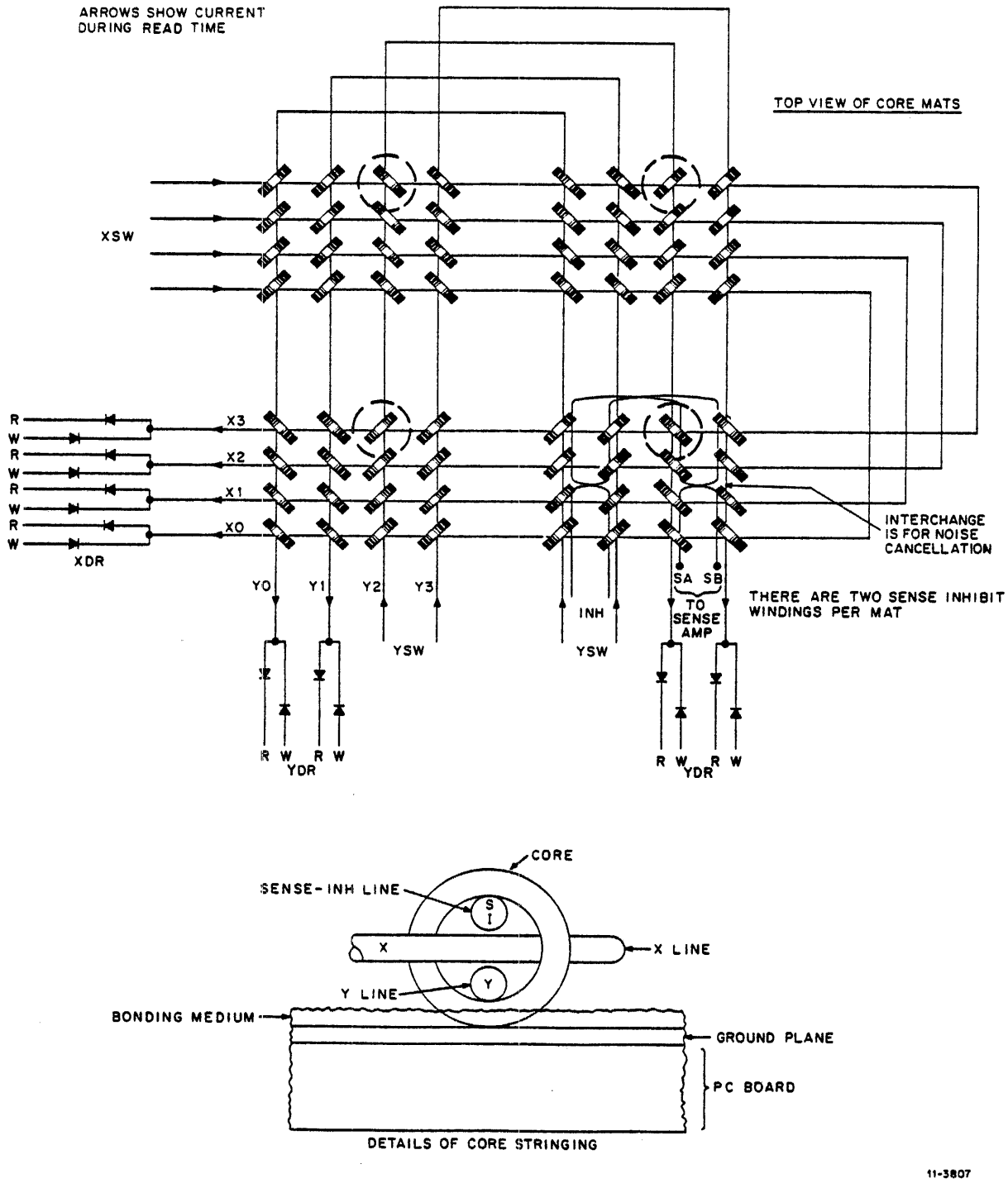


Figure 2-14 Core Select Wiring for a 3-Wire, 3-D, 16 Word by 4-Bit Memory

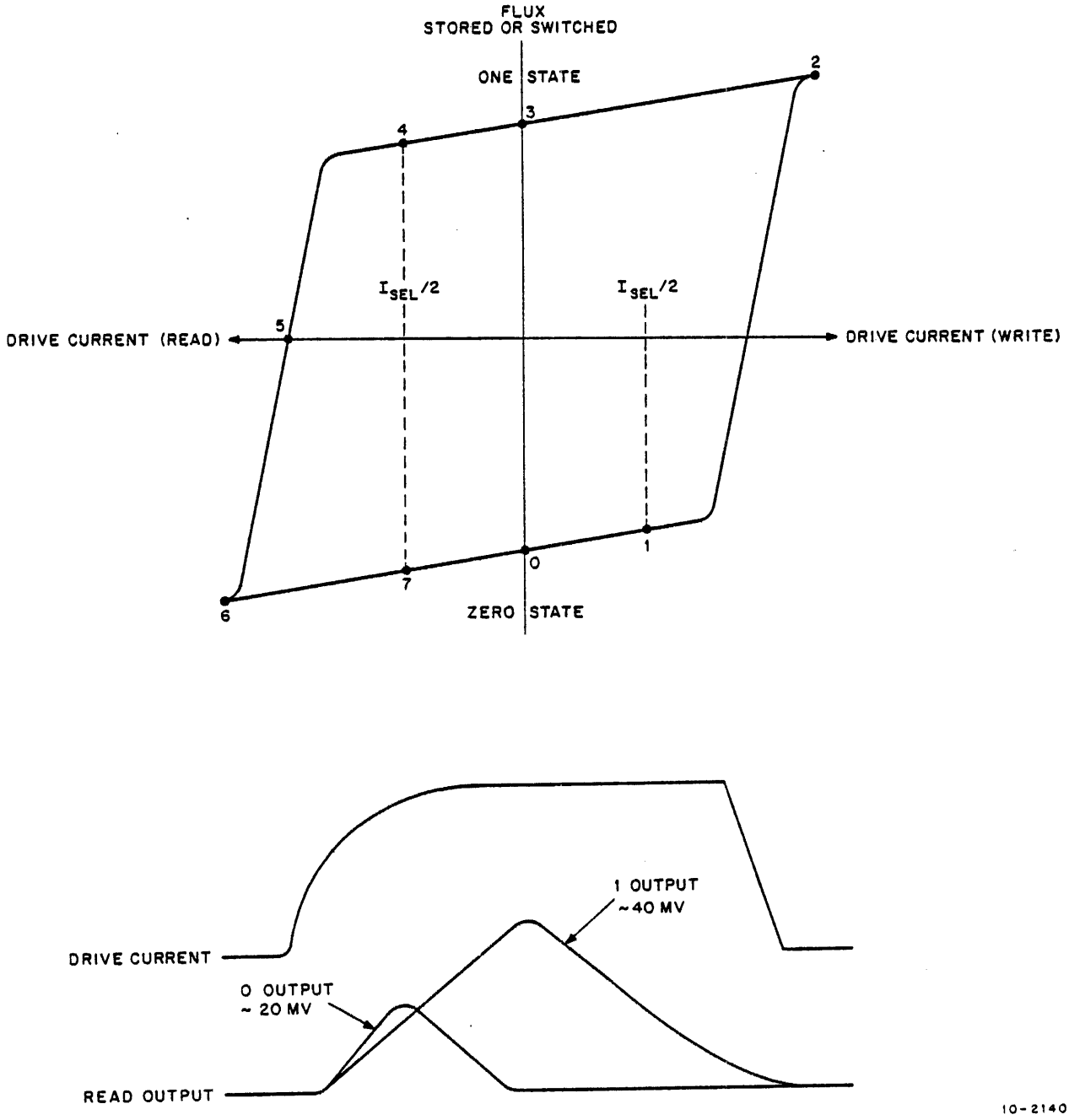


Figure 2-15 Hysteresis Loop and Read Outputs for a Ferrite Core

The operation described above causes a 1 to be written in a core. To write a 0 in a selected core, the sense/inhibit line (not driven when writing a 1) is driven in the opposite direction to the current in the select lines. The sense/inhibit line is wound in parallel with the Y line and the inhibit current cancels out the effect of the Y half-select current. This causes the same action as that previously described when only one select line is driven and the core does not change state. Assuming the core is initially in the 0 state at the beginning of the core write operation, it remains in the 0 state when drive currents end and a 0 is written in core. (The selected cores are initially in the 0 state as a result of the core read operation, which precedes the core write during the SM core cycle.) To summarize the core write operation:

1. Cores are initially in the 0 state.

2. Cores corresponding to bit positions in the word equaling 0 are inhibited and cores corresponding to bits equaling 1 are not inhibited.

3. X-Y currents select a core on each mat and 1s are written as the coincident currents cause the uninhibited cores to switch state. The inhibited cores remain in the 0 state.

2.4.3 Basic Core Read

Reading of a core in the 1 state is accomplished by passing full select current through the X and Y lines and then sensing the voltage induced in the sense/inhibit line (not driven during the read) as the core moves to the 0 state. Select currents are generated as in the core write operation but they are in the opposite direction.

With reference to Figure 2-15, a half-select read current causes a flux change in a core corresponding to a move along the hysteresis loop from point 3 to point 4. The small flux change induces a correspondingly small voltage in the sense/inhibit line. Upon removal of the half-select current, the core returns to the 1 state.

When a core is selected, the coincident X and Y select currents cause the core to switch states and move from the 1 state, through point 5, to point 6. A large voltage is induced in the sense/inhibit line in the region of point 5. (This is the area of maximum flux change.) When drive currents end, the core moves to point 0 and remains in the 0 state. This is a read-destroy operation, where the 1 content of the core has been sensed (read) and the core is left containing a 0. When reading, the output from the sense/inhibit winding is strobed at the point where the 1 output peaks.

If a core is originally in the 0 state and half-select current is applied, the core moves to point 7 and returns to the 0 state when the current is removed. If full-select current is applied for a time, the core moves to point 6 and returns to the 0 state. Both of these flux changes cause voltages to be induced in the sense/inhibit line but the voltages are small (2–3 mV) compared to the 1 output (40 mV). The difference in amplitude between the 1 and 0 outputs from a selected core provides the method for recovering data stored in core memory.

Although the ratio of 1 to 0 outputs for a single core is large, a problem arises during the core read operation, stemming from the fact that a sense/inhibit line passes through half the cores on the mat and a number of these cores (those threaded by the driven X and Y lines) are subjected to half-select currents. To prevent the cumulative effect of the small voltages induced by each core from masking the 1 output, the sense/inhibit line is wound through half of the cores that it threads in one direction and through the remaining half in the opposite direction. The method of winding (bow tie) is shown in Figure 2-14. It causes the half-select outputs from half of the cores to oppose and cancel the outputs from the other half. However, due to the differences in half-select outputs (some of the driven cores can be in the 0 state, others in the 1 state) and due to small differences in core characteristics, unwanted outputs never completely cancel and the resulting sense/inhibit line voltage, termed delta noise, can

approach a value of 20 mV. (The relative magnitude of 0 and 1 outputs is shown in Figure 2-15.) Delta noise comprises the 0 output and adds or subtracts from the 1 output. The sense/inhibit line output is strobed at or near its peak when delta noise has subsided and the ratio of 1 to 0 output amplitudes is greatest. The core read can be summarized as follows:

1. X-Y currents select a core on each mat.

2. Cores in the 1 state switch to the 0 state when selected and those already in the 0 state do not change state.

3. A 1 is read by sensing the large voltage induced in the sense/inhibit line for each mat when a selected core switches. Minimal voltage is induced when a core containing a 0 is selected.

4. All selected cores are left in the 0 state as a result of the read operation.

2.4.4 X-Y Selection

Figure 2-16 shows the basic X-Y select circuitry for one 19-bit section of an MB20 storage module. The 15 bits of core address are decoded (as shown) to select circuits in a driver/switch matrix so that one X line and one Y line are driven in the core array. When the timing and control signals from the controller enable the selected drivers and switches (and their current sources) current is switched through one of the 256 X lines and one of the 128 Y lines. The current path is through the current source (not shown in Figure 3-8) the selected driver, a stack diode, the X or Y line, and the selected switch. When the lines are driven, the coincident current at the intersection of the X and Y lines selects one of the 32,768 (256 × 128) cores on a mat. This set of 19 cores (one on each mat) corresponds to the 19-bit word being addressed in the stack.

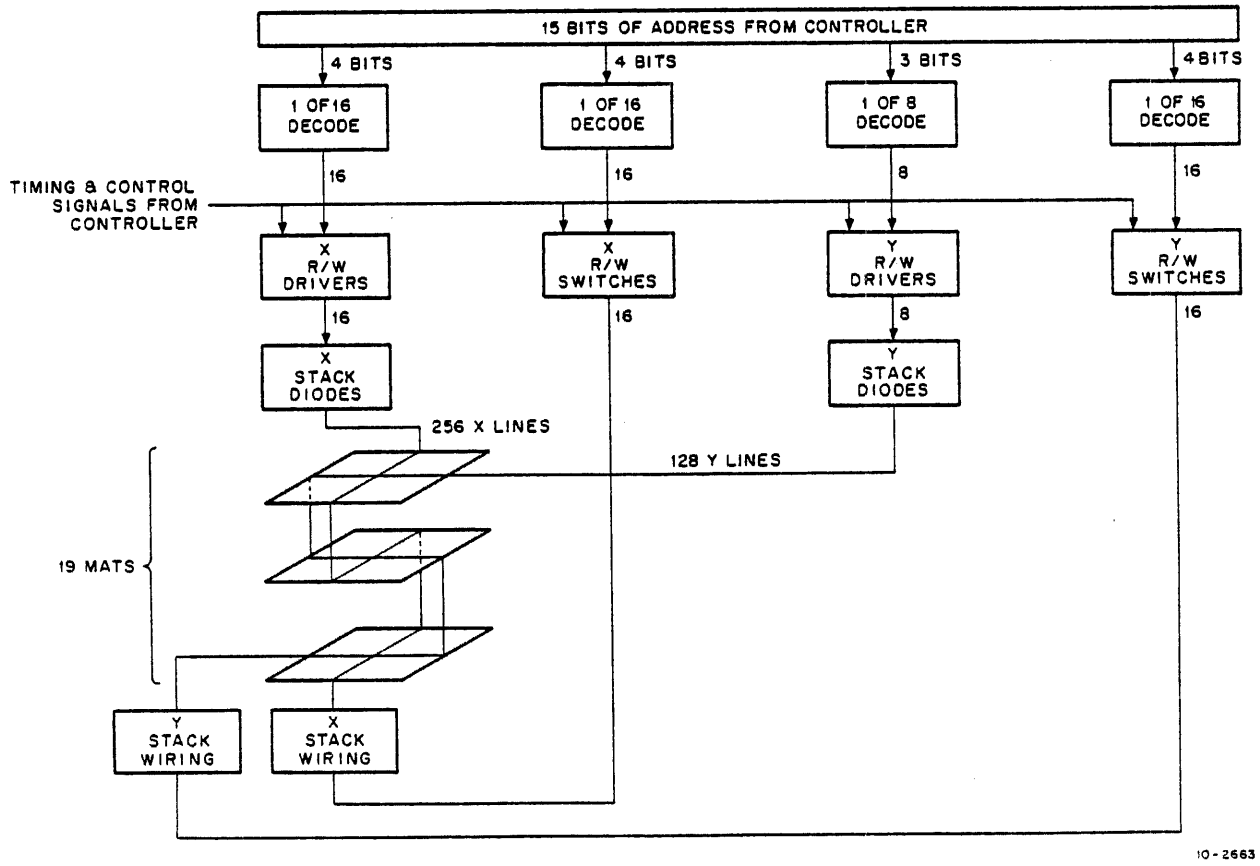


Figure 2-16 SM Word Select Circuits, Basic Block Diagram

The selected X-Y lines are switched twice during an SM core cycle. The first time, data is read from the selected stack address (Subsection 2.4.3). The same X-Y lines are then driven again, but in the opposite direction, to write data in the selected stack address (Subsection 2.4.2).

Figure 2-17 shows a portion of the X line selection matrix. Note that drivers and switches are differentiated by function, either read or write, and by polarity, either positive or negative. Read switches and write drivers are connected to current sources and are considered positive. Write switches and read drivers connect to ground and are considered negative. During a core cycle, the core address decoders select one of the R/W driver pairs and one of the R/W switch pairs in the X selection matrix. For the read operation (the first part of a core cycle) one X read switch and one X read driver are enabled to provide a current path from the X read current source through the switch, one of the X lines, a stack diode, and the driver to ground. A Y read switch and Y read driver are also selected in the Y selection matrix, providing a current path through one of the Y lines. When the selected drivers and switches are enabled, half-select currents flow in both the X and Y lines to select a core location. When switch, driver, and stack diode are conducting current, potentials are such that the remaining stack diodes in the selection matrix are reverse-biased. This isolates the active current path from the rest of the select circuitry and stack wiring.

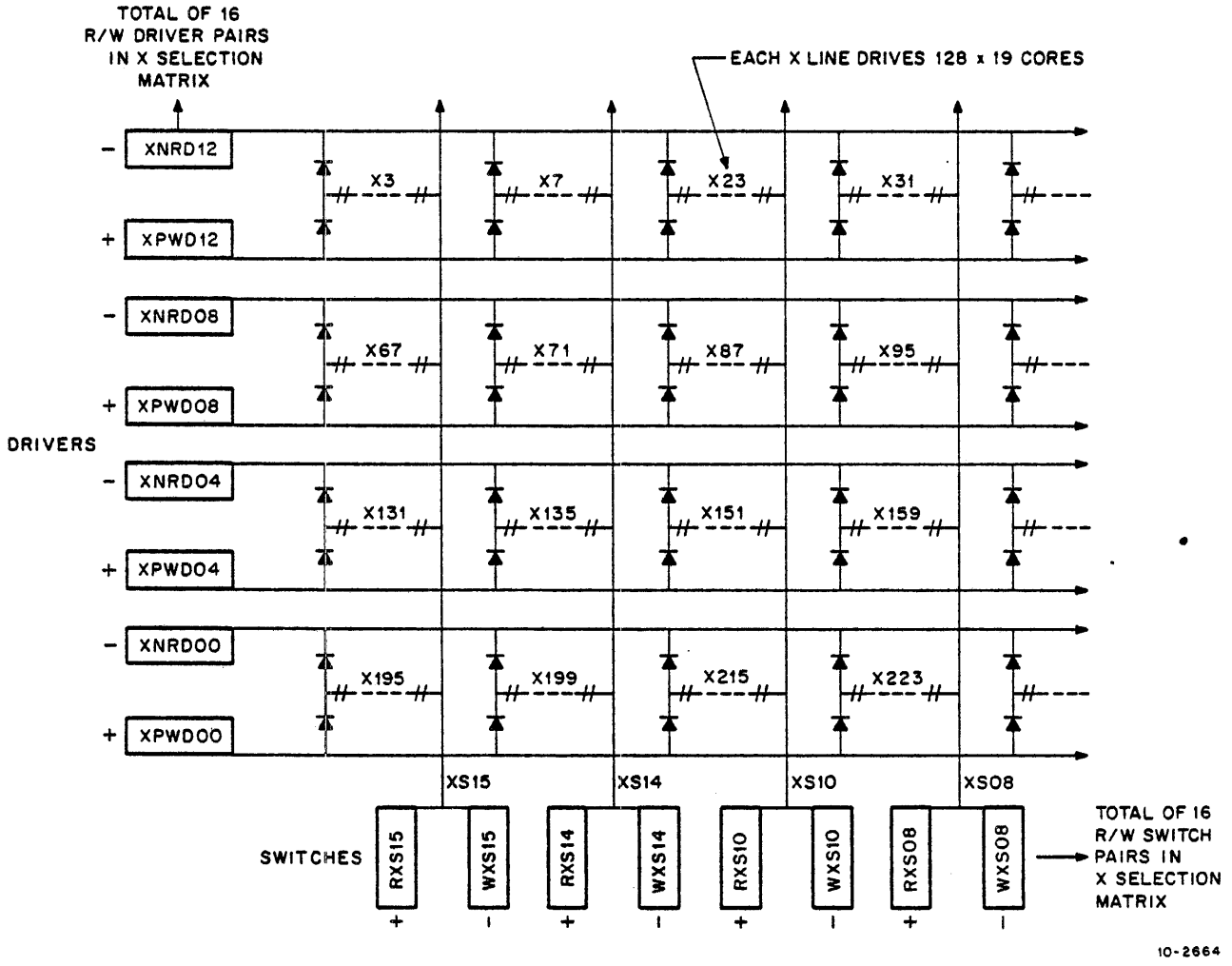


Figure 2-17 X-Line Selection

For the core write operation (the second part of an SM core cycle), the write drivers and the write switches are enabled in the selected driver/switch pairs. As for the read operation, a half-select current flows in the same X-Y lines but the current is in the opposite direction. (During the write, the current source output is gated through the driver and grounded at the switch.) Again the stack diodes isolate the active circuitry from the rest of the memory. The X-Y select circuits are discussed in more detail in Section 3.

2.4.5 Data Buffering and Sense/Inhibit Functions

The data register and associated sense/inhibit circuitry perform the following functions:

1. Sense amplifiers sense and strobe the read outputs on the sense/inhibit lines.

2. The data register stores the read data so that it may be transmitted to the MBox (SBus read or RMW) and restored in core (SBus read) during the core write cycle. The data register also accepts and stores new data from the MBox (SBus write or RMW) so that it can be deposited in core during the core write cycle.

3. Inhibit drivers drive the sense/inhibit lines to cause 0s in the new or restored data to be written in core.

Components in the read/write data path for a typical data bit (D11) are shown in Figure 2-18. Note that with two sense windings per mat with each threading half the cores, the data path is split; that is, two sense amplifiers and two inhibit drivers are associated with one data bit. The one sense amplifier selected (by SENSE STROBE 0 or 1) and the one inhibit driver selected (by INH 1 or 2) during a memory reference depend on core address bit 34.

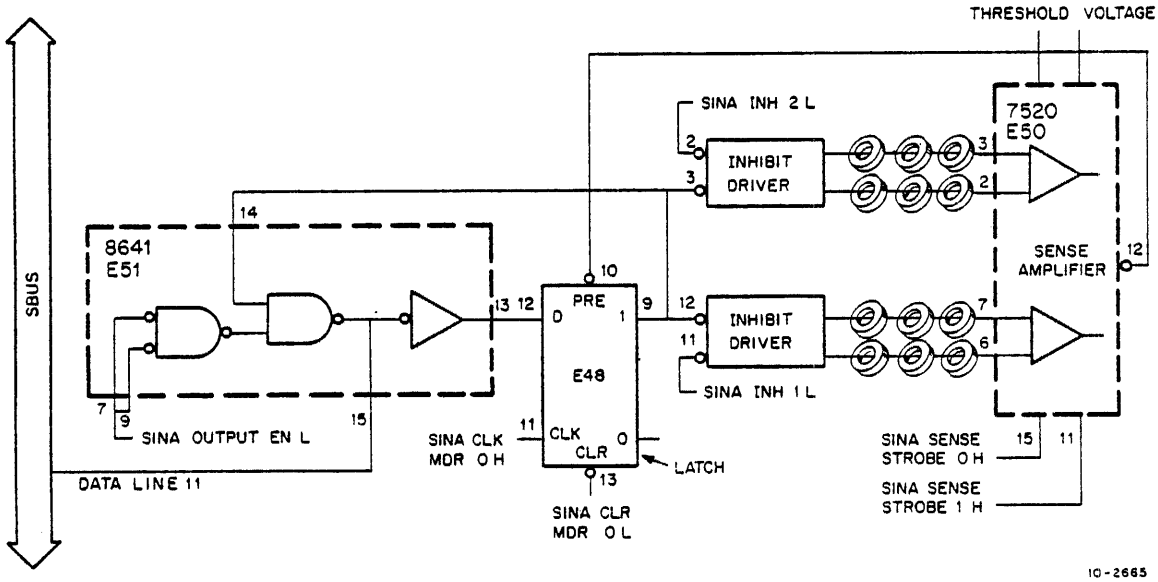


Figure 2-18 Interconnection of SBus, Data Register, Sense Amplifier, and Inhibit Driver

For the SBus read and RMW operations, CLR MDR is generated at the beginning of the core read cycle to clear the data register. A SENSE STROBE signal is generated next to gate the sense amplifier 1 outputs and direct-set the data register flip-flops. The strobe timing is critical and it is preset to optimum position by cutting jumpers on the SM driver module. Margining capability is provided, however, and the strobe can be moved to early or late positions under diagnostic program control. With the data register loaded, OUTPUT is generated in the SM and the read data is transmitted on the SBus to the MBox. SENSE STROBE is inhibited during the core read cycle if the memory reference is an SBus write operation.

CLK MDR is generated to clock the contents of the SBus data lines into the data register. This occurs during the SBus write and RMW operations when new data is to be written in core. With new data (or data read during the previous core read cycle) loaded in the data register, the core write cycle is initiated and the output of each register flip-flop is gated by INH 1 or 2 to turn on an inhibit driver when the flip-flop contains a 0. The resulting inhibit current cancels the effect of the Y write-select current and 0 is written in core.

2.4.6 Core Read Cycle

With reference to Figure 2-19, SM operation begins when the SM select levels and core address are asserted by the controller at the start of an SBus read, write, or RMW operation. If an SM is selected, the select levels assert STACK SEL in the SM's driver modules. STACK SEL then enables the SM's inhibit drivers, X-Y current generators, and stack charge circuits so that they can be activated by the SM control and timing signals generated by the controller's timing module (Subsection 2.3.7).

STACK SEL also enables the address decoder circuits in the SM. This allows the core address from the controller to select a Y driver/switch combination and an X driver/switch combination. As for the enabled current generators and inhibit drivers, the selected driver/switch combinations are activated when control signals are generated by the timing module.

The first of the timing module control signals is A EARLY. It activates the X-Y read current generators, the selected X driver, and the selected Y driver/switch combination. With all Y select circuits activated, Y read current flows to start the core read cycle.

At the same time that Y read current is switched by A EARLY, the SM's data register flip-flops are cleared, provided that an SBus read or RMW operation initiated the core cycle. Timing module signals CLEAR 0 and 1, which assert CLR MDR 0 and 1 in the SM, perform the clearing functions. The register flip-flops are cleared for the SBus read and RMW operations because core read data is subsequently loaded from the sense amplifiers via the direct-set inputs (a "clear-set" load). The register flip-flops are not cleared for an SBus write operation because there is no load of core read data. Instead, SBus write data is clocked into the flip-flops via the D inputs. SBus write data is loaded by Bn CLK (n = 0–3) from the control module. Bn CLK is asserted after A EARLY and before the core write cycle. The exact time depends on when the data word's address is acknowledged by the controller; that is, when the MBox has asserted the write data on the SBus.

The leading edge of RD LATE, asserted by the timing module, activates the remainder of the X read select circuits. With Y read current already flowing, the selected X switch is turned on, causing X read current to flow. The sense amplifiers are then strobed (if SBus read or RMW) to load the core read data into the data register flip-flops.

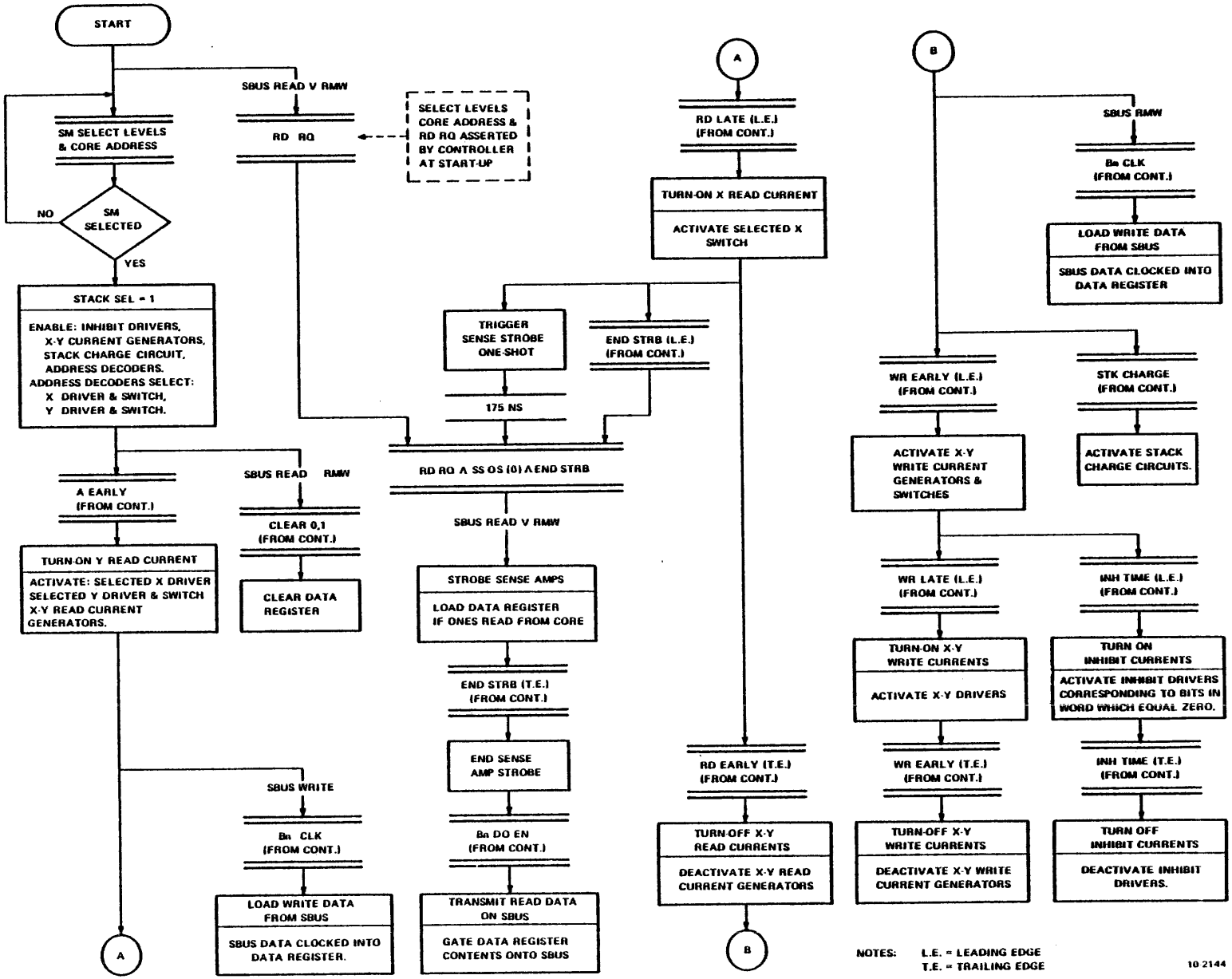


Figure 2-19 MB20 Storage Module, Sequence of Operation

The sense amplifier strobe is generated by RD RQ (asserted by the timing module at start-up), the output of a one-shot (triggered when X read current is switched), and END STRB (asserted by the timing module). The three signals are gated so that the leading edge of the strobe occurs when the one-shot times out. This allows the strobe to be adjusted and margined by changing the one-shot's duration. The strobe is negated by the trailing edge of END STRB.

The core read cycle ends when X-Y read currents are turned off by the trailing edge of RD EARLY. (The leading edge of this signal, which is asserted after A EARLY and before RD LATE, initiates no action.) Bn DO EN (n = 0–3) is asserted by the control module after the sense amplifiers have been strobed. The signal asserts OUTPUT EN in the SM to transmit the data register contents on the SBus. Like Bn CLK (during an SBus write operation), the occurrence of Bn DO EN depends on when the data word's address is acknowledged by the controller; that is, when the SBus data is to be strobed off the SBus by the MBox. For an SBus read operation, Bn DO EN may not be generated until after the core write cycle has started. For an SBus RMW operation, it is asserted prior to the start of the core write cycle.

2.4.7 Core Write Cycle

The core write cycle follows the core read cycle to write the data register contents into memory. For an SBus read operation, the register flip-flops hold the data read during the previous core read cycle. Thus, the write cycle restores the read data in core. (Data must be restored as the read operation leaves the cores in the 0 state.) For the SBus write and RMW operations, when new data is to be stored in memory, the data register flip-flops are loaded from the SBus prior to the start of the core write cycle. Bn CLK clocks the data into the flip-flops as explained previously.

The core write cycle is initiated by the leading edge of WR EARLY from the timing module. The signal activates the X-Y write current generators and the selected X-Y switches. Stack charge circuits are also activated by STK CHARGE. These circuits reverse-bias the unselected stack diodes and shorten write current rise time. STK CHARGE is asserted by the timing module at the same time as WR EARLY.

Data is written in core when X-Y write currents and inhibit currents are turned on by timing module signals WR LATE and INH TIME. The WR LATE signal activates the selected X-Y drivers. INH TIME activates the inhibit drivers. (The inhibit drivers that are activated are those corresponding to bits in the data word that equal 0.) The write core cycle ends when the trailing edge of WR EARLY and INH TIME turn off the stack currents.

SECTION 3
LOGIC DESCRIPTION

3.1 CONTROLLER

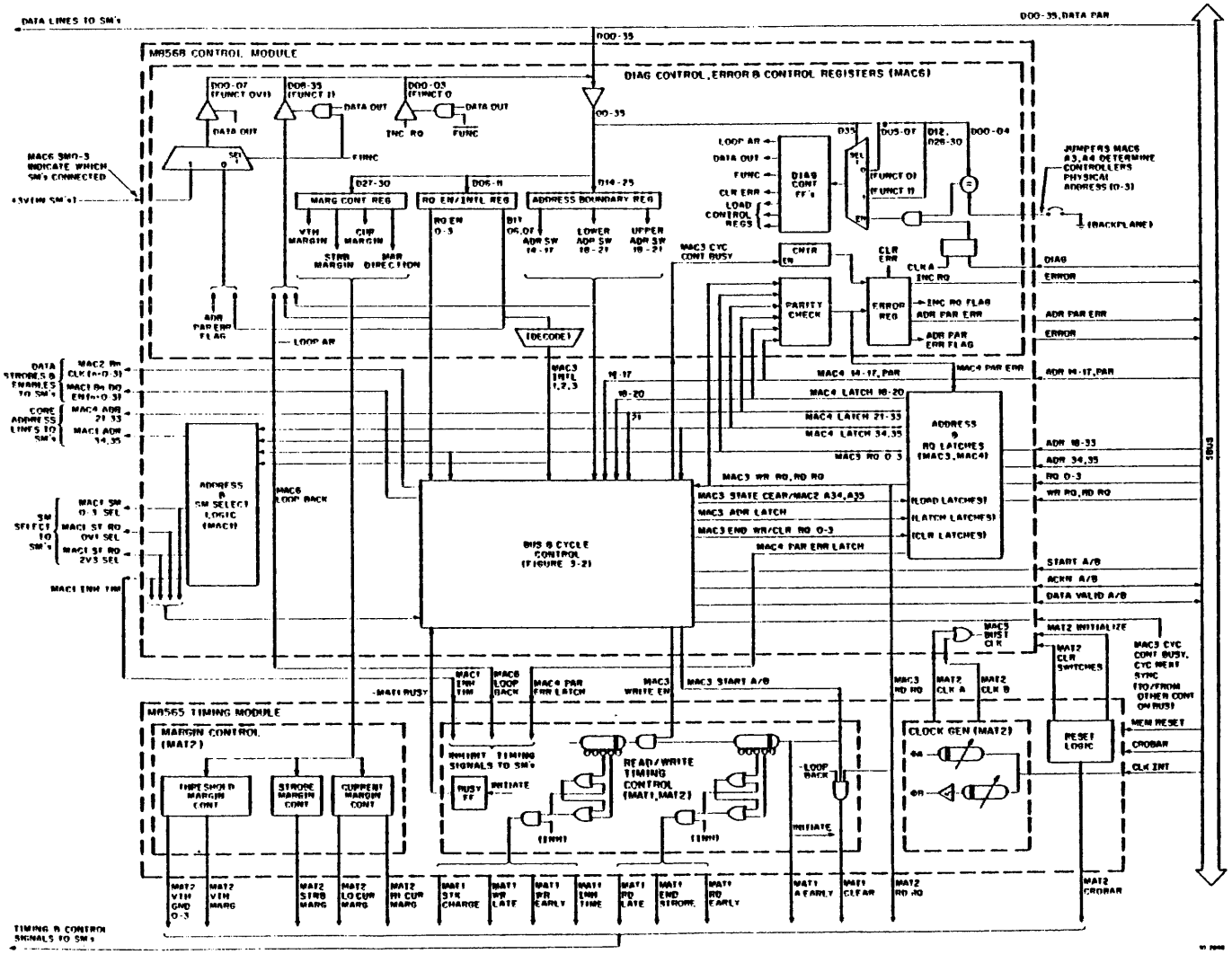
The MB20 controller consists of an M8568 Control module and an M8565 Timing module. Major logic elements for both control and timing modules are shown in Figure 3-1. The bus and cycle control logic is detailed in Figure 3-2. A description of controller operation at the logic level follows. Reference should be made to both figures and to the Field Maintenance Print Set.

3.1.1 Diagnostic Cycle Control

Address boundary, margin control, and request enable/interleave registers are provided in the controller to store the control information transferred during the TO MEMORY portion of the SBus diagnostic cycle. (The diagnostic cycle is described in Subsection 2.1.2 and SBus timing is shown in Figure 2-2.) The controller also has an error register, and logic is provided to gate the error flags, together with other status information, onto the SBus during the FROM MEMORY portion of the cycle. A set of diagnostic control flip-flops load and clear the appropriate registers and cause the specified status information to be gated onto the SBus. Figure 3-3 shows control signal timing for the cycle.

When SBUS DIAG is received by the control module, MAC6 FLOP sets on the next phase A clock (MAC BUSI CLK A). If the controller has been addressed by the SBus diagnostic cycle; that is, if the hard-wired address (MAC6 DIAG A3 and A4) matches the SBus data lines D00–04, comparator output MAC6 SELECT will be true. This signal, together with FLOP, causes MAC6 IN EN to be asserted. IN EN enables two 4 × 2 mixer circuits which gate the appropriate SBus data line inputs to the data inputs of the diagnostic control flip-flops, depending on the function code specified by data line D35. The enabled flip-flops are then set to perform the various control functions. For example, if the function code equals 0 (D35 = 0) and the load enable bit is on (D12 = 1), the mixer output MAC6 LOAD RQ/IN IN will be asserted, enabling flip-flop MAC6 LOAD RQ/INTL. The flip-flop sets on the next phase A clock and loads the interleave mode and request enable levels specified by data lines D06–11 (Table 2-2, Function 0). Control flip-flop MAC6 CLR ERR is set in a similar fashion during a function 0 cycle when the error register is to be cleared (D05 = 1). For a function 1 cycle (D35 = 1), the mixer outputs enable control flip-flops which set to load the address boundary registers and to load and clear the margin control register. Loop-around mode can also be set. The control flip-flops that can be asserted for each function code are shown in Figure 3-3.

To control the gating of status bits onto the SBus, flip-flops MAC6 OUT A, DEL, and FUNC are set as follows. OUT A is enabled directly by IN EN, and DEL is enabled by a mixer output when IN EN goes true. (Both mixer inputs are tied to +3 V so that DEL is enabled to set, regardless of function code.) FUNC is also enabled by a mixer output (with input at +3 V) but it is enabled only for function 1. On the next phase A clock, the flip-flops set but status is not gated to the SBus until MAC6 DATA OUT goes true. DATA OUT is asserted when a flip-flop enabled by OUT A is clocked on by MAC5 BUSI CLK. This clock is the OR of the phase A and B clocks, occurring at twice the SBus clock frequency. DATA OUT remains asserted as long as OUT A is set. DEL is used to keep OUT A set through two phase A clock periods to give a duration for DATA OUT as shown in Figure 3-3. It also latches FUNC, keeping it set through the same interval.



ure 3-1 MB20 Controller, Detailed Block Diagram

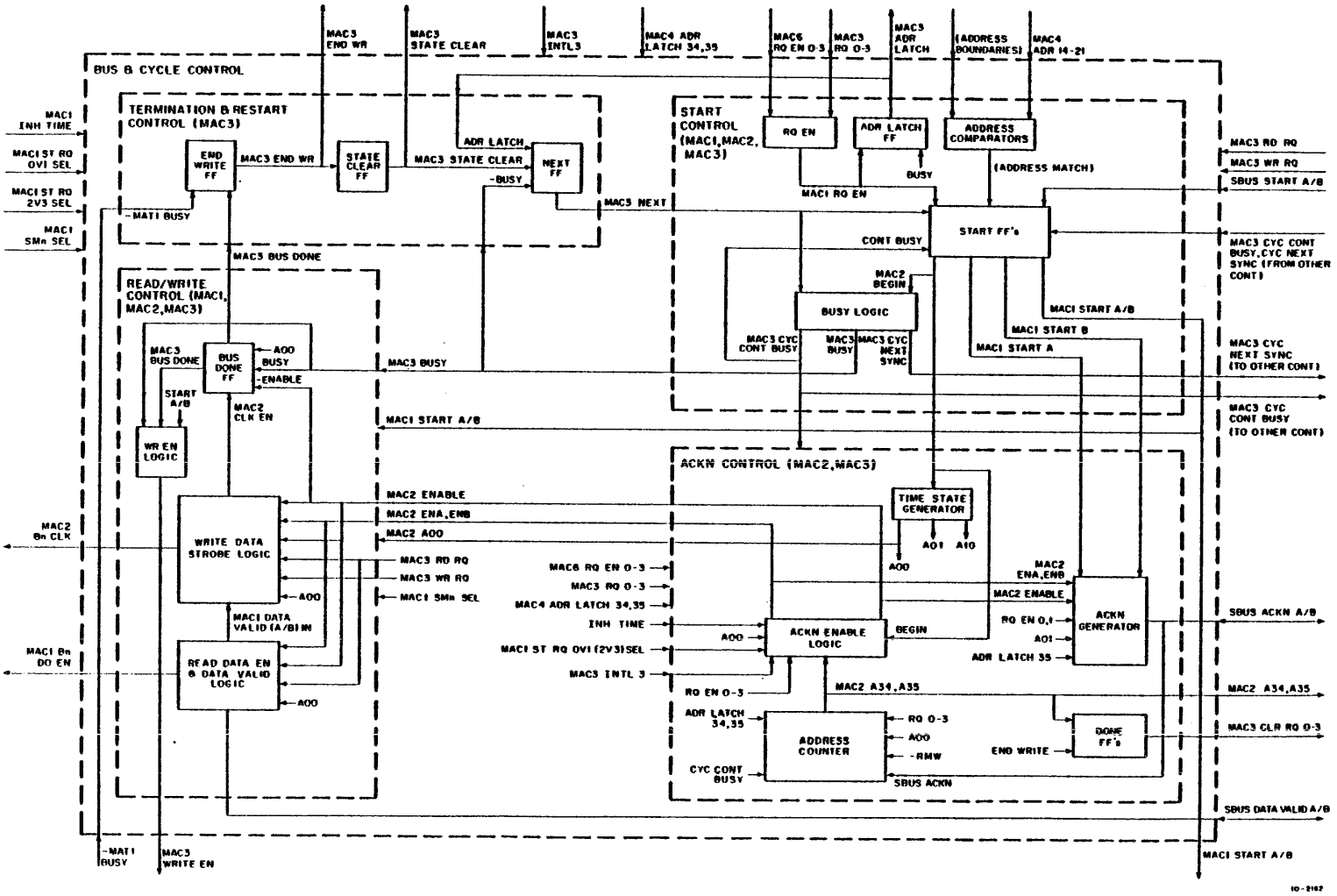


Figure 3-2 MB20 Bus and Cycle Control

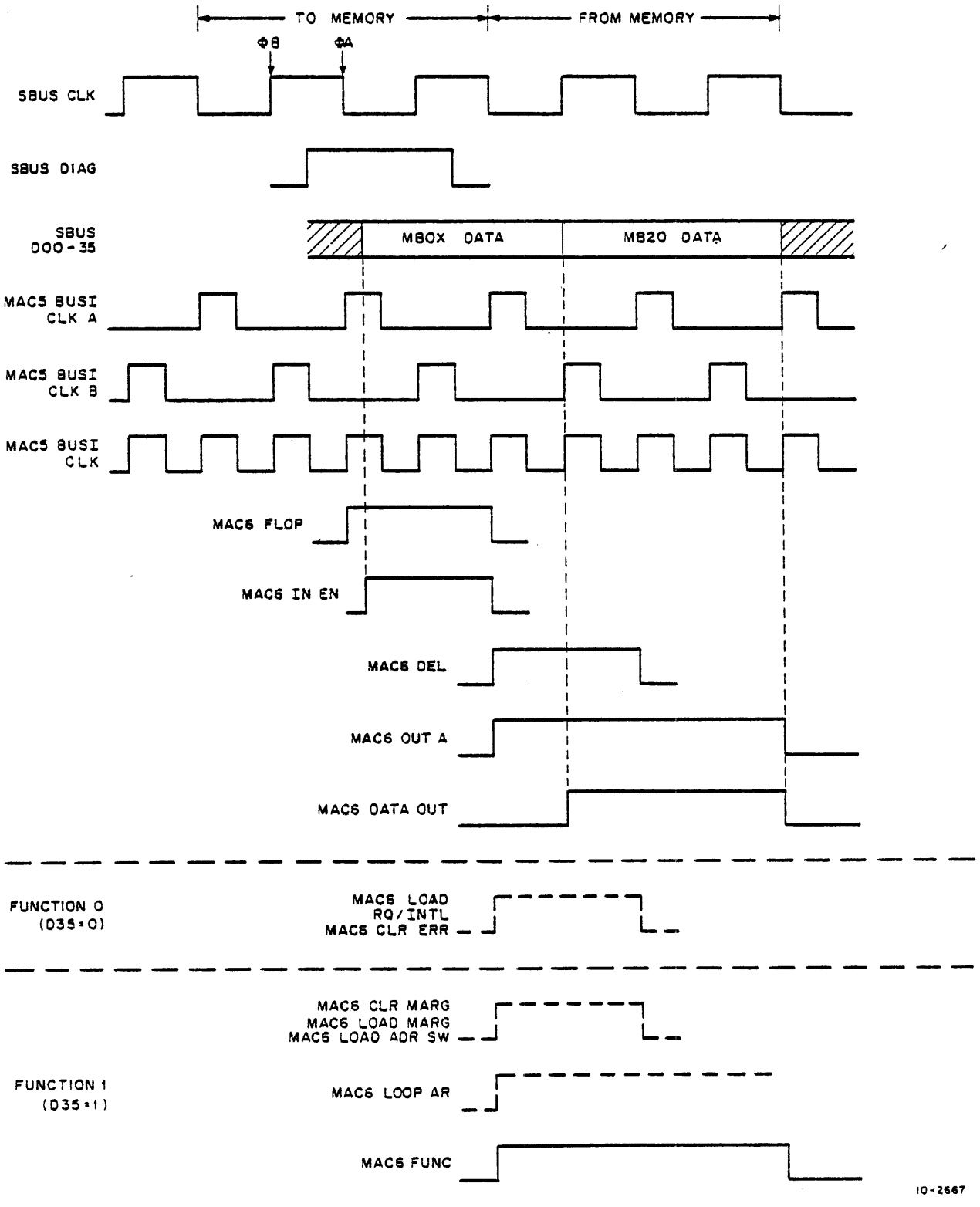


Figure 3-3 Diagnostic Control Timing Diagram

When DATA OUT is asserted, the SBus transmitters are enabled and the status bits are sent to the MBox. Because the information transmitted depends on the function code, DATA OUT is gated with FUNC to select the appropriate transmitters. For example, bit positions 08–35 in the status word (Figure 2-3) contain information for function 1 only. Consequently, the transmitters for these bit positions are enabled only when FUNC = 1. Similarly, transmitters for bits 00–03 are enabled only for function 0 when FUNC = 0. Because information is transmitted in bit positions 04–07 for both function codes, FUNC cannot be gated with the enable level as for the other bit positions. Instead, the transmitters are enabled by only the DATA OUT signal and FUNC selects the correct status information through a 4 × 2 mixer with outputs MAC6 D04–07. DATA OUT remains asserted, enabling the SBus transmitters, until the control flip-flop OUT A is cleared to end the operation. Diagnostic cycle time in the controller equals four phase A clock periods as shown in Figure 3-3.

3.1.2 Memory Addressing and Storage Module Selection

The SBus request and address lines are asserted by the MBox and received by the control module at the beginning of a memory reference. Because the request and address line values are used by the controller for SM selection, core selection, and control purposes during the entire memory reference; that is, after the MBox disconnects and negates the SBus information, the values must be stored in latch circuits until MB20 operation ends. The requests are stored in latches MAC4 RD RQ, MAC4 WR RQ, and MAC3 RQ 0–3. The address is held in latches MAC4 LATCH 18–35. (Address bits ADR 14–17 and ADR PAR are not latched because their values are used only at the beginning of MB20 operation, when the address is compared with the address boundary register and the address parity computation is made.)

MAC3 ADR LATCH controls operation of the latch circuits. If equal to 0, the latch outputs equal the SBus inputs. When the controller is started, ADR LATCH is set to 1 and the address and request information accompanying the SBUS START level is retained in the latch circuits. All latches except LATCH 34, 35, and RQ 0–3 hold this information until the end of MB20 operation when ADR LATCH is cleared. The latch circuits for the starting address and word requests have additional logic as follows.

LATCH 34 and 35 are updated with the contents of the address counter at the end of a core cycle (Subsection 2.3.7). This is done by signals MAC3 END WR and MAC3 STATE CLEAR. END WR first disables the latching path within the latch circuits and the outputs go to 0. STATE CLEAR then goes to 1 and END WR returns to 0. With ADR LATCH still true (another core cycle to follow) and blocking the SBus inputs, the address counter outputs MAC2 A34 and A35 are gated by STATE CLEAR to update the latches.

Also occurring at the end of a core cycle is the clearing of one or more of the word request latches MAC3 RQ 0–3. The latches that are cleared correspond to the words accessed during the cycle. A MAC3 DONE flip-flop is set as each word is acknowledged and when END WR sets, DONE 0–3 are gated to generate MAC3 CLR RQ 0–3. When true, a CLR RQ signal disables the latching paths and the corresponding RQ latch is cleared.

Table 3-1 lists the lines asserted by the controller to select a core location.

The outputs of the address latch circuits are either used directly or are gated as shown depending on the interleave mode. MAC4 ADR 21–33 are the buffered outputs of the address latches and these, together with MAC1 ADR 34 and 35, are gated to the core address decoder inputs of all SMs connected to the controller. ADR 34 and 35 equal LATCH 34 and 35 in no-interleave mode (Subsection 2.2.3) and they equal LATCH 19 and 20 in 4-way interleave mode (Subsection 2.2.1). In 2-way interleave mode (Subsection 2.2.2), ADR 35 is gated as in 4-way interleave mode, but ADR 34 is asserted as a function of the stored word requests as shown in Table 3-1. This decoding is required because a controller does not select just one core address in the selected SMs (as in 4-way interleave mode). One of two possible SM locations have to be selected and the one available address bit (LATCH 34) is used

only when the controller is addressing the starting address (S) or updated starting address (S'); that is, when the address in the latches is odd and the controller is odd or when the address is even and the controller is even. In a controller selecting a word with an address other than S or S', ADR 34 must be calculated depending on which word has been requested for the current core cycle and it becomes a function of the stored word requests. Also, the calculation depends on whether a controller is odd or even.

Table 3-1 Memory Address Selection

Address Latches (MAC4 LATCH n)			Address Lines to SM
21–33			→ MAC4 ADR 22–33
34	*See below	19	→ MAC1 ADR 34
35	20		→ MAC1 ADR 35
No Interleave	2-Way Interleave	4-Way Interleave	

*Decoding to generate MAC1 ADR 34 in 2-way interleave mode

Controller	Address Latches (MAC4 LATCH n)	Request Latch (MAC3 RQ n)	MAC1 ADR 34
EVEN	$\overline{34}$, $\overline{35}$	(0 V)	0
	$\overline{34}$, 35	RQ2	1
	34, $\overline{35}$	(+3 V)	1
	34, 35	$\overline{RQ0}$	1
ODD	$\overline{34}$, $\overline{35}$	$\overline{RQ1}$	1
	$\overline{34}$, 35	(0 V)	0
	34, $\overline{35}$	RQ3	1
	34, 35	(+3 V)	1

The decoding to assert MAC1 ADR 34 is done with a 2 × 4 mixer circuit that uses the binary value of S or S' (LATCH 34, 35) to select a stored word request or a 0 or 1 logic level (0 V or +3 V). Only one of the mixer circuits is enabled depending on the odd/even status of the controller. The preloaded request enable levels determine odd/even status and MAC6 RQ EN 2 and 3 connect to the mixer's enable inputs. When LATCH 35 equals 0 in the even controller or 1 in the odd controller, 0 V and +3 V are gated through the mixer to determine the value of bit 34. Although the 0 and 1 logic levels are used, from the table it is seen that they input the value of LATCH 34. This is for the case mentioned previously, where the value of the address latch is used directly to generate the core address when referencing S or S'. The other active controller in an interleaved operation must generate ADR 34 by gating the request latches for the same value of LATCH 34 and 35.

To illustrate 2-way interleave mode memory addressing, assume an SBus address of 0s and requests for words 0, 1, and 3. Words 0 and 1 have an SM core address equal to 00 (2 LSBs) and word 3 has an address of 10. (Refer to Subsection 2.2.2 for quad-word distribution.) LATCH 34 and 35 equal the starting address (00) during the first core cycle. Also, two controllers will be active with the even controller selecting word 0 and the odd controller selecting word 1. The even starting address (00) in the even controller gates the value of LATCH 34 through the mixer so that address bit 34 equals 0. This causes location 00 to be addressed as required. The odd controller must also generate this address, and it is seen from Table 3-1 that when LATCH 34 and 35 are equal to 00 and RQ1 is true, bit 34 does equal 0. LATCH 34 and 35 are then updated to a value of 11 at the end of the core cycle. During the second core cycle, word 3 is selected by the odd controller. Again (with reference to the table), LATCH 34 and 35 select the value of LATCH 34 as address bit 34. Since LATCH 34 is equal to 1, the correct core address (10) is generated for the second core cycle interval.

Decoding logic similar to that for generating bit 34 is provided to generate MAC1 INH TIM. This signal is asserted in IL2 mode for a special case occurring when words with addresses S, S+2, and S+3 are requested. As seen in Figure 2-5, two core cycle intervals are required to access the three words. Also, because the operation is interleaved, two controllers go active and step together through parallel core cycles. The controller accessing the single address S+3 must do so during the second cycle, however. Thus, to maintain synchronization with the other controller during the first core cycle, it performs a dummy cycle by inhibiting read/write select currents in the referenced SM. INH TIM does this and it is wired to the timing module to deselect the SM and to prevent the SM timing signals from being generated for this special case. The signal is asserted by either the odd or even controller for four request latch combinations, one for each value of starting address. These are shown in Table 3-2.

Table 3-2 Decoding for Special Case in 2-Way Interleave Mode

Controller	Address Latches (MAC4 LATCH n)	Request Latches (MAC3 RQn) To Assert MAC1 INH TIME
EVEN	$\overline{34}$, 35	RQ1, $\overline{RQ2}$, RQ3
	34, 35	RQ3, $\overline{RQ0}$, RQ1
ODD	$\overline{34}$, $\overline{35}$	RQ0, $\overline{RQ1}$, RQ2
	34, $\overline{35}$	RQ2, $\overline{RQ3}$, RQ0

Six control module outputs are used for SM selection. These are listed in Table 3-3.

Each of the four select levels MAC1 SMn SEL (n = 0–3) connect to one of the four storage modules SM 0–3. The other two select levels each connect to two SMs; MAC1 ST RQ 0V1 SEL to SM0 and SM2 and MAC1 ST RQ 2V3 SEL to SM1 and SM3. A single SM is connected by two of the select lines and both must be asserted for an SM core cycle to take place. For example, the controller must assert SM0 SEL and ST RQ 0V1 SEL to select SM0.

One SM is selected by a controller in no-interleave and 2-way interleave modes. ST RQ 0V1 SEL and ST RQ 2V3 SEL are both forced to ONE in these modes and selection is made by decoding LATCH 19 and 20 to generate the appropriate SMn SEL level. The binary value of the two address latches corresponds to the SM selected as shown in Table 3-3. Also shown is SM selection in 4-way interleave mode where one or two SMs are selected by a controller, depending on the number of words requested. The value of LATCH 18 causes a pair of the SMn SEL levels to be asserted, either the levels for SM0 and SM1 or the levels for SM2 and SM3, while the stored word requests raise one or both of ST RQ 0V1 SEL or ST RQ 2V3 SEL. The three lines (one word requested) or the four lines (two words requested) that are raised combine to select the correct SMs.

Table 3-3 Storage Module Selection

Controller	Request Latches (MAC3 RQn)			Select Line to SM
EVEN	N/A both SEL levels = 1		RQ0	MAC1 ST RQ 0V1 SEL
ODD			RQ1	
EVEN	In 2-way and no-interleave modes		RQ2	MAC1 ST RQ 2V3 SEL
ODD			RQ3	
	no interleave	2-way interleave	4-way interleave	
	Address Latches (MAC4 LATCH n)			
EVEN/ODD	$\overline{20}$, $\overline{21}$	$\overline{19}$, $\overline{20}$	$\overline{19}$	MAC1 SM0 SEL
EVEN/ODD	20, $\overline{21}$	19, $\overline{20}$	19	MAC1 SM2 SEL
EVEN/ODD	$\overline{20}$, 21	$\overline{19}$, 20	$\overline{19}$	MAC1 SM1 SEL
EVEN/ODD	20, 21	19, 20	19	MAC1 SM3 SEL
	no interleave	2-way interleave	4-way interleave	

Three 2 × 4 mixer circuits are used to generate the SM select levels. Two circuits are used for the SMn SEL lines, each using the stored interleave mode bits, MAC6 BIT 06 and 07, at the select inputs. These bits select one of the two address latches to be decoded in no interleave and 2-way interleave modes or they select a 1 logic level in 4-way interleave mode. To complete the decoding in no-interleave and 2-way interleave modes, the appropriate mixer output is enabled with the second of the two address latches. In 4-way interleave mode, the single address latch that is decoded enables the correct mixer output and gates the 1 input selected by BIT 06 and 07.

One mixer circuit is used to generate ST RQ 0V1 SEL and ST RQ 2V3 SEL. In 4-way interleave mode, MAC3 INTL3 (asserted in 4-way interleave mode) and MAC6 RQ EN 1 at the mixer select inputs pick a word request latch depending on controller odd/even status. ST RQ 0V1 SEL is true when RQ0 is true in the even controller or if RQ1 is true in the odd controller. Similarly, ST RQ 2V3 SEL is generated by RQ2 in the even controller or RQ3 in the odd. The mixer outputs are both asserted in no-interleave and 2-way interleave modes and do not depend on odd/even status; INTL3 will be false causing +3 V to be selected, regardless of the state of RQ EN 1.

Two timing module outputs also play a part in SM selection. MAT2 SPECIAL +3V1 and 2 are ANDed with the other two select levels (SMn SEL and ST RQ 0V1 SEL or ST RQ 2V3 SEL) in each storage module. Normally, the +3 V signals are asserted, allowing the other two levels to select the appropriate SM. However, both +3 V signals are negated by MAT2 PAR ERR INH cycles to deselect all SMs whenever an address parity error has been detected or when the controller is in loop-around mode. The signals are also negated by INH TIME; that is, for the special case occurring in 2-way interleave mode.

3.1.3 Start Control

One of two START signals can be asserted by the MBox to initiate a memory reference; SBUS START A is generated coincident with phase A of SBUS CLK and SBUS START B is generated coincident with phase B (Subsection 2.1.1). When received by the control module, SBUS START A sets flip-flop MAC1 START A on the next phase A clock provided conditions for start-up are met. Correspondingly, SBUS START B sets MAC1 START B on the phase B clock. Either the START A or START B flip-flop starts controller operation and only one will be set depending on which SBUS START level is asserted.

The data inputs of the START flip-flops are enabled by the outputs of a 2 × 4 mixer circuit. This mixer and its input gating comprise the logic which defines the start and restart conditions discussed in Subsections 2.3.1 and 2.3.6. For example, MAC1 RQ EN enables the mixer outputs and is the result of a comparison between the word request latches (MAC3 RQ 0–3) and the preloaded request enable levels (MAC6 RQ EN 0–3). It defines the start-up condition that at least one of the word requests must be enabled in the controller. If the MBox has not directed word requests to the controller, MAC1 RQ EN will be false, the START flip-flops cannot be enabled through the mixer, and the SBUS START level is effectively ignored.

Start and restart control depend mainly on controller busy status. Two busy indicators connect to the mixer select inputs and determine the gating of the SBUS START level for start-up control or the MAC3 NEXT signal for restart control. The busy indicators are the controller's CYC CONT BUSY flip-flop and the CYC CONT BUSY flip-flop output wired from the other controller on the same SBus. (MAC3 CYC CONT BUSY sets when a controller is started and is true to the end of the last core cycle.) The other controller's busy status is a factor as both controllers operate as a pair in interleaved operation.

With both controllers inactive, the mixer gates SBUS START to enable the START flip-flop, provided the SBus address falls within the address space specified by the preloaded address boundary registers. The address comparison is made by three 4-bit comparator circuits, the outputs of which are gated with SBUS START at the input to the mixer. All three comparator outputs must be true. MAC4 14–17 must equal MAC6 ADR SW 14–17 (MAC4 EQUAL TO = 1), MAC4 LATCH 18–21 must be less than or equal to MAC6 UPPER ADR SW 18–21 (MAC4 LESS THAN = 1), and MAC4 LATCH 18–21 must be greater than or equal to MAC6 LOWER ADR SW 18–21 (MAC4 GREATER THAN = 1).

If the controller is inactive and the other controller is busy, the mixer selects SBUS START through gating which prevents start-up when the operation is interleaved (both odd and even word requests in 2-way or 4-way interleave mode). The two controllers must start together in this case as explained previously. When the other controller goes inactive, the mixer select inputs change to gate SBUS START and start the controller as explained in the preceding paragraph. If the operation is not interleaved, the controller will start with the other controller still busy. This speeds overall operation for systems which normally make many non-interleaved memory references; e.g., a system which does not contain a Cache.

Once a controller is active, the mixer directs NEXT to the START flip-flops in case a controller restart is necessary. Restarts occur in 2-way interleave and no-interleave modes when a multiword request necessitates that successive core cycles be generated (Figure 2-5). As for the first core cycle, core cycles resulting from restarts must be synchronized in an interleaved operation. MAC3 CYC NEXT SYNC ensures this and it is gated with NEXT at the mixer input. CYC NEXT SYNC is the OR of NEXT and the negation of CONT BUSY from the other controller. It will delay the enabling of the START flip-flops until the other controller, if active, is also ready to restart. Core cycles will then start together as required.

The mixer gates the enable level to both START flip-flops during a restart so that one or the other sets on the next clock, either phase A or B. This allows core cycles to start as soon as possible; that is, on the first occurring clock phase. Because CYC NEXT SYNC is still true for the clock period following restart, the negation of MAC2 B is used to disable the mixer input, preventing the other START flip-flop from setting. B goes true (negation goes false) as soon as one of the START flip-flops has set.

START A or START B starts controller operation by asserting MAC3 START AVB and MAC2 BEGIN. The START AVB signal is wired to the timing module to start the control signal timing train necessary for SM operation. BEGIN starts the control logic sequence by setting flip-flops MAC3 CYC CONT BUSY, MAC3 BUSY, MAC2 ENABLE, and MAC2 A00 on the next MAC5 BUSI CLK. BUSY, in turn, sets MAC3 ADR LATCH to latch the SBus address and request line contents as explained in Subsection 3.1.2.

3.1.4 ACKN Control

ACKN control signal timing is shown in Figure 3-4. The ACKN control logic consists mainly of the following:

1. A 2-bit address counter that is initially loaded with the starting address asserted on the SBus. The counter is then incremented to the address of the next word to be accessed as each ACKN is generated.

2. An ENABLE flip-flop and associated compare logic that allows ACKN to be generated when the starting address, updated starting address, or the incremented address in the address counter corresponds to the appropriate preloaded request enable level.

3. Clearing logic for ENABLE which stops generation of ACKN when the correct number of addresses have been acknowledged during a core cycle.

The address counter consists of two flip-flops (MAC2 A34, A35) clocked by MAC5 BUSI CLK; a 2 × 4 mixer circuit used to condition the counter flip-flops and to control the preloading, latching, and incrementing of the counter; and a circuit (MAC2 NEW 34, 35) that computes the next value of the counter based on the current value and the stored word requests. With CYC CONT BUSY still equal to 0 at start-up, the mixer selects LATCH 34 and 35 to condition the counter flip-flops. The latches hold the starting address and the counter flip-flops are clocked to this value at the same time that the START flip-flop sets. To compute the counter's next value, A34 and A35 are connected to another 2 × 4 mixer circuit which generates the next address at mixer outputs MAC2 NEW 34 and 35. The next address depends on which words have been requested and it is generated by having the current address in the counter select various combinations of the word request latches (MAC3 RQ 0–3) at the mixer inputs. The address counter is incremented when the new address is loaded in the counter flip-flops. (The increment occurs when ACKN is generated.) NEW 34 and 35 will then change to the next address in preparation for the next increment.

The address counter contents are compared with the RQ EN levels to generate ACKN signals. However, to generate the first ACKN as soon as possible and with minimum circuit delay, the START flip-flops are gated directly to acknowledge the starting address.

LATCH 35 is compared with the RQ EN levels and if the address is odd and the controller is odd, or if the address is even and the controller is even, START A asserts SBUS ACKN A and START B asserts SBUS ACKN B. Only one START flip-flop will be set and the ACKN signal is asserted on the clock phase corresponding to the START level received.

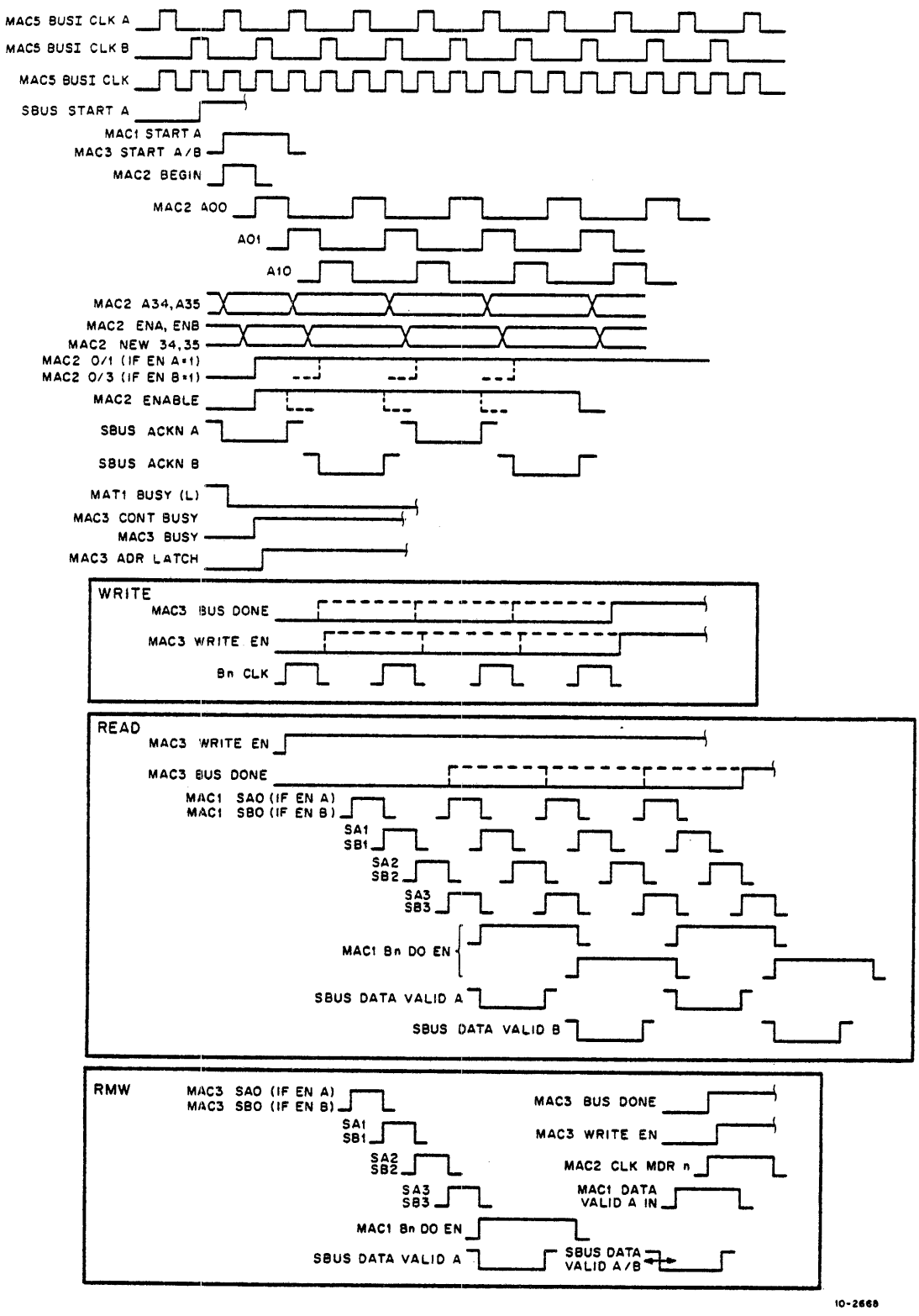


Figure 3-4 Memory Control Timing Diagram (Control Module)

As stated in Subsection 3.1.2, the START flip-flop also asserts MAC2 BEGIN which sets MAC2 ENABLE and MAC2 A00 on the next MAC5 BUSI CLK. A00 is the first stage of a three flip-flop ring counter (MAC2 A00, A01, A10) that is used as a time state generator. With SBUS ACKN still asserted to acknowledge the starting address, A00 switches the mixer inputs gated to the address counter flip-flops causing NEW 34, 35 to update the counter to the next address. To acknowledge the next address (and all others following the first address in a core cycle), enable levels MAC2 EN A and EN B are used. These levels are the outputs of a 2 × 4 mixer circuit which have the address counter outputs connecting to the select inputs. The mixer then selects the preloaded RQ EN levels depending on the address.

If the corresponding RQ EN level is set for the word address in the address counter, EN A and/or EN B goes to 1 (Table 3-4) to enable one of the inputs to both SBUS ACKN A and B.

Table 3-4 EN A and EN B

	Address Latch (MAC4 LATCH n)	Request Enable Level (MAC6 RQ EN n)	EN A	EN B
No Interleave and 2-Way Interleave Modes	$\overline{34}$, $\overline{35}$	RQ EN 0	1	1
	$\overline{34}$, 35	RQ EN 1	1	1
	34, $\overline{35}$	RQ EN 2	1	1
	34, 35	RQ EN 3	1	1
4-Way Interleave Mode	$\overline{34}$, $\overline{35}$	RQ EN 0	1	0
	$\overline{34}$, 35	RQ EN 1	1	0
	34, $\overline{35}$	RQ EN 2	0	1
	34, 35	RQ EN 3	0	1

The address is then acknowledged when a flip-flop sets on the next clock phase and asserts the other input to either SBUS ACKN A or B. There is a flip-flop for each ACKN signal, each clocked by the corresponding clock phase and both enabled by ENABLE and time state A01. If the next clock following time state A01 is phase A, the flip-flop clocked on phase A sets to assert SBUS ACKN A. Correspondingly, SBUS ACKN B is generated by the other flip-flop if the next clock is phase B. Timing is such that successive ACKN signals in one core cycle interval are generated on alternate phases.

EN A and EN B are also used to determine when all addresses have been acknowledged during a core cycle interval. A 2 × 4 mixer circuit gates EN A to flip-flop MAC2 0/1, and EN B to flip-flop MAC2 2/3. When a word is acknowledged, the asserted enable level (A/B) is stored in the corresponding flip-flop on the next MAC5 BUSI CLK. Once set, 0/1 and 2/3 are latched by the mixer circuit and are gated to condition the ENABLE flip-flop depending on interleave mode. Gating is such that ENABLE will be cleared on the clock following time state A00 after the correct number of words have been acknowledged. For example, with two words requested in an interleaved operation (words 0 and 2 in the even controller, or words 1 and 3 in the odd controller), MAC1 ST RQ 0V1 SEL and MAC1 ST RQ 2V3 SEL will be true at the input to the ENABLE flip-flop. In 4-way interleave mode, the AND of these two select levels causes ENABLE to clear when both 0/1 and 2/3 are set. This is done because first EN A and then EN B (or vice versa, depending on the order words are accessed) are asserted for this case (Table 3-4) and both words have been acknowledged when both 0/1 and 2/3 have been set. If just one word is requested in 4-way interleave mode, or if the controller is in no-interleave or 2-way interleave mode (except for special case), ENABLE is cleared by either 0/1 or 2/3. This allows only

one ACKN to be generated per core cycle as required. For the special case in 2-way interleave mode (Subsection 2.3.1), both controllers start but an address is not acknowledged by one controller during the first core cycle interval. Consequently, ENABLE must be cleared in this controller after the other controller acknowledges the starting address. Since EN A or EN B will not be asserted as usual, MAC1 INH TIME is used to disable the latching input and ENABLE is cleared on the clock following the negation of BEGIN.

Four flip-flops, MAC3 DONE 0–3, are used to keep track of which words have been acknowledged during an operation. Each flip-flop is enabled by an output from a 4 × 2 mixer circuit. The mixer is configured to decode the binary value of A34 and A35 so that one of the four outputs will be asserted, depending on the address counter contents. During time state A00, with ENABLE still true (MAC2 A = 1), the mixer is enabled to assert an output and the DONE flip-flop corresponding to the address in the address counter is set on the next MAC5 BUSI CLK. Thus, a flip-flop is set for each word acknowledged as the address counter increments during the operation. Once set, the DONE flip-flops are gated to clear the word request latches at the end of a core cycle (Subsection 3.1.7).

3.1.5 Read/Write Control (Control Module)

In addition to acknowledging the address of each word, the control module generates the control signals necessary to strobe write data from the SBus into the SM data register and gate read data from the data registers onto the SBus. It must also generate DATA VALID signals concurrent with placing read data on the SBus. Timing for control module signals is shown in Figure 3-4.

MAC2 Bn CLK (n = 0–3) are the write data strobe signals. Each of the four flip-flop outputs connect to an SM and only one will be set at any one time. Each flip-flop is clocked by MAC5 BUSI CLK and enabled by a MAC1 SMn SEL level, MAC2 EN A or B, and MAC2 CLK EN. At least one SMn level will be asserted (Table 3-3) and CLK EN will be true for the SBus write operation (RD RQ = 0) during time state A00 as long as ENABLE is set. With the data word on the SBus and the controller acknowledging the word address, an EN A or EN B level will also be true to enable generation of the the write strobe just after the SBUS ACKN signal is asserted. In no-interleave and 2-way interleave modes, EN A and EN B both go true and the one write strobe generated per core cycle is determined by the single SMn SEL level that is asserted. In 4-way interleave mode, two SMn SEL levels are asserted but only one of the EN A or EN B levels is true (Table 3-4). One or two strobes can be generated per core cycle and combinational gating of the levels enables the correct Bn CLK flip-flop when a word is acknowledged.

During a read-modify-write operation (MAC2 RMW = 1), CLK EN is asserted by the SBUS DATA VALID signal accompanying the modified data from the MBox. To ensure that the correct strobe is generated at this time, MAC2 RMW is used during the first (read) portion of the operation to inhibit the incrementing of the address counter when the word address is acknowledged. Inhibiting the count results in EN A and EN B having the same value when CLK EN is generated for the read-modify-write operation as when it is generated for the write operation. Thus, EN A and EN B, together with the SMn SEL levels, will enable the correct write strobe as described in the preceding paragraph.

MAC1 Bn DO EN 0–3 are the signals wired to the SMs to gate the read data onto the SBus data lines. The flip-flop outputs are asserted during the SBus read and RMW operation. The logic to enable and set the signals is similar to that for generating the write data strobes; the major difference is that the EN A and EN B levels do not enable the flip-flops directly. Instead, each connect to the input of a 6-stage shift register. The last shift register stages, MAC1 SA3 and SB3, then enable the Bn DO flip-flops. The input gates to the shift registers are asserted during time state A00 when ENABLE is set, just as the enable gates for the write data strobes, but the shift registers introduce a six clock period delay before a Bn DO EN flip-flop is set. The delay is to allow time for the SM to complete the core read cycle (initiated by the START flip-flop) and load its data register. It is the data register contents that are gated on the SBus by a Bn DO EN signal.

The shift register outputs are also used to enable MAC1 DATA VALID A out and MAC1 DATA VALID B out. SA3 or SB3 enables both flip-flops and one will set on the next clock phase (A or B) asserting DATA VALID (A or B) on the SBus at the same time that a Bn DO EN signal is asserted. The DATA VALID signal for a word is generated on the same clock phase as the ACKN signal for that word.

MAC3 BUS DONE is set to signal the end of SBus dialogue with the MBox. Together with MAC3 END WR, which signals the end of the SM core cycle, it causes the controller to terminate or restart as explained in Subsection 3.1.7. MAC 3 BUS DONE EN enables the BUS DONE flip-flop. It is one of the outputs from a 2 × 4 mixer circuit having MAC4 RD RQ and MAC4 WR RQ as select levels. The mixer is enabled when all ACKN signals have been generated (ENABLE = 0). For an SBus write operation, this is the end of SBus activity and the mixer gates the previously set MAC3 BUSY signal to set BUS DONE on the next MAC5 BUSI CLK. For an SBus read operation, BUS DONE EN is asserted by MAC1 RD CYC DONE DLY during the second A10 time state following the negation of ENABLE. This sets BUS DONE one clock period before the last DATA VALID signal. It is set early so that during multi-word read operations, where the core cycle ends before all DATA VALID signals have been asserted, the controller can begin the termination sequence and be ready for another memory reference shortly after SBus activity ends. During a RMW operation, the DATA VALID signal asserted by the MBox is the last SBus signal generated. CLK EN goes true at this time and it is gated directly through the BUS DONE EN mixer circuit to set BUS DONE.

MAC3 WRITE EN is wired to the timing module to enable the start of the control signal timing train that causes an SM write core cycle. The write core cycle will start provided the core read cycle has ended. WRITE EN is the output from the other half of the 2 × 4 mixer circuit used to generate BUS DONE EN. During an SBus write or RMW operation, all write data has been strobed from the SBus when SBus dialogue has ended and the mixer selects BUS DONE to assert WRITE EN. For an SBus read operation, the start of the core write cycle is not SBus-dependent. The cycle serves only to restore the data read from core. Thus, WRITE EN is asserted by BUSY at the beginning of the operation, enabling the core write cycle to start immediately after the core read cycle ends.

3.1.6 Read/Write Control (Timing Module)

Read/write control signals generated by the timing module are listed in Subsection 2.3.7. Control signal timing is shown in Figure 3-5. Circuitry to generate the R/W control signals consists mostly of R-S flip-flops that are set and cleared by delay line outputs. Some of the SM control signals are inhibited during special operating modes. MAT1 START INH CYCLES prevent RD EARLY from being transmitted to the SMs when the controller is in loop-around mode (MAC6 LOOP BACK = 1) or when it is performing a dummy cycle for the special case in 2-way interleave mode (MAC1 INH TIME = 1). Another signal, MAT1 PAR ERR INH CYCLES, also inhibits RD LATE, WR EARLY, and WR LATE for these two cases. PAR ERR INH CYCLES is also asserted when an address parity error occurs (MAC4 PAR ERR LATCH = 1). In addition to inhibiting transmission of the control signals to the SMs, PAR ERR INH CYCLES also negates SM select levels MAT2 SPECIAL +3V1 and 2 (Subsection 3.1.2). Deselecting the SMs and inhibiting the control signals prevents read/write currents in the referenced SMs. Subsection 2.1.8 explains why this is necessary in loop-around mode and when an address parity error occurs. The reason for inhibiting SM read/write currents for the special case in 2-way interleave mode is explained in Subsection 2.3.1.

When MAC3 START A/B is generated by the control module, MAT1 A EARLY is asserted to turn on the core Y read select currents, starting SM operation. At the same time, MAT1 INITIATE is asserted to generate MAT1 CLEAR 0 and 1. The CLEAR signals clear the SM data registers if the operation is a read or read-modify-write (MAC4 RD RQ = 1) and if loop-around mode is not set (MAC6 LOOP BACK = 0). INITIATE also drives the first delay line to start the timing train and it toggles the R-S flip-flops that generate MAT1 RD EARLY and MAT1 BUSY. RD EARLY times the core read select circuits. MAT1 BUSY is the timing module busy indicator.

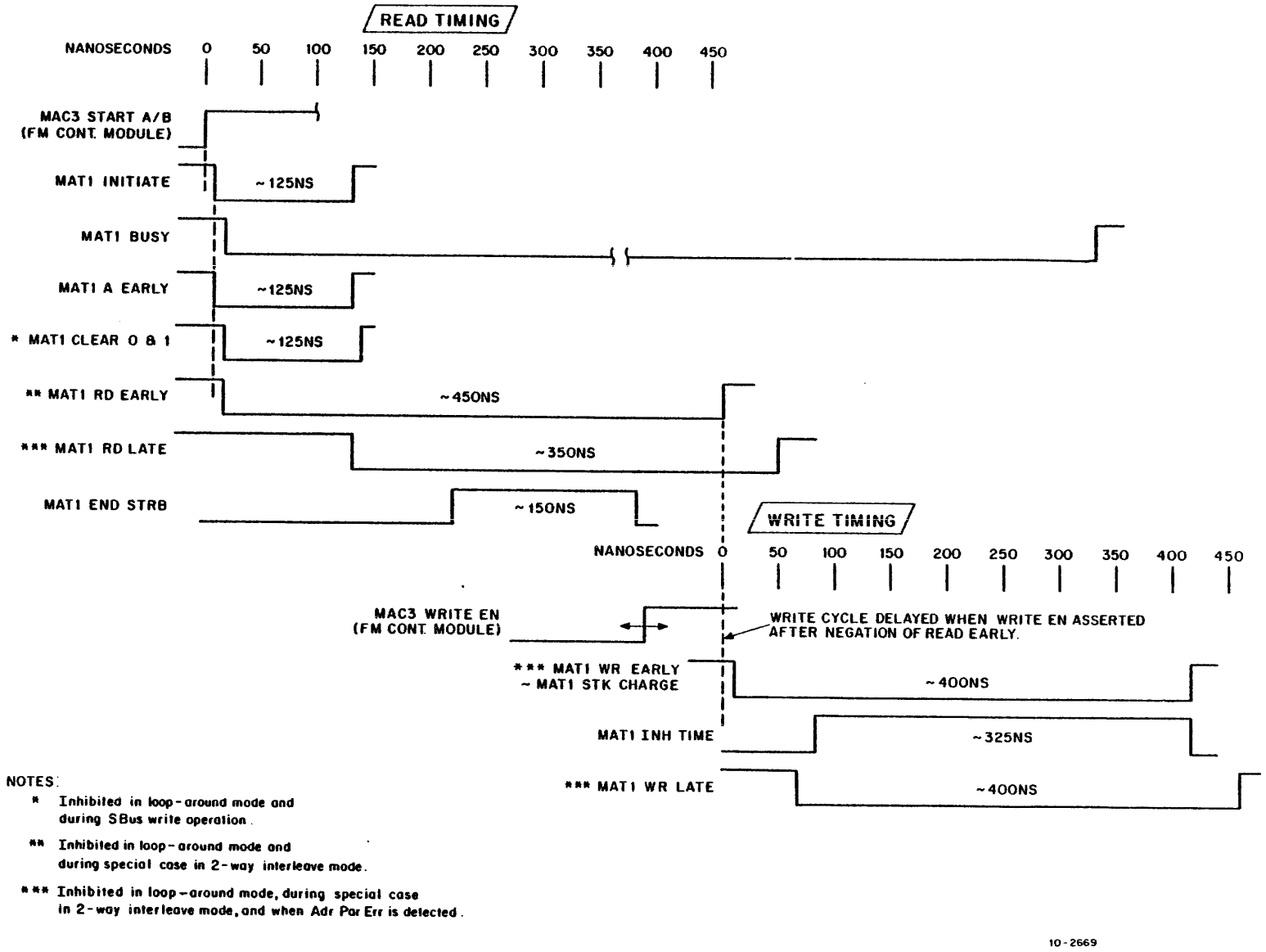


Figure 3-5 Memory Control Timing Diagram (Timing Module)

The first delay line output toggles a flip-flop which negates both INITIATE and A EARLY. The flip-flop output is ANDed with START AVB to allow generation of both signals initially. When negated, the flip-flop disables the inputs and establishes a signal duration for INITIATE and A EARLY equal to the tapped delay (125 ns). The same delay line output also toggles the flip-flop that asserts MAT1 RD LATE. This signal turns on the core X read select currents. As successive delay line outputs go true, a flip-flop is toggled to generate MAT1 END STROBE, and the RD EARLY and RD LATE signals are cleared to end core read cycle timing.

WRITE EN from the control module edge-triggers a D-type flip-flop when received by the timing module. The stored enable level is ANDed with one output from the R-S flip-flop which generates RD EARLY. As soon as the core read cycle ends, or if it has already ended, and the R-S flip-flop is in the state to negate RD EARLY, the first delay line in the core write cycle timing train is driven, and an R-S flip-flop is toggled to generate the first of the write control signals. The signals, MAT1 WR EARLY and its complement, MAT1 STK CHARGE, activate the core write-select circuits. Delay line outputs then toggle the flip-flops associated with MAT1 WR LATE and MAT1 INH TIME. These signals turn on the write select and inhibit currents. Successive delay line outputs then clear the stored write enable level, the four core write control signals, and the MAT1 BUSY signal to end the core cycle timing train. MAT1 BUSY is wired to the control module to initiate controller termination or restart. When MAT1 BUSY is negated, its complement, MAT1 END WRITE EARLY, is asserted to toggle the R-S flip-flop ANDed with START AVB. This allows START AVB to initiate another core cycle timing train.

3.1.7 Termination and Restart Control

The negation of MAT1 BUSY in the timing module signals the end of the SM core cycle. When received by the control module, it sets MAC3 END WR and starts the termination or restart sequence. Timing is shown in Figure 3-6. Synchronizing flip-flops are used to receive the negation of MAT1 BUSY. This is because timing module signals are sequenced by delay lines and they are asynchronous with respect to MAC5 BUSI CLK in the control module. Two stages of synchronization are required for reliable operation at the controller's clock rate. With SBus dialogue ended (BUS DONE = 1), the second synchronizing flip-flop enables END WR to set on the next MAC5 BUSI CLK. END WR disables the latching input to MAC3 BUSY and enables MAC3 STATE CLEAR. It also gates the four MAC3 DONE flip-flops to assert one or more of MAC3 CLR RQ 0–3. These signals clear the corresponding stored word requests (MAC3 RQ 0–3) as discussed in Subsection 3.1.2. If all the stored word requests enabled by the preloaded request enable levels are cleared at this time, all words have been accessed for the memory reference in progress and MAC1 RQ EN (asserted at start-up) will go to 0. The controller then uses this signal to terminate. If RQ EN is still asserted after END WR occurs, there are more words to access and the controller restarts. Operation is as follows.

With END WR set, STATE CLEAR sets and BUSY clears on the next MAC5 BUSI CLK. STATE CLEAR gates the address counter contents to update MAC4 LATCH 34 and 35 (Subsection 3.1.2) in case of a restart and it asserts MAC3 CLR to clear BUS DONE and the ACKN control flip-flops. It also asserts MAC3 CLR A to clear loop-around mode (MAC6 LOOP AR) if the controller is to terminate (RQ EN = 0) and the operation is read (RD RQ = 1). BUSY going to 0 removes the set input to R-S flip-flop MAC3 ADR LATCH. The latching input is also disabled by RQ EN = 0 if the controller is to terminate. ADR LATCH then clears to unlatch the address and request latch circuits in preparation for the next memory reference. ADR LATCH going to 0 also causes MAC3 CYC CONT BUSY to clear on the next MAC5 BUSI CLK. This terminates controller operation. If the controller is to restart (RQ EN = 1), ADR LATCH remains latched when BUSY is cleared, causing flip-flop MAC3 NEXT to set on the next MAC5 BUSI CLK. NEXT restarts the controller, as explained in Subsection 3.1.3. NEXT remains set until BUSY goes true again at the start of the next core cycle.

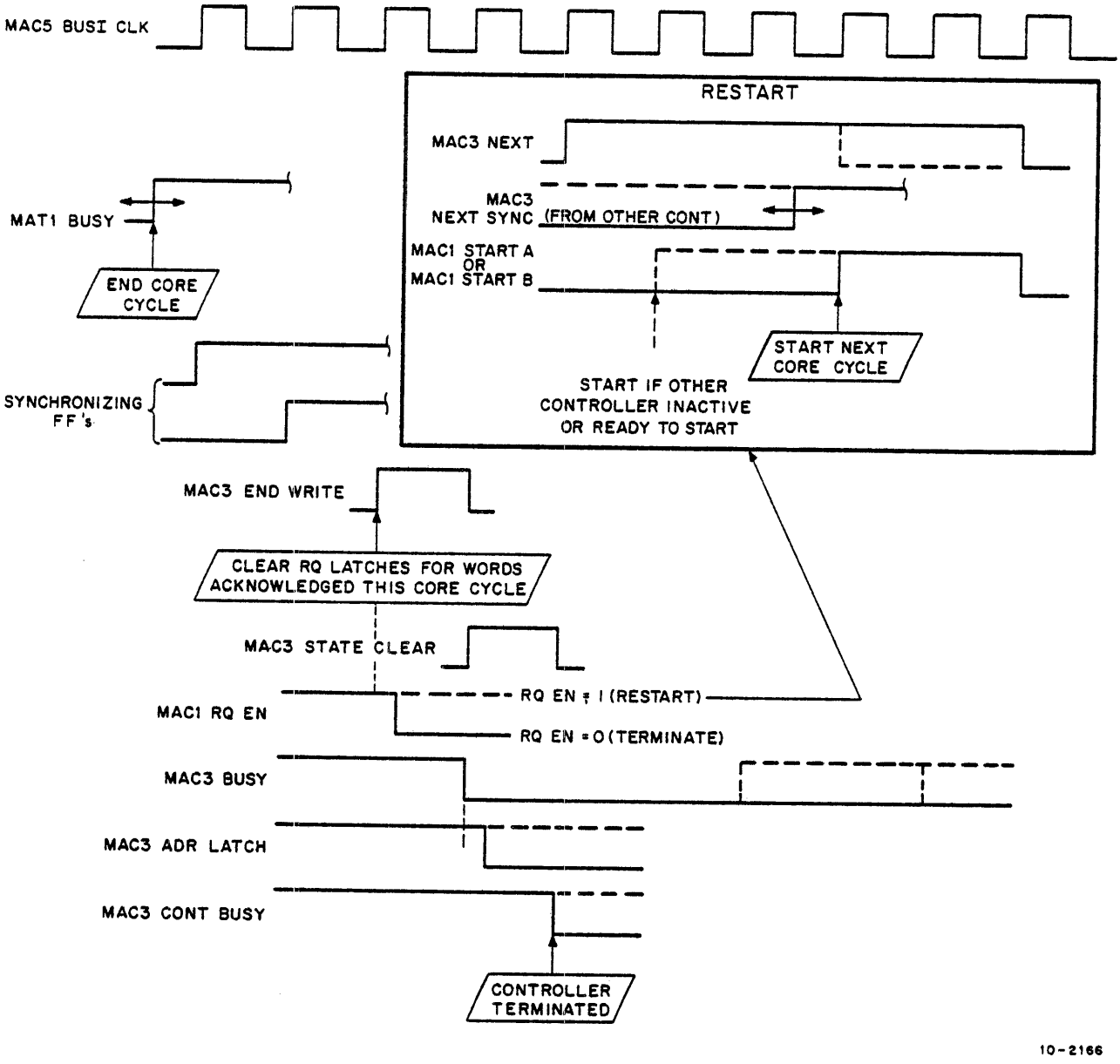


Figure 3-6 Termination and Restart Timing Diagram

3.1.8 Error Logic

Error checking in the MB20 system is described in Subsection 1.2.8. Error conditions checked in the MB20 controller are:

1. Incorrect address/request line parity
2. Incomplete memory reference

Correct address/request line parity is odd. MAC4 14–17, MAC4 LATCH 18–35, MAC3 RQ 0–3, MAC4 RD RQ, MAC4 WR RQ, and MAC4 PAR connect to a 4-element parity checker with output MAC4 PAR ERR. If incorrect (even) parity is detected at the start of a memory reference (MAC3 START A/B = 1 ∧ MAC3 ADR LATCH DLY = 0), PAR ERR causes flip-flop MAC6 ADR PAR ERR to be set by the phase A clock. MAC6 ADR PAR ERR then asserts SBUS ADR PAR ERR to flag the error in the MBox. When the next phase A clock occurs, MAC6 ADR PAR ERR clears (input cutoff when START A/B goes false) to give an SBus error signal duration of one SBus clock period.

The negation of MAC3 ADR LATCH DLY at the input to MAC6 ADR PAR ERR prevents the flip-flop from being set by parity network glitching during a restart. (Parity network inputs, RQ 0–3 and ADR LATCH 34 and 35 are modified at the end of a core cycle as explained in Subsection 3.1.7.) MAC3 ADR LATCH DLY sets on the first MAC5 BUSI CLK following the assertion of MAC3 ADR LATCH and it remains set for the entire MB20 operation.

In addition to asserting the SBus error signal, MAC6 ADR PAR ERR also enables MAC6 ADR PAR ERR FLAG. This flip-flop, which sets and latches on the next phase A clock, is the error status bit transmitted to the CPU over the SBus during the SBus diagnostic cycle (Table 2-2, Function 0).

An address parity error condition inhibits R/W currents in the referenced SM(s) as explained in Subsection 3.1.6. Signal MAC4 PAR ERR LATCH, the latched value of MAC4 PAR ERR, is used for this purpose. It remains asserted for the entire MB20 operation.

An incomplete memory reference error is detected by a 6-stage binary counter. MAC6 INC RQ CT, the AND of the all 6 counter stages, is asserted when the count equals 64 during MB20 operation. (The counter is held cleared whenever the MB20 controller is inactive by the negation of MAC3 CONT BUSY at the ENABLE input.) Because a controller is never active for 64 SBus clock periods during normal operation, the assertion of INC RQ CT indicates a malfunction. Specifically, it indicates that the controller is in a hung condition, either because of a controller failure, or because a signal has not been received on the SBus.

When an incomplete memory reference has been detected, MAC6 INC RQ CT causes flip-flop MAC6 INC RQ to be set by the next phase A clock. INC RQ then asserts MAT2 INITIALIZE and MAT2 INITIALIZE A to return all flip-flops in the timing module to their initial state and to reset the control module by generating MAC3 CLR A, MAC3 PWR CLR, and MAC3 PWR CLR1. The MB20 is then ready to perform another SBus operation. INC RQ also asserts the SBUS ERROR line to flag the error in the MBox.

As for an address parity error, an incomplete memory request sets a status indicator that is read during the SBus diagnostic cycle (Table 2-2, Function 0). MAC6 INC RQ CT enables the status flip-flop, MAC6 INC RQ FLAG, to set and latch on the next phase A clock.

Error indicators are cleared by means of the SBus diagnostic cycle (Table 2-2, Function 0). SBus data line D05 sets MAC6 CLR ERR to direct-clear both MAC6 ADR PAR ERR FLAG and MAC6 INC RQ FLAG. The flags are not cleared by SBUS CROBAR or SBUS MEM RESET.

3.1.9 Margin Control

Storage module operation can be tested under margin control. Four control bits (three turn-on bits and a direction/clear bit) are asserted during the SBus diagnostic cycle to allow either "high over normal" or "low under normal" values of X-Y select current, sense amplifier threshold voltage, or sense strobe timing. The control bits are stored in control flip-flops MAC6 CUR MARGIN, MAC6 VTH MARGIN, MAC6 STRB MARGIN, and MAC6 MAR DIRECTION. The flip-flops are clocked by MAC6 LOAD MARG on the third phase A clock of the diagnostic cycle. They are loaded from SBus data lines D27–30. (Control bit definitions are listed in Table 2-2, Function 1.) LOAD MARG is asserted when one of the three turn-on bits is asserted; that is, when MAC6 BITS 27V28V29 = 1.

The fourth control bit, which corresponds to data line D30, has a dual function. When no margins are to be turned on (MAC6 MAR OFF = 1), it generates MAC6 CLR MAR to direct-clear all four margin control flip-flops. When a turn-on bit is asserted, it loads control flip-flop MAC6 MAR DIRECTION to serve as a high or low margin indicator.

The four margin control flip-flops are used to generate the necessary control signals to margin the SMs. The single control line MAT2 STRB MARG determines the timing for the sense strobe. In an SM, the voltage on the line biases a transistor in a one-shot's external timing circuit to move the leading edge of the strobe ±15 ns. In the controller, the signal is generated by a NAND gate and an AND gate with outputs that are connected together through a diode. When the strobe margin turn-on bit is off, neither gate is enabled, the diode is reverse-biased, and MAT2 STRB MARG is approximately 2.0 V. When strobe margin is on and low margin is specified (MAC6 MAR DIRECTION = 0), only the NAND gate is enabled. This grounds the control line and causes an early strobe. If a high margin is specified (MAC6 MAR DIRECTION = 1), only the AND gate is enabled and MAT2 STRB MARG goes to approximately +4 V to cause a late strobe.

Two control lines are necessary to margin the SM X-Y select currents. For high current margin, a gate enables transistor Q4 to ground MAT2 HI CUR MARG which increases the amplitude of the bias current generator output in an SM. The increase in bias current increases the select currents by approximately six percent. Similarly, MAT2 LOW CUR MARG is grounded by transistor Q5 when low current margin is specified. This reduces the bias current, decreasing the select currents by approximately six percent.

Control line MAT2 VTH MARGIN connects to voltage dividers in the SM sense/inhibit modules that provide the positive threshold voltage for the sense amplifiers. The control bits to low-margin the threshold voltage enable a gate which turns on transistor Q3. This connects the control line to ground through a resistance in the transistor's emitter circuit. The resistance (four 1780-ohm resistors in parallel, with each resistor having a separate ground connection) shunts the sense amplifier voltage dividers and reduces the threshold voltage. Each of the four ground connections (MAT2 VTH GND 0–3) is made in one of the G236 driver modules associated with an SM, causing the shunt resistance to decrease (more parallel resistors) as more SMs are connected. The result is to make the margin circuit self-compensating; that is, the changing resistance maintains the same margin voltage, independent of system configuration. Transistor Q2 is turned on to high-margin the threshold voltage. This places a positive voltage (~3 V) on MAT2 VTH MARGIN to increase the threshold voltage. The voltage is supplied by a voltage divider on the timing module.

3.1.10 Controller Reset Logic

The MB20 is reset by SBUS CROBAR when the system is powered up or down, and by SBUS MEM RESET, which is generated by a diagnostic function via the console processor.

Both SBUS CROBAR and SBUS MEM RESET assert MAT2 CLR SWITCHES. CLR SWITCHES clears the address boundary registers, the RQ EN flip-flops, and BIT 06 and 07 (the stored interleave mode). CLR SWITCHES also generates MAT 2 INITIALIZE and MAT2 INITIALIZE A.

INITIALIZE A, which is also asserted when the controller detects an incomplete memory request (Subsection 3.1.8), aborts any SM core cycles that are in progress by returning all timing module flip-flops to their initial state. INITIALIZE resets the control module by asserting MAC3 CLR A, MAC3 PWR CLR, and MAC3 PWR CLR1.

The primary function of CLR A, also asserted at the end of a read or read-modify-write operation (RD RQ = 1∧ STATE CLEAR = 1∧ RQ EN = 0), is to clear MAC6 LOOP AR. It also clears four diagnostic control flip-flops on the same IC.

PWR CLR1 asserts MAC3 CLR, which is also asserted by STATE CLEAR at the end of a core cycle, to clear BUSY, BUS DONE, and the flip-flops in the ACKN control logic. The remaining control module flip-flops, except for the error flags, are cleared directly by PWR CLR1 and PWR CLR. The two error flags (MAC6 ADR PAR ERR and MAC6 INC RQ FLAG) are cleared only during the SBus diagnostic cycle, as explained in Subsection 3.1.8.

3.2 STORAGE MODULE

The MB20 storage module consists of two 32K × 19-bit core memory sections. Each section consists of an H224-B stack module, a G236 driver module, and a G116 sense/inhibit module. One section uses only 18 bits of the total 19-bit capacity to store bits 00–17 of the data word; the other section uses all 19 bits to store bits 18–35 plus the parity bit. A block diagram for one storage module section is shown in Figure 3-7.

A circuit-level description of storage module operation follows. Reference should be made to the block diagram and to the module circuit schematics in the Field Maintenance Print Set.

3.2.1 Stack Select

Table 3-5 lists the four select levels connecting to each of the storage modules associated with a controller. The select levels are asserted by the controller as explained in Subsection 3.1.2.

When an SM is referenced, the four asserted select lines cause DRVC STACK SEL to be asserted in the SM's two G236 driver modules. STACK SEL enables the SM's X-Y current generators, inhibit drivers, and address decoders. (Enabling the decoders causes the core address to select the X-Y drivers and switches.) The enabled circuits allow R/W currents to be generated, causing a core cycle in the addressed SM, when control and timing signals are initiated by the controller's timing module.

3.2.2 Address Decoders

To select a word in memory, the 15 bits of core address from the controller (MAC4 ADR 21–23 and MAC1 ADR 34–45) are decoded in each of the SM's two G236 driver modules as follows:

1. ADR 23, 24, 25, 26 select 1 of 16 X-R/W switches
2. ADR 21, 27, 28, 29 select 1 of 16 X-R/W drivers
3. ADR 22, 30, 31, 32 select 1 of 16 Y-R/W switches
4. ADR 33, 34, 35 select 1 of 8 Y-R/W drivers.

The decoding is illustrated in Figure 3-8. The correspondence between the core address and the controller's address latches is also shown.

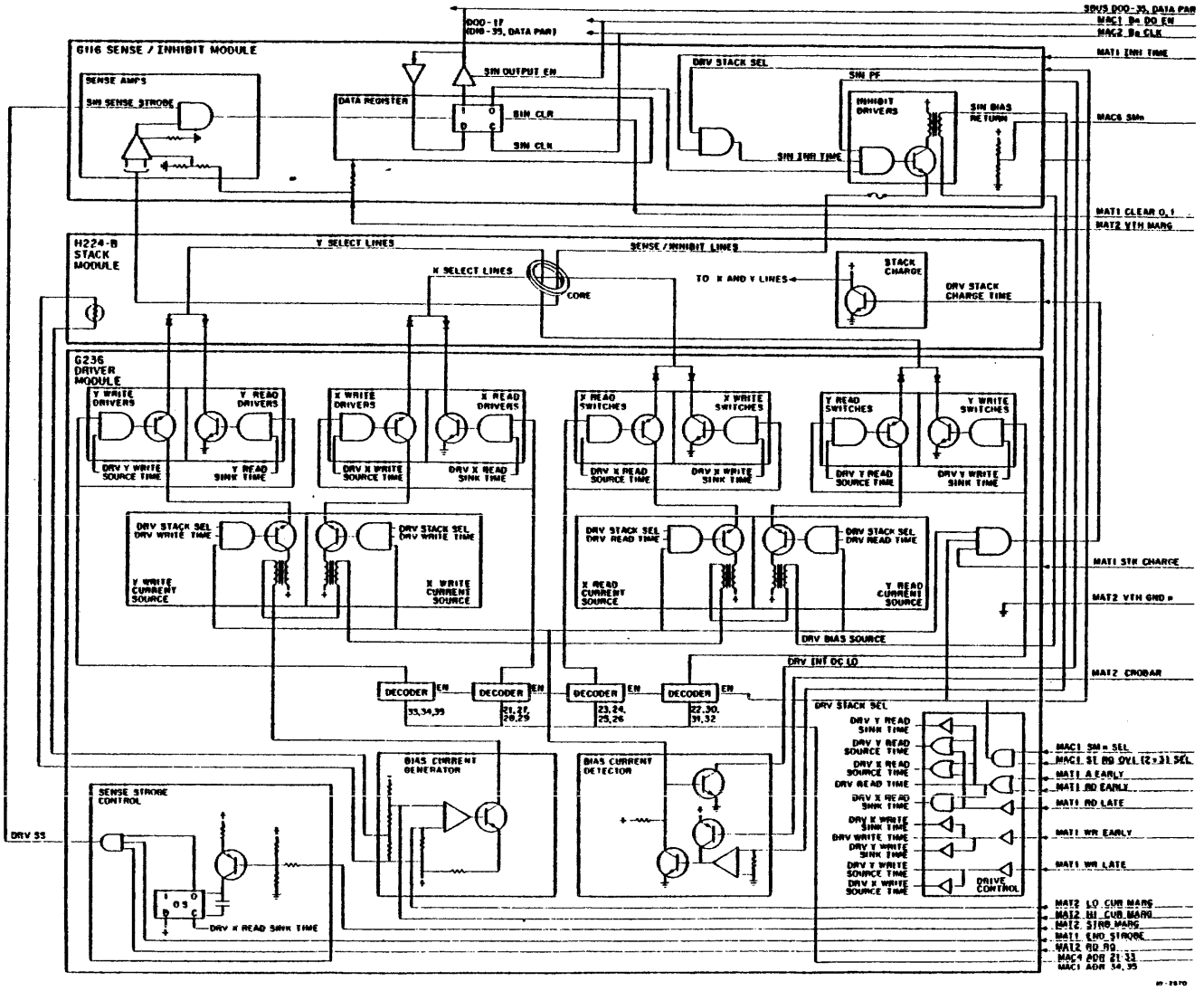


Figure 3-7 MB20 Storage Module Section, Detailed Block Diagram

Table 3-5 SM Select Levels

Storage Module	Select Levels to Assert DRV STACK SEL
SM0	MAC1 SM0 SEL MAC1 ST RQ 0V1 SEL
SM1	MAC1 SM1 SEL MAC1 ST RQ 2V3 SEL
SM2	MAC1 SM2 SEL MAC1 ST RQ 0V1 SEL
SM3	MAC1 SM3 SEL MAC1 ST RQ 2V3 SEL

NOTE: DRV STACK SEL also conditioned by MAT2 SPECIAL +3 V 1 and MAT2 SPECIAL +3 V 2 in all storage modules (SM0–3).

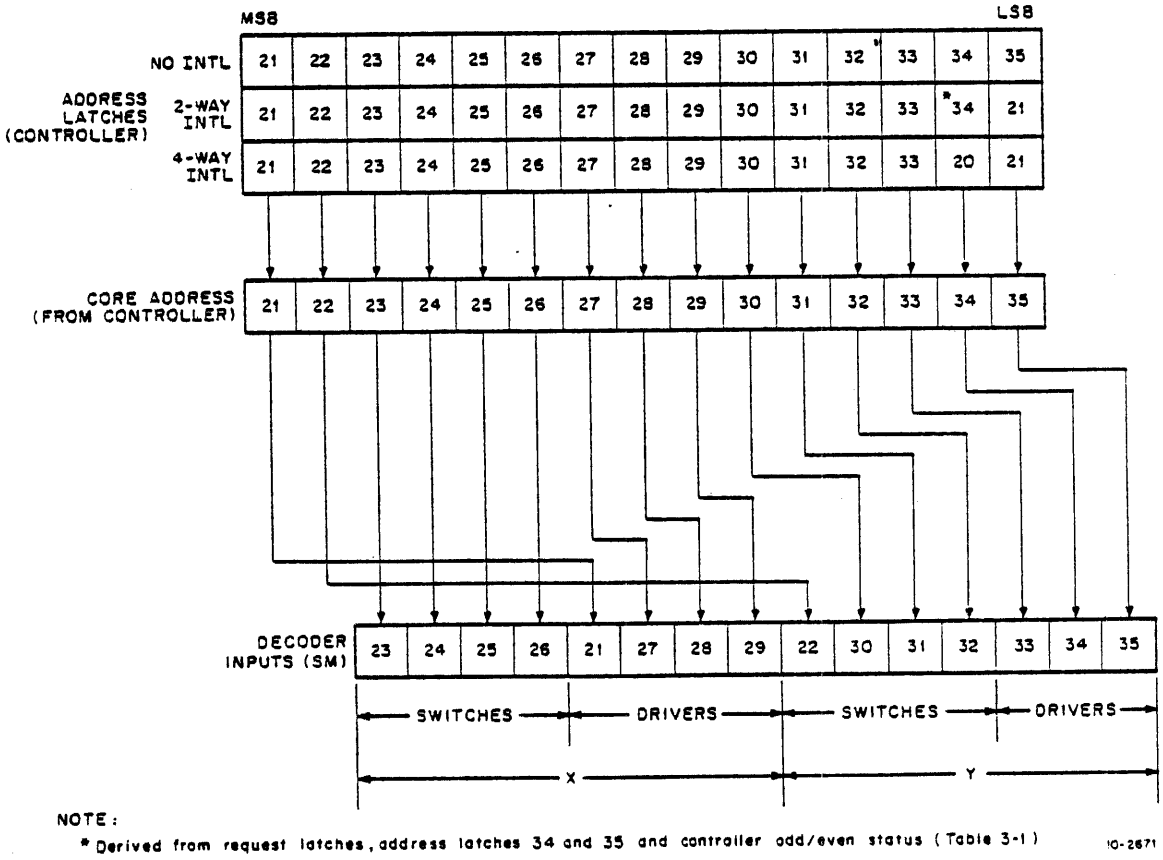


Figure 3-8 Switch and Driver Selection

The basic decoder units in each driver module are three 4-to-16-line decoders to select the switches and X drivers and one 4-to-10-line decoder to select the Y drivers. Each decoder asserts a single output corresponding to the binary value of the address bits connected to the weighted inputs (8, 4, 2, 1). For example, -ADR 23–26 equal to a binary value of 7 (0111) asserts DRV 07 SEL to select switch DRV XS07 in the X selection matrix.

The address decoder outputs are asserted only when the SM has been selected by the controller, that is, when DRV STACK SEL = 1. STACK SEL enables the 4-to-16-line decoders directly by asserting both of the decoder's AND EN inputs. Although the 4-to-10-line decoder has no enable input, the negation of STACK SEL connects to the highest-order input. (Only three of the four inputs are needed to decode the three core address bits.) Thus, the eight decoder outputs that are used (0–7) are not asserted until STACK SEL goes true.

3.2.3 X-Y Drive Control

Control signals are generated in each G236 driver module to control the read/write drivers, switches, and current generators associated with word selection. A timing diagram is shown in Figure 3-9. The signals are derived from control and timing signals asserted by the controller's timing module (Subsection 3.1.6).

DRVC READ TIME is the first SM drive control signal generated during the core read cycle. It is derived from the OR of controller signals A EARLY and RD EARLY. A EARLY is used to assert READ TIME; the trailing edge of RD EARLY negates it. READ TIME enables the X-Y read current generators, and it asserts DRVC Y READ SINK TIME to enable the Y read drivers. It also asserts DRVC X READ SOURCE TIME and DRVC SOURCE TIME to enable the X-Y switches. With all Y select circuits activated, Y read current flows as long as READ TIME is true.

RD LATE is asserted by the controller approximately 125 ns after RD EARLY switches Y line current. RD LATE is ANDed with READ TIME to generate DRVC X READ SINK TIME. This signal turns on X read current by enabling the X read drivers, the last of the X select circuits to be activated. With both X and Y currents switched, a word is read from the core stack. X and Y read currents remain on until the trailing edge of RD EARLY negates READ TIME, which disables both the X and Y read current generators.

Controller signal WR EARLY starts the core write cycle. It asserts DRVC WRITE TIME, which enables the X-Y write current generators, and it asserts DRVC X WRITE SINK TIME and DRVC Y WRITE SINK TIME, which enable the X-Y switches. WR LATE from the controller then asserts both DRVC X WRITE SOURCE TIME and DRVC Y WRITE SOURCE TIME to activate the X-Y drivers and switch on both X and Y write currents. The write core cycle ends when WRITE TIME is negated by the trailing edge of WR EARLY, disabling the write current generators.

3.2.4 Drivers and Switches

The drivers and switches in an SM section form a selection matrix that directs current through the X and Y lines in the core stack. Current is switched in the proper direction as selected by the read and write operations. X-Y selection is described in Subsection 2.4.4.

Figure 3-10 shows a simplified schematic of a driver/switch combination interconnected with the core stack and the associated current generators. A Y driver/switch combination, R/W switch YS07 and R/W driver YPWD7/YNRD7, is used as an example. Assuming these circuits are selected by the address decoders (E16 and E30), the following occurs during a core cycle.

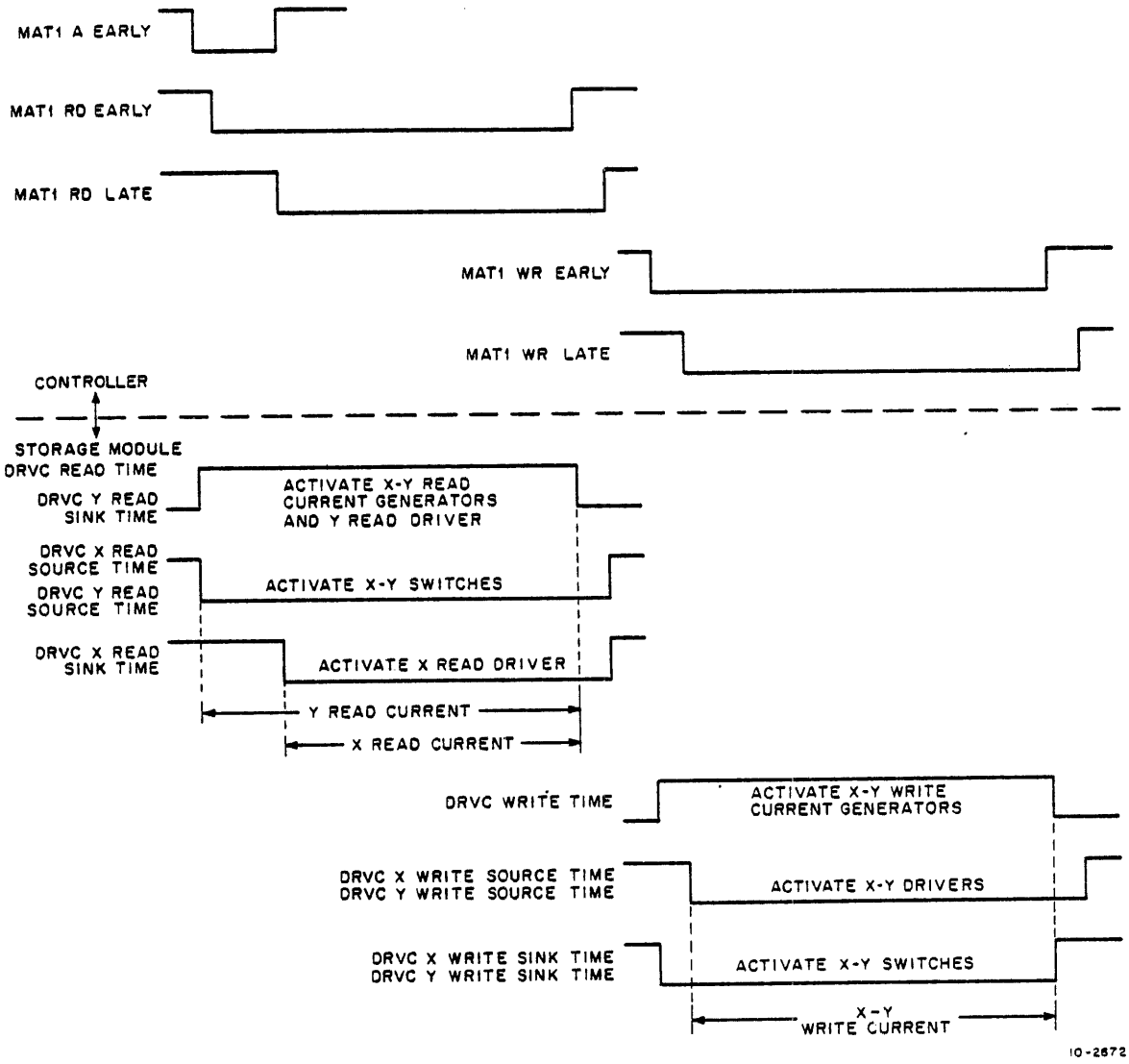


Figure 3-9 X-Y Drive Control Timing Diagram

When timing signals Y READ SINK TIME and Y READ SOURCE TIME are asserted at the start of the core read operation, negative read driver E32 and positive read switch E20 are turned on. At the same time, READ TIME also enables the Y read current generator allowing current to flow through the switch, D47, the Y line, a stack diode, and the driver to ground. READ TIME is then negated, turning off the read current by disabling the current generator. Y read current is switched on for approximately 450 ns.

To generate Y write current, timing signal WRITE TIME is asserted first to enable the Y write current generator. Signals Y WRITE SOURCE TIME and Y WRITE SINK TIME are asserted next to turn on positive write driver E31 and negative write switch E19. Y write current then flows through the driver, a stack diode, the Y line, and the switch to ground. (Note that write current direction is opposite to read current direction.) Current is switched off by the negation of WRITE TIME, which disables the write current generator and causes a Y write current duration of approximately 350 ns.

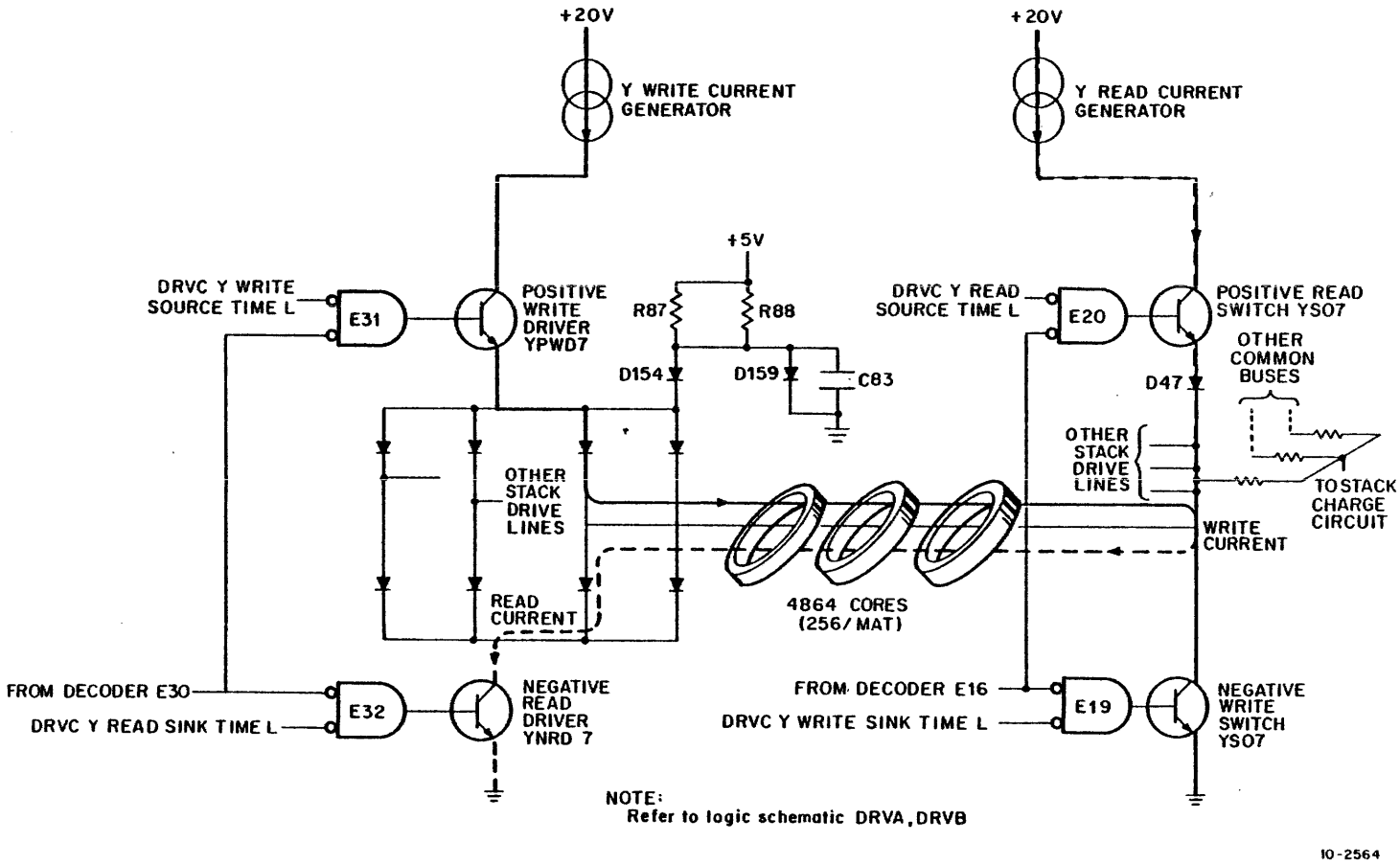


Figure 3-10 Typical Y-Line Read/Write Switches and Drivers

During the time that Y read and write currents are switched by the active Y driver/switch combination, a selected X driver/switch combination is switching read and write currents through an X line. Operation is similar except that timing signal X READ SINK TIME, which enables the X negative read drivers, is asserted after READ TIME. This causes X read current to be switched on approximately 75 ns after the start of Y read current. The staggered read currents are to minimize the delta noise generated in the stack. X write current timing is the same as for the Y axis.

3.2.5 X-Y Current Generator

Four current generators (sources) in the G236 driver module supply X read current, Y read current, X write current, and Y write current for an SM section. Current amplitude is controlled by a temperature-compensated dc bias current supply. The current generators are enabled by timing signals READ TIME and WRITE TIME which are generated by the X-Y drive control logic (Subsection 3.2.3).

Figure 3-11 shows the bias current supply and the Y write current generator. The saturating transformer T2 is normally saturated hard by bias current in the winding designated by pins 7 and 8. In order for the magnetic core of T2 to start to switch its magnetic flux in the opposite direction, the ampere turns applied by the bias current must be executed by an equal, but opposite, MMF in another winding. As long as T2 remains saturated, it is a low impedance to changes in current in any of its windings. However, once T2 starts to switch magnetic flux, large voltages may be induced across its windings in response to any additional changes in net current. That is, the saturating transformer acts like an ideal current source: low impedance with less than a specified current in winding 6–9, and a high impedance to additional current changes once the specified current amplitude is reached. The specified current is primarily determined by the bias current value and the turns ratio of the transformer. The third winding (5–10) conducts current only after the drive current pulse has ended and restores some current to the +20 V supply during that period. Although some losses occur, most of the energy absorbed by the transformer during the current pulse is restored to the power supply at the end of the pulse.

The bias current supply, which also provides dc bias current for the inhibit driver circuits on the G116 sense/inhibit module, drives all bias current windings in series. LRC filter networks are provided at intervals to ensure that the bias current does not acquire ac components and that large voltages do not build up along the series path. L1, C65, and R17 comprise such a filter network, protecting Q7 and Q8 from transients.

The resistor network in the bias current supply (consisting of a stack thermistor, R7–15, and zener diode D1) provides a temperature-compensated reference voltage to pin 3 of operational amplifier E1. The amplifier's output (pin 6) biases Q7, which controls the bias current flowing through Q8 and the bias windings of the saturating transformers. Jumpers W5, W6, and W7 in the resistor network are cut to adjust the bias current to its optimum value. The jumpers are installed during manufacture and should not be changed in the field.

Small variations in bias current, as evidenced by changes in the emitter voltage of Q8, are fed back to pin 2 of E1 via R2. This causes the operational amplifier, which uses its gain to maintain a voltage on pin 2 nearly equal to the voltage on pin 3, to adjust its output and regulate the bias current at a value controlled by the reference voltage.

Margin lines MAT2 LOW CUR MARG and MAT2 HI CUR MARG from the controller connect to the operational amplifier input pins to provide a means of changing R/W currents in the stack. (Margin control is discussed in Subsection 3.1.9.) For high current margins, amplifier input pin 3 is grounded through R6 (470KΩ) by the margin line, causing an increase in the bias current and a corresponding increase (~six percent) in R/W currents. Similarly, for low current margins, grounding input pin 2 through R1 (470KΩ) lowers the bias current causing the R/W currents to decrease (~six percent).

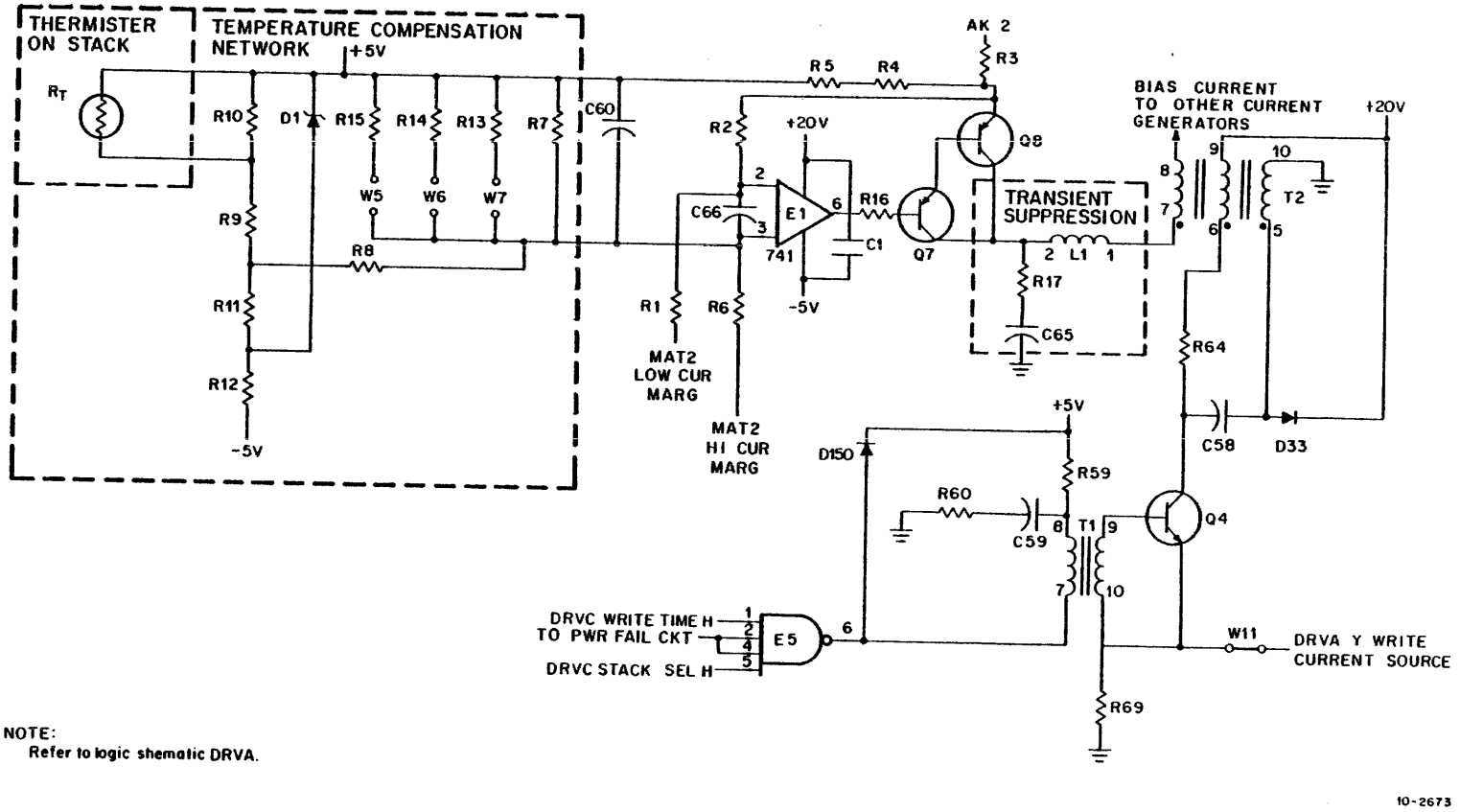


Figure 3-11 Bias Current Supply and Y Write Current Generator

Operation of the write current generator is as follows: the output of E5 goes low when the current generator is activated. Current is then coupled through T1 to saturate transistor Q4. With a driver and switch enabled in the selection matrix providing a current path, current flows through windings 6–9, Q4, and out to the write drivers. When Q4 is turned off by E5 (coupling through T1), current flows through D33 from winding 5–10 until the core of T3 has been completely resaturated by the bias current. This places energy back in the power supply.

The read current generators have additional components to control the leading edge of read current. With reference to the G236 circuit schematic, diodes D7 and D8 and resistor R49 form an anti-overshoot circuit that connects from current generator output DRV X READ CURRENT SOURCE to ground. The circuit conducts to "steal" some of the read current during the rise time interval, allowing the current to rise to its proper value and not beyond it. A similar circuit is used in the Y read current generator.

In addition to the anti-overshoot circuit in the X read current generator, diodes D17 and D16 are used to limit the voltage applied to the X read switches, thus making the X read current rise time less dependent on the accuracy of the 20 V power supply. This is required because the core output signals are affected more by X read current than by Y read current. (Y read current flows before the X read current, hence the X read current does the actual core switching.)

3.2.6 Stack Charge Circuit

The stack charge circuit assists the stack capacitance in recovering and shortens the rise time of the stack current. It also reduces unwanted currents in the unselected lines associated with the selected driver. It is located on the H224-B stack module.

Figure 3-12 shows the stack charge circuit. Its output is taken from the emitter of transistor Q1 and goes to the junction of each X and Y read/write switch pair via a resistor. This common interconnection is labeled V_0 in the figure. It is desired that V_0 be approximately 0 V (ground) during a read operation, and approximately +20 V during a write operation. The effective stack capacitance associated with each line is shown as C_{STACK}.

During a read operation, the DRVA STK CHARGE TIME signal is low, making the output of E1 (pin 6 and 8) high, thus saturating Q2 and Q3, and turning off Q1 and Q4. The output voltages of the circuit are also held low by the parallel combination of L1 and D123, and also L2 and D122. A current thus flows from +5 V, through R35 and R36, through L1 and L2, through R37 and R38 and through Q2 and Q3, to ground. Q1 and Q4 are off since their base-emitter junctions are not forward-biased.

During a write operation, the DRVA STK CHARGE signal goes high, making the output of E1 (pin 6 and 8) go low, thus turning off Q2 and Q3. Current that was flowing through L1 and L2 is forced to continue to flow, but now must flow into the base of Q1 and Q4. Hence, with Q1 and Q4 turned on (saturated) and Q2 and Q3 off, the output is equal to 20 V less $V_{CE\ SAT}$ of Q1 or Q4. Current spiking from Q1 and Q4 on the transitions is prevented by D122 and D123. When Q2 and Q3 turn on again, Q1 and Q4 must be fully off before current can flow through D122 and D123. This is because if D122 and D123 are forward-biased, the base-emitter junction of Q1 and Q4 are reversed-biased.

3.2.7 Inhibit Drivers

Inhibit driver circuitry is illustrated in Figure 3-13. The driver pair for data bit 11 is shown. It is typical of all 19 inhibit driver pairs on the G116 sense/inhibit module.

Only one of the inhibit drivers in the inhibit driver pair is selected during a core write cycle; that is, either SINA INH 1 or SINA INH 2 is asserted at the input (pin 2 or 11) to E46, depending on address bit 34 (SINA A34) of the core address. Figure 3-8 shows the correspondence between the core address bit and the SBus address for the three interleave modes.

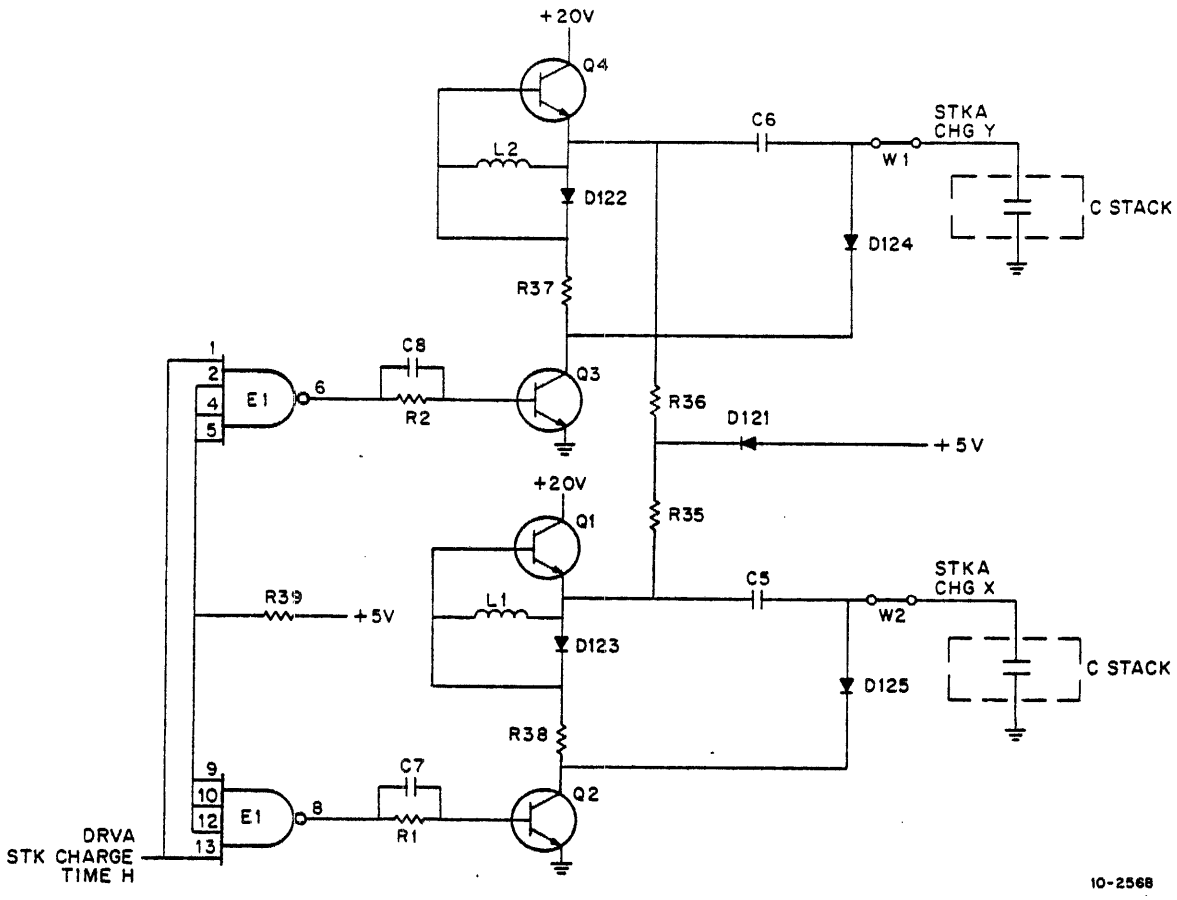


Figure 3-12 Stack Charge Circuit

When the SINA INH signal is asserted and the associated data register flip-flop is cleared; that is, when a 0 is written in bit 11 of the selected word, the output of E46 (pin 1 or 13) goes high, driving current through the primary winding of E47 to which the selected gate is connected. The secondary winding of that transformer provides the current to turn on Q27 or Q28. When turned on, current flows from the +20 V power supply, through fuse F14, saturating transformer E45, Q27 or Q28, isolation diodes, the sense/inhibit winding, and the diodes (D92 and D93 or D90 and D91) to ground. The value for the inhibit current is primarily determined by the bias current to transformer E45 (winding 3-12). Bias current generation and the operation of saturating transformers is discussed in Subsection 3.2.5.

Each leg of the sense/inhibit sees half inhibit current; approximate 370 mA. Capacitor C125, diodes D88 and D89, and resistor R14 help reduce the power dissipated in Q27 or Q28 during turn-off.

Capacitors C123 and C124 allow the gate to pump reverse current into the transformer primary; they also help to decrease the turn-on time of Q27 or Q28.

3.2.8 Sense Amplifiers

Sense amplifiers are used to detect 1 outputs from the core stack. Figure 3-13 shows the sense amplifier circuit for data bit 11 interconnected with the associated sense/inhibit windings and inhibit driver pair. The circuit is typical of all the sense amplifier circuits on the G116 sense/inhibit module. It consists of the sense amplifier IC, terminating capacitors for the sense/inhibit windings, and a threshold voltage network.

There are two sense amplifiers per data bit with each sense amplifier input connecting to 16K cores. During a core read operation, the inhibit driver connection is an open circuit through driver transistor Q27 or Q28. The effect of the inhibit driver circuits and isolation diodes D92 and D93 or D90 and D91 can be ignored during the read operation because the diodes are reversed-biased.

Sense amplifier E50 (type 7520), being a dual IC package, connects to both sense/inhibit lines of one mat. Each interior circuit consists of a preamplifier and sense amplifier. The inputs to the internal sense amplifiers are available to facilitate accurate strobe timing. Both circuits share a reference voltage (or threshold voltage) amplifier (pins 4 and 5). In this application, pin 4 is grounded through resistor R182 and a positive threshold voltage of approximately 17 mV is supplied to pin 5.

Margin control line MAT2 VTH MARGIN from the controller connects to all the threshold voltage networks in the sense/inhibit module. It also connects to the +5 V power supply (through a resistor) in the sense/inhibit module (pin FR1), thus providing the positive voltage that is divided to establish the threshold voltage. For low-margin operation, VTH MARGIN is grounded through a resistance in the controller to shunt the threshold voltage networks and lower the voltage ~7.5 percent. For high margin operation, the controller connects VTH MARGIN to +3 V to raise the threshold voltage ~7.5 percent. Although switched by the controller, the ground (for low margin) originates in the SMs. This is to keep the low margin voltage constant, no matter how many SMs are in the system. That is, each of the four ground connections, MAT2 VTH GND n (n = 0–3), connects from the controller to one of the G236 driver modules (pin AM1) in each SM. A line is grounded, changing the shunt resistance and maintaining a constant margin voltage, whenever an additional SM is installed; that is, when the associated G236 module is plugged in.

3.2.9 Sense Strobe Control

The outputs of the sense amplifiers in the G116 sense/inhibit module are controlled by SINA SENSE STROBE. Similar to SINA INH 1 and 2, either SENSE STROBE 0 (and 2) or SENSE STROBE 1 (and 3) is asserted, depending on the state of core address bit 34. The strobes connect to the sense amplifier IC (Figure 3-13) to cause the sense amplifier's output to be asserted for the duration of the strobe signal when a 1 is to be read from core.

SENSE STROBE is generated by DRVA SS in the G236 driver module. The leading edge of DRVA SS is controlled by the output of a one-shot (~175 ns duration) that is triggered by DRVC X READ SINK TIME. (This signal activates the X read current, which causes the cores to switch and core outputs to appear at the sense amplifier's inputs.) The trailing edge of the strobe is controlled by controller signal MAT1 END STRB. The two signals, END STRB and the 0 output of the one-shot, are ANDed to assert DRVA SS 0 and 1 (and thus SINA SENSE STROBE 0 and 1) when the 1 output from core is at or near its peak amplitude. Timing is shown in Figure 3-14.

The sense amplifiers are strobed only during the SBus read or RMW operations. This is controlled by level MAT2 RD RQ A(1) from the controller which is true during these operations. It is gated with END STRB and the one-shot's output at the input to DRVA SS 0 and 1. During the SBus write operation, RD RQ A(1) is false to prevent the strobes from occurring.

The duration of the one-shot in the sense strobe control is controlled by transistor Q6 and a resistor network in the one-shot's external timing circuit. Jumpers W1–W4 in the resistor network are cut to adjust the strobe's leading edge to its optimum position. This is a factory adjustment; the jumper configuration should not be changed in the field.

The sense strobe is margined by changing the voltage on margin control line MAT2 STRB MARGIN. The line, which connects to the resistor network that biases Q14, has a value of ~2.0 V during normal operation. For an early strobe (low margin), the line is grounded in the controller to decrease the duration of the one-shot by ~15 ns. For a late strobe (high margin), the line is connected to +5 V through a resistor to increase the duration of the one-shot ~15 ns.

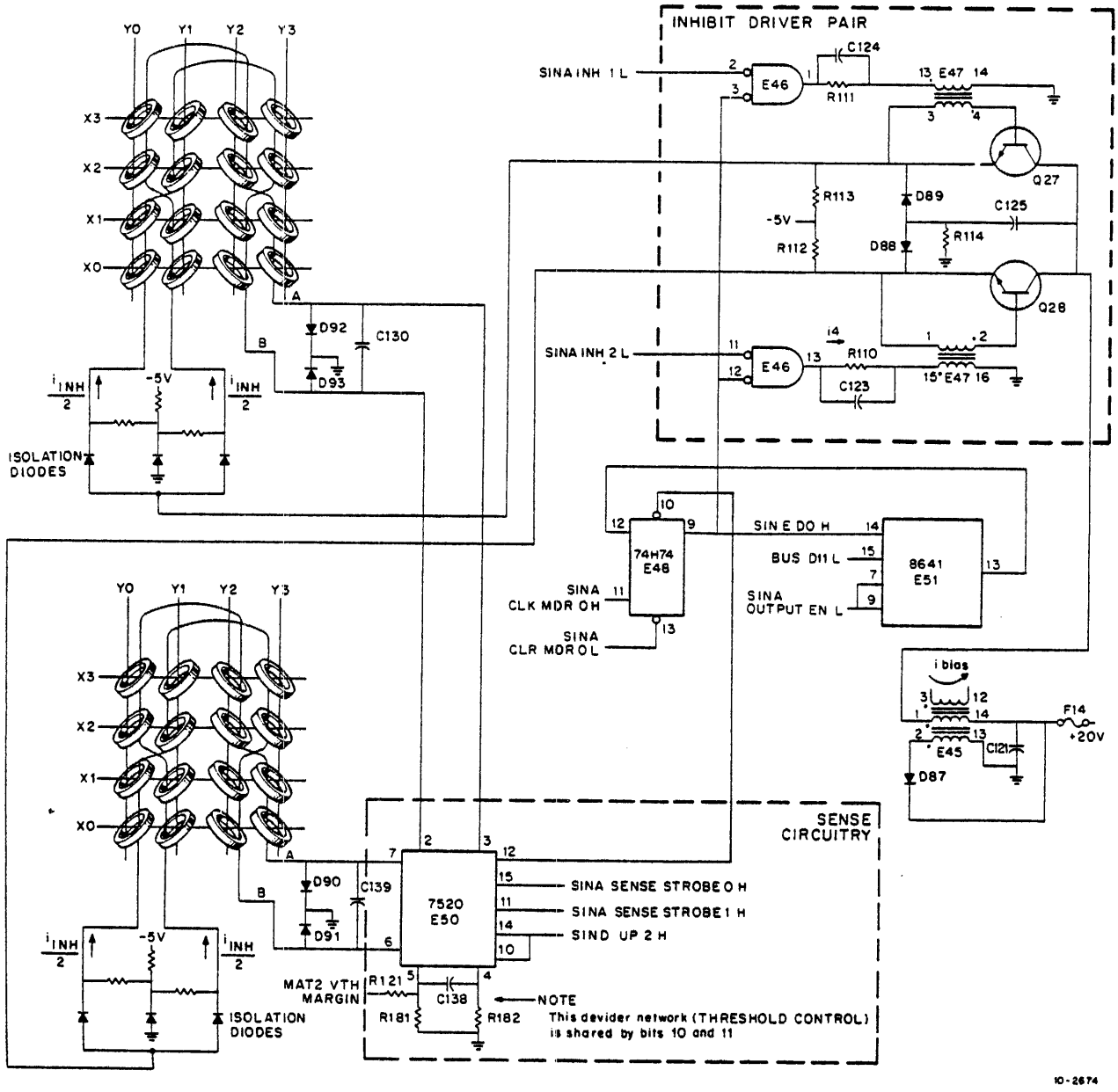


Figure 3-13 Sense Amplifier and Inhibit Driver

3.2.10 Data Register

The data register consists of 37 D-type flip-flops on the SM's G116 sense/inhibit modules. During the core read cycle (core cycle initiated by SBus read or RMW operation), the flip-flops are direct-cleared by SINA CLR MDR 0 and 1 and then direct-set by the sense amplifier outputs when 1s are read from core. (Timing is shown in Figure 3-14.) CLR MDR 0 and 1 are generated by controller signals MAT1 CLEAR 0 and 1. The strobing of the sense amplifiers to set the flip-flops is discussed in Subsection 3.2.9.

Prior to the core write cycle (core cycle initiated by the SBus write or RMW operation), the register flip-flops are clocked by SINA CLK MDR 0 and 1 to load write data from the SBus. The SBus data lines are connected to the D inputs. CLK MDR is generated by controller signal MAC2 Bn CLK (n = 0–3).

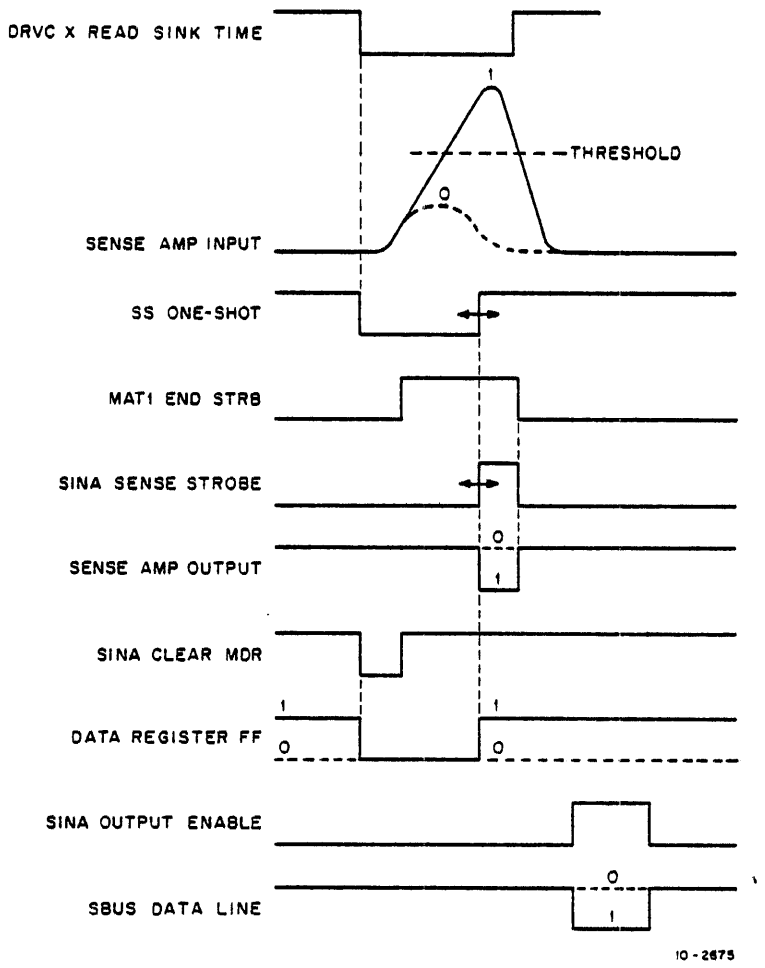


Figure 3-14 Timing Diagram for the Sense Portion of a Read Operation

3.2.11 Bias Current Detector and SM Reset Logic

Without bias current, the saturating transformers in the current generator and inhibit driver circuits do not sufficiently limit stack currents. Thus, a bias current detector circuit (in the G236 driver module) is used to disable the drive circuits if a bias current dropout should occur.

Bias current originates in the driver module, where it is applied to the current generators (in series), and outputs to the G116 sense/inhibit module on line DRVA I BIAS SOURCE. In the sense/inhibit module, it is applied to the inhibit drivers (in series) and routed back to the driver module on line SINA I BIAS RETURN. The return line connects to the bias current detector, which completes the bias current loop by providing a path to ground through a small resistance (R19). The detector circuit monitors the voltage drop across this resistance to sense a loss of bias current.

When bias current drops, a differential amplifier (E1) turns on transistor Q10. The switching of Q10 grounds an input on the AND enable gate for each current generator, thus turning off and disabling the X-Y read/write currents. Grounding the enable inputs also turns off transistor Q9. This causes DRVA INT DC LO to go high, asserting SINA PF in the sense/inhibit module. Similar to the disabling of the current generators, SINA PF connects to each inhibit driver's AND enable and turns off all inhibit currents. The inhibit currents (and X-Y read/write currents) remain disabled until bias current is restored.

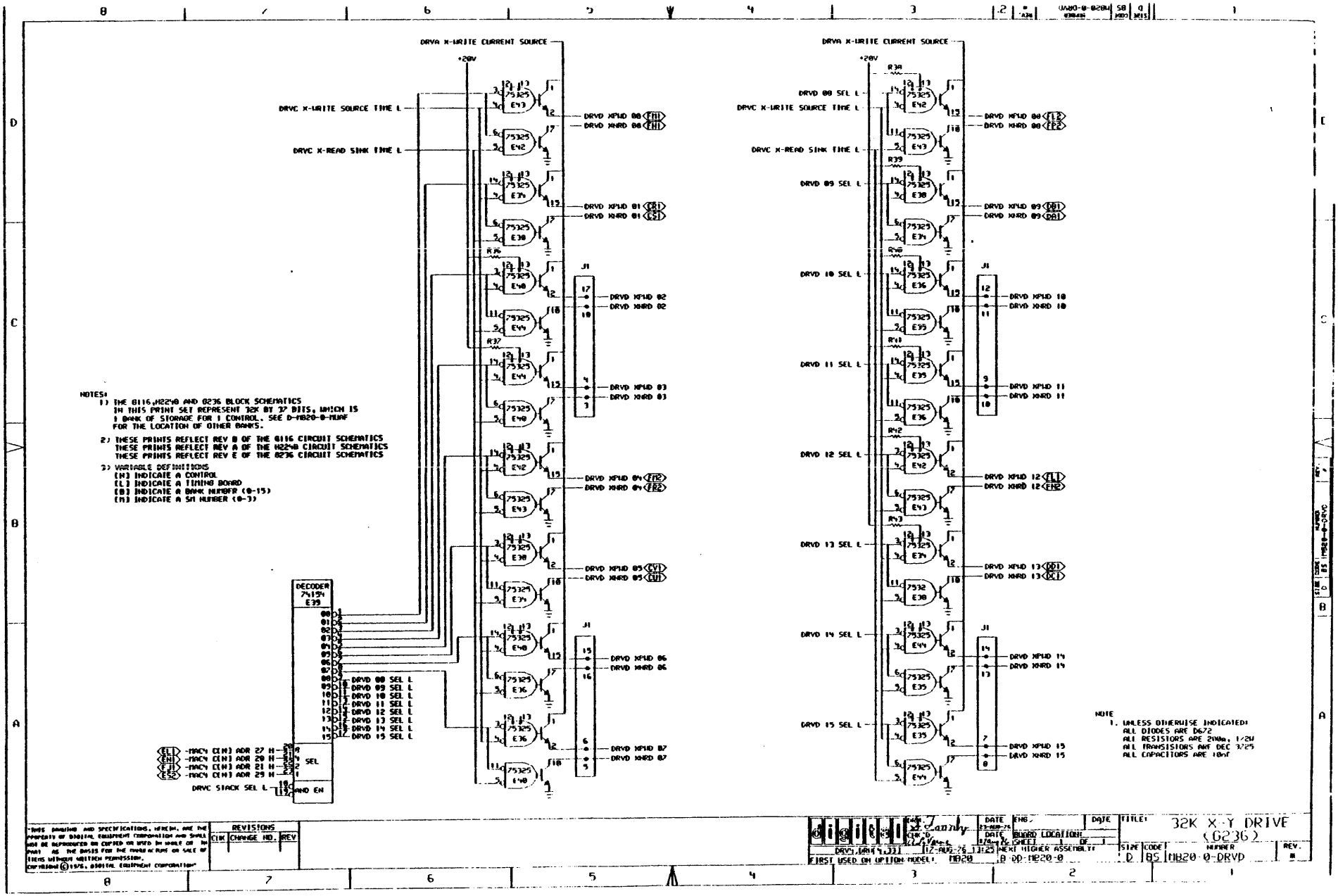
DRVA INT DC LO also prevents inhibit currents when a G236 driver module (and thus the bias current generator and detector) is removed from the system. INT DC LO goes high to assert SINA PF, just as if the bias current detector (now removed) had sensed the absence of current.

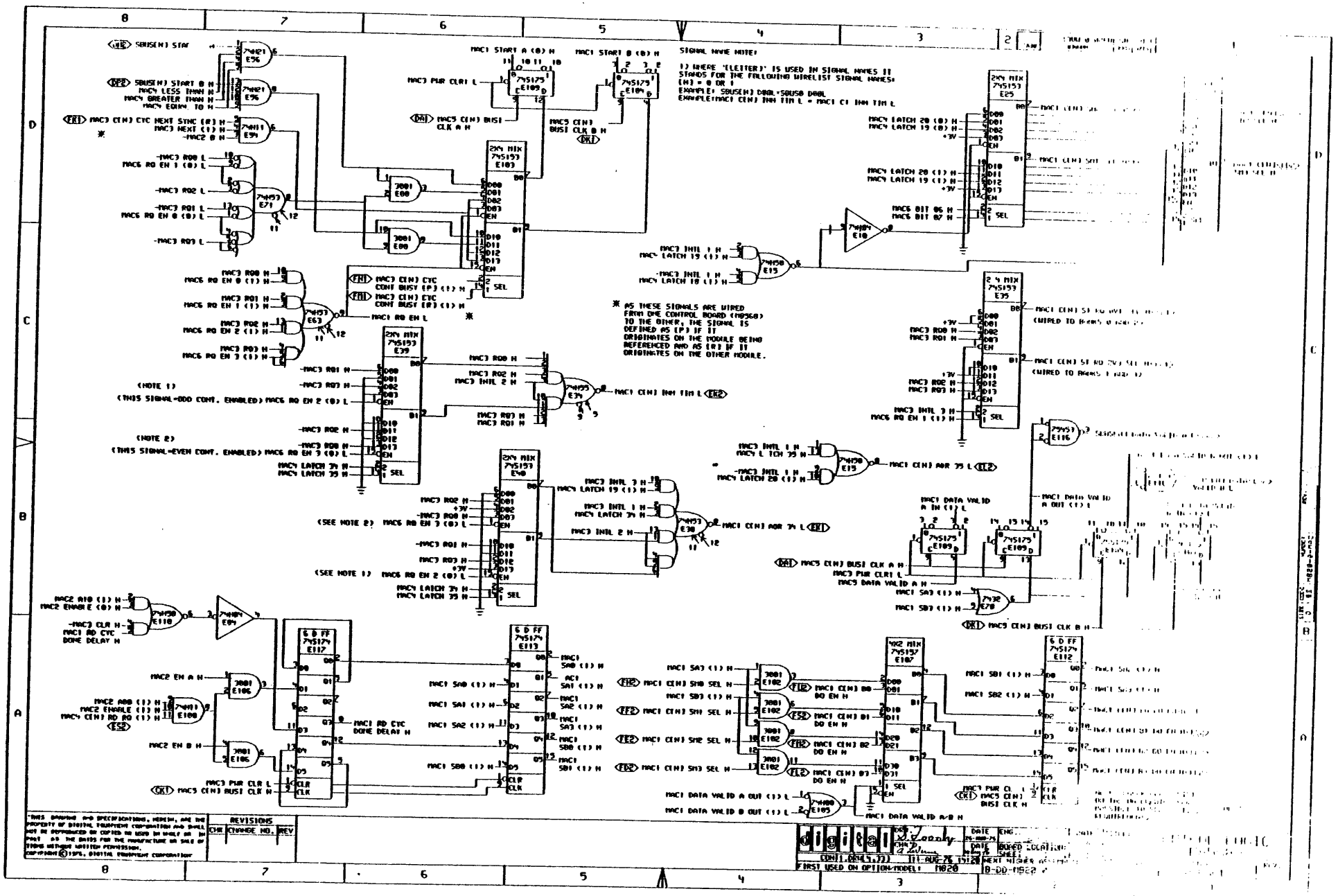
The bias current detector circuitry is also used to prevent uncontrolled stack currents from destroying data in core when the system is powered up and powered down. Controller signal MAT2 CROBAR turns on transistor Q11, which switches Q10 and inhibits the stack currents during this interval.

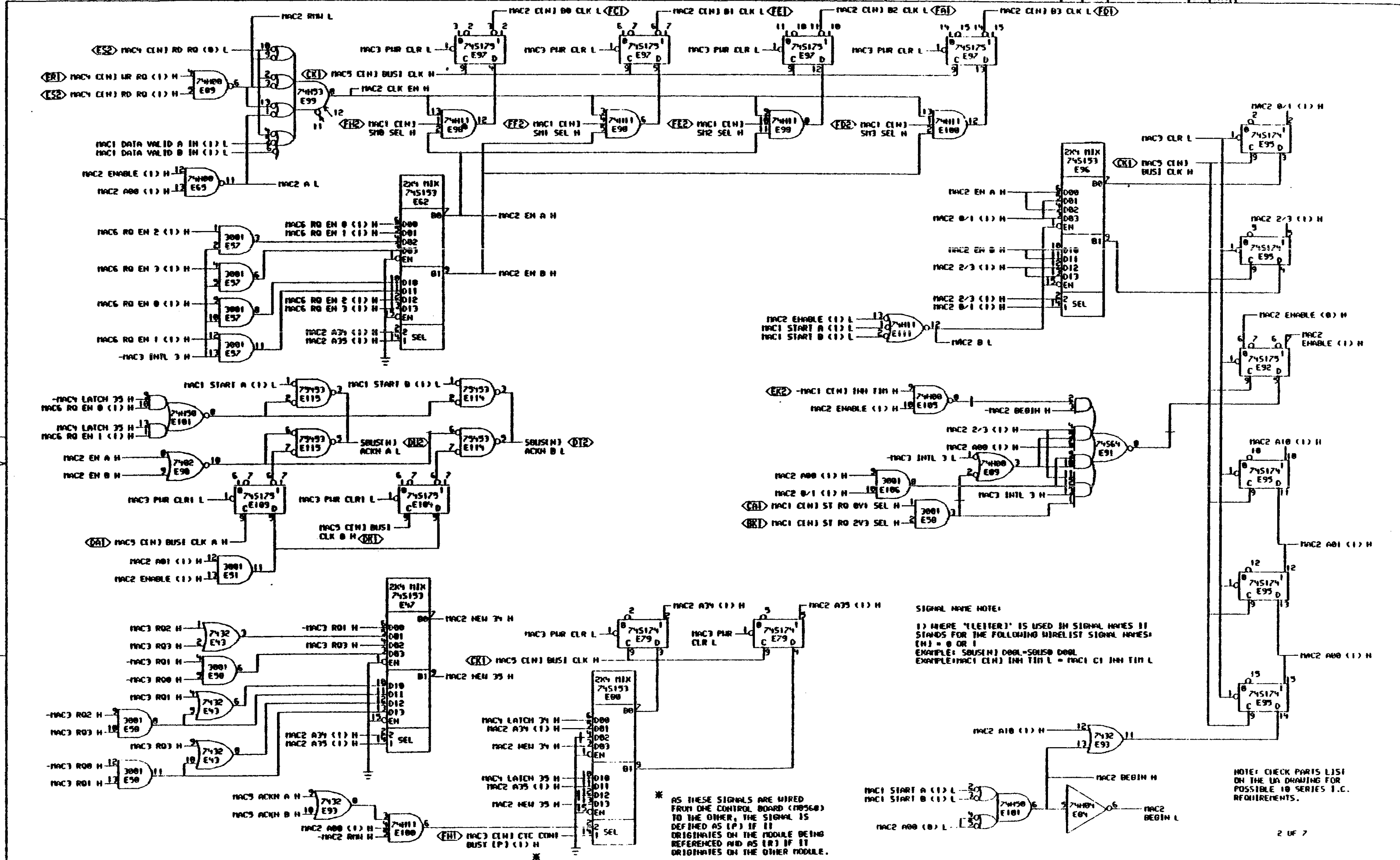
APPENDIX A
ABBREVIATIONS AND MNEMONICS

A	A/Address/Phase A
ACKN	Acknowledge
ADR	Address
AMP	Amplifier
AR	Around
B	B/Bank/Phase B
BUSI	Bus (Internal)
C	Controller
CLK	Clock
CLR	Clear
CNTR	Counter
CONT	Control/Controller
CT	Count
CUR	Current
CYC	Cycle
D	Data
DEL	Delay
DIAG	Diagnostic Cycle
DLY	Delay
DO	Data Out
DRV	Driver
EN	Enable
ERR	Error
FF	Flip-flop
FUNC	Function
GND	Ground
I	Current
IL	Interleave
INC	Incomplete
INH	Inhibit
INT	Internal
INTL	Interleave
I/O	Input/Output
LD	Load
LE	Leading Edge
LSB	Least Significant Bit
MAC	Memory Control
MAR	Margin
MARG	Margin
MAT	Memory Timing
MC	Memory Controller
MEM	Memory
MSB	Most Significant Bit
MU	Module Utilization
N	Number
OS	One-shot
PAR	Parity
PI	Priority Interrupt
POS	Positive
PWR	Power
RD	Read
RMW	Read-Modify-Write
RQ	Request
S	Starting Address
S'	Updated Starting Address
SA	Sense Amplifier/Shift Register A
SB	Shift Register B
SBUS	Storage Bus
SEL	Select
SIN	Sense Inhibit
SM	Storage Module
SS	Sense Strobe
ST	Start
STK	Stack
STRB	Strobe
SW	Switch
SYNC	Synchronize
TE	Trailing Edge
THRESH	Threshold
TIM	Time/Timing
V	Or/Voltage
VTH	Voltage Threshold
WR	Write
X	X Line
XNRD	X Line Negative Read Driver
XPWD	X Line Positive Write Driver
XS	X Line Selection Switch
Y	Y Line
YNRD	Y Line Negative Read Driver
YPWD	Y Line Positive Write Driver
YS	Y Line Selection Switch

INDEX


DRVA X-WRITE CURRENT SOURCE
+20V
DRVC X-WRITE SOURCE TIME L
DRVC X-READ SINK TIME L
DRVD XPLD 00
DRVD XHRD 00
DRVD XPLD 01
DRVD XHRD 01
DRVD XPLD 02
DRVD XHRD 02
DRVD XPLD 03
DRVD XHRD 03
DRVD XPLD 04
DRVD XHRD 04
DRVD XPLD 05
DRVD XHRD 05
DRVD XPLD 06
DRVD XHRD 06
DRVD XPLD 07
DRVD XHRD 07
DECODER 74154 E39
DRVD 00 SEL L
DRVD 01 SEL L
DRVD 02 SEL L
DRVD 03 SEL L
DRVD 04 SEL L
DRVD 05 SEL L
DRVD 06 SEL L
DRVD 07 SEL L
DRVC STACK SEL L
SEL
AND EN
DRVA X-WRITE CURRENT SOURCE
+20V
DRVD 00 SEL L
DRVC X-WRITE SOURCE TIME L
DRVC X-READ SINK TIME L
DRVD 01 SEL L
DRVD 10 SEL L
DRVD 11 SEL L
DRVD 12 SEL L
DRVD 13 SEL L
DRVD 14 SEL L
DRVD 15 SEL L
DRVD XPLD 08
DRVD XHRD 08
DRVD XPLD 09
DRVD XHRD 09
DRVD XPLD 10
DRVD XHRD 10
DRVD XPLD 11
DRVD XHRD 11
DRVD XPLD 12
DRVD XHRD 12
DRVD XPLD 13
DRVD XHRD 13
DRVD XPLD 14
DRVD XHRD 14
DRVD XPLD 15
DRVD XHRD 15
NOTES:
1) THE 8116,H224B AND 8236 BLOCK SCHEMATICS IN THIS PRINT SET REPRESENT 32K BY 32 BITS, WHICH IS 1 BANK OF STORAGE FOR 1 CONTROL. SEE D-MB20-0-MAP FOR THE LOCATION OF OTHER BANKS.
2) THESE PRINTS REFLECT REV B OF THE 8116 CIRCUIT SCHEMATICS
THESE PRINTS REFLECT REV A OF THE H224B CIRCUIT SCHEMATICS
THESE PRINTS REFLECT REV E OF THE 8236 CIRCUIT SCHEMATICS
3) VARIABLE DEFINITIONS
(N) INDICATE A CONTROL
(L) INDICATE A TIMING BOARD
(B) INDICATE A BANK NUMBER (0-15)
(M) INDICATE A SM NUMBER (0-3)
NOTE
1. UNLESS OTHERWISE INDICATED:
ALL DIODES ARE D672
ALL RESISTORS ARE 2kΩ, 1/2W
ALL TRANSISTORS ARE DEC 3/25
ALL CAPACITORS ARE 10nF
REVISIONS
CLK CHANGE NO. REV
DRVD,DRVF 4.331 12-AUG-76 13:25
FIRST USED ON UPTION MODEL: MB20
NEXT HIGHER ASSEMBLY: B-DD-MB20-0
TITLE: 32K X-Y DRIVE (8236)
SIZE D
CODE 85
NUMBER MB20-0-DRVD
REV





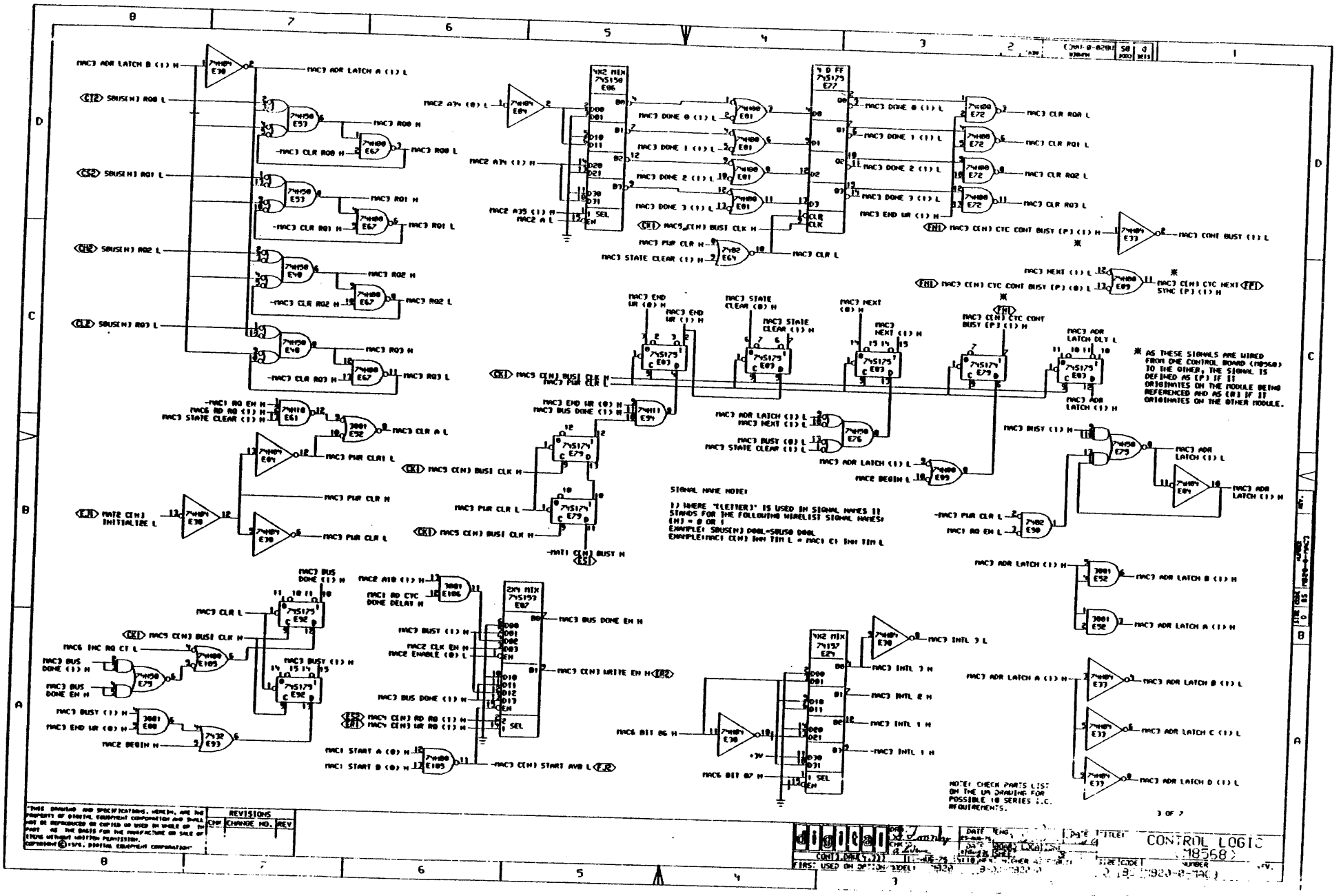
REVISIONS		
CHK	CHANGE NO.	REV

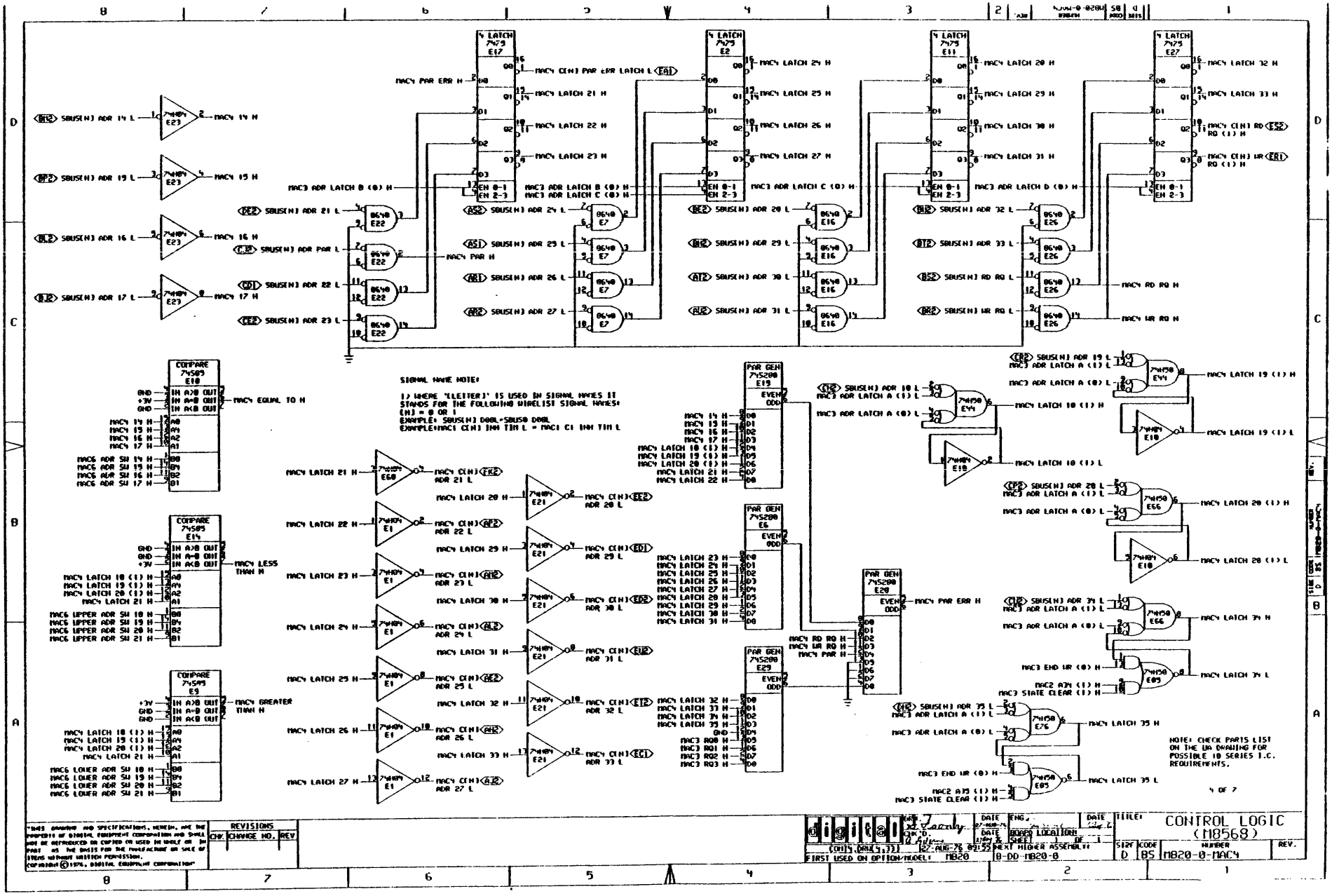
digital

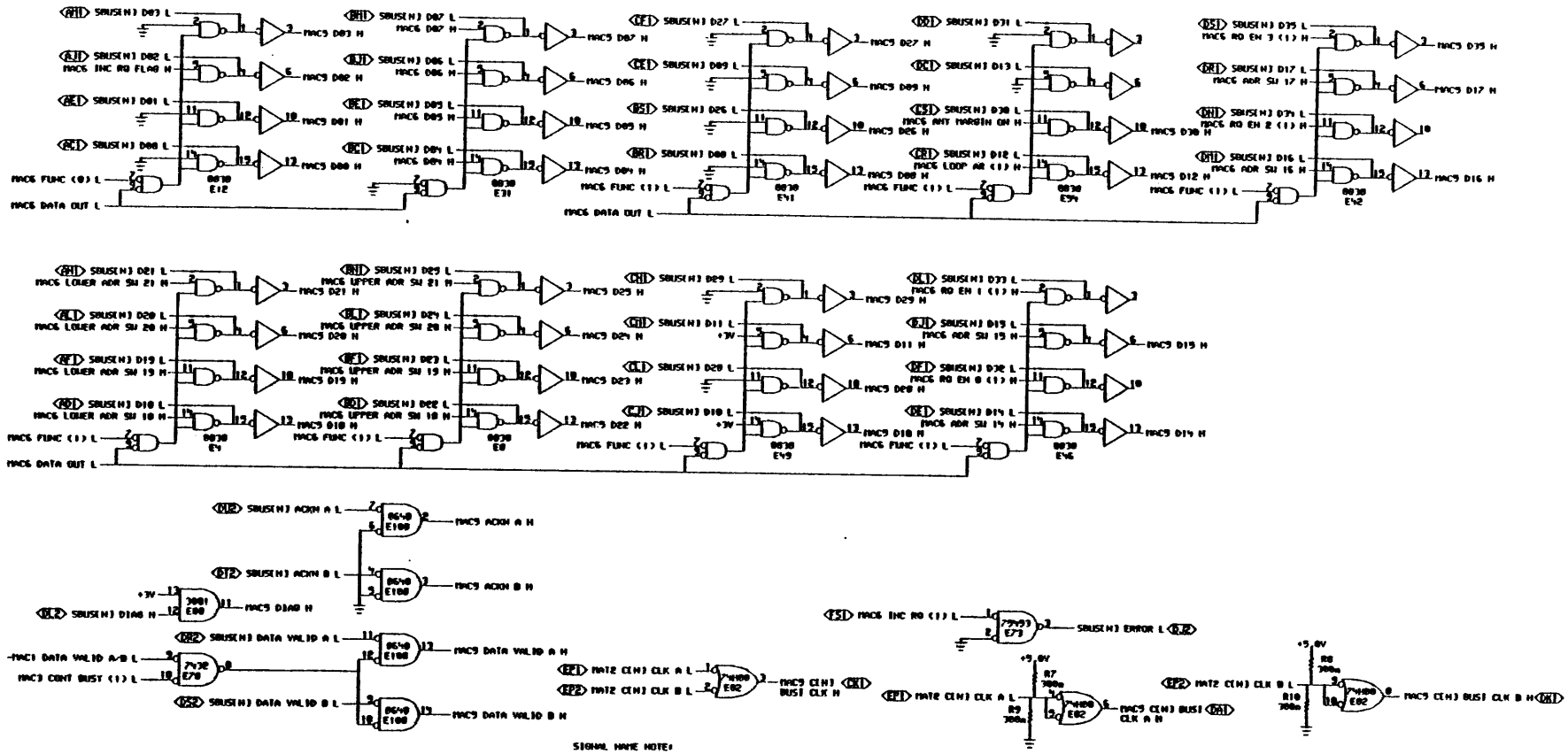
DRN. — DATE 25-AUG-76 — ENG. — DATE
CHK'D. — DATE — BOARD LOCATION: — SHEET 1 OF 1
CONT2.DRW(4,33) 11-AUG-76 14:10 — NEXT HIGHER ASSEMBLY: B-DD-MB20-0
FIRST USED ON OPTION/MODEL: MB20

TITLE: CONTROL LOGIC (M8568)

SIZE	CODE	NUMBER	REV.
D	BS	MB20-0-MAC2	


MAC3 ADR LATCH B (1) H
MAC3 ADR LATCH A (1) L
SBUS(N) R00 L
MAC3 R00 H
MAC3 CLR R00 H
MAC3 R00 L
SBUS(N) R01 L
MAC3 R01 H
MAC3 CLR R01 H
MAC3 R01 L
SBUS(N) R02 L
MAC3 R02 H
MAC3 CLR R02 H
MAC3 R02 L
SBUS(N) R03 L
MAC3 R03 H
MAC3 CLR R03 H
MAC3 R03 L
MAC2 A24 (0) L
MAC2 A24 (1) H
MAC2 A25 (1) H
MAC2 A L
MAC3 DONE 0 (1) L
MAC3 DONE 1 (1) L
MAC3 DONE 2 (1) L
MAC3 DONE 3 (1) L
MAC3 END WR (1) H
MAC3 CLR R00 L
MAC3 CLR R01 L
MAC3 CLR R02 L
MAC3 CLR R03 L
MAC3 C(N) BUSY CLR H
MAC3 PWR CLR H
MAC3 STATE CLEAR (1) H
MAC3 CLR L
MAC3 C(N) CYC CONT BUSY (P) (1) H
MAC3 CONT BUSY (1) L
MAC3 NEXT (1) L
MAC3 C(N) CYC CONT BUSY (P) (0) L
MAC3 C(N) CYC NEXT SYNC (P) (1) H
MAC3 END WR (0) H
MAC3 STATE CLEAR (0) H
MAC3 NEXT (0) H
MAC3 ADR LATCH DLY L
MAC3 ADR LATCH (1) H
MAC3 ADR LATCH (1) L
MAC3 BUSY (1) H
MAC3 BUS DONE (1) H
MAC3 BUS DONE EN H
MAC3 BUSY (0) L
MAC2 BEGIN L
MAC3 PWR CLR L
MAC3 RD EN H
MAC3 ADR LATCH B (1) H
MAC3 ADR LATCH A (1) H
MAC3 ADR LATCH A (1) L
MAC3 ADR LATCH B (1) L
MAC3 ADR LATCH C (1) L
MAC3 ADR LATCH D (1) L
MAC3 INTL 3 L
MAC3 INTL 3 H
MAC3 INTL 2 H
MAC3 INTL 1 H
MAC3 INTL 1 L
MAC6 BIT 06 H
MAC6 BIT 07 H
MAC3 C(N) WRITE EN H
MAC3 C(N) START AVB L
MAC1 START A (0) H
MAC1 START B (0) H
MAC1 RD CYC DONE DELAY H
MAC2 CLK EN H
MAC2 ENABLE (0) L
MAC6 INC RD CT L
MAC2 A10 (1) H
MAC3 C(N) INITIALIZE L
MAC3 C(N) BUSY H
SIGNAL NAME NOTE:
1) WHERE "(LETTER)" IS USED IN SIGNAL NAMES IT STANDS FOR THE FOLLOWING WIRELIST SIGNAL NAMES: (N) = 0 OR 1
EXAMPLE: SBUS(N) R00L=SBUS0 R00L
EXAMPLE: MAC1 C(N) INH TIM L = MAC1 C1 INH TIM L
* AS THESE SIGNALS ARE WIRED FROM ONE CONTROL BOARD (M9560) TO THE OTHER, THE SIGNAL IS DEFINED AS (P) IF IT ORIGINATES ON THE MODULE BEING REFERENCED AND AS (R) IF IT ORIGINATES ON THE OTHER MODULE.
NOTE: CHECK PARTS LIST ON THE UA DRAWING FOR POSSIBLE 10 SERIES I.C. REQUIREMENTS.
CONTROL LOGIC M8568
3 OF 7


SIGNAL NAME NOTE:
1) WHERE 'LETTER' IS USED IN SIGNAL NAMES IT STANDS FOR THE FOLLOWING WIRELIST SIGNAL NAMES: (N) = 0 OR 1
EXAMPLE: SBUS(N) D00L=SBUS0 D00L
EXAMPLE:MAC1 C(N) INH TIM L = MAC1 C1 INH TIM L
NOTE: CHECK PARTS LIST ON THE DA DRAWING FOR POSSIBLE 10 SERIES I.C. REQUIREMENTS.
4 OF 7
CONTROL LOGIC (M8568)
M820-0-MAC4
REVISIONS
FIRST USED ON OPTION/MODEL: M820



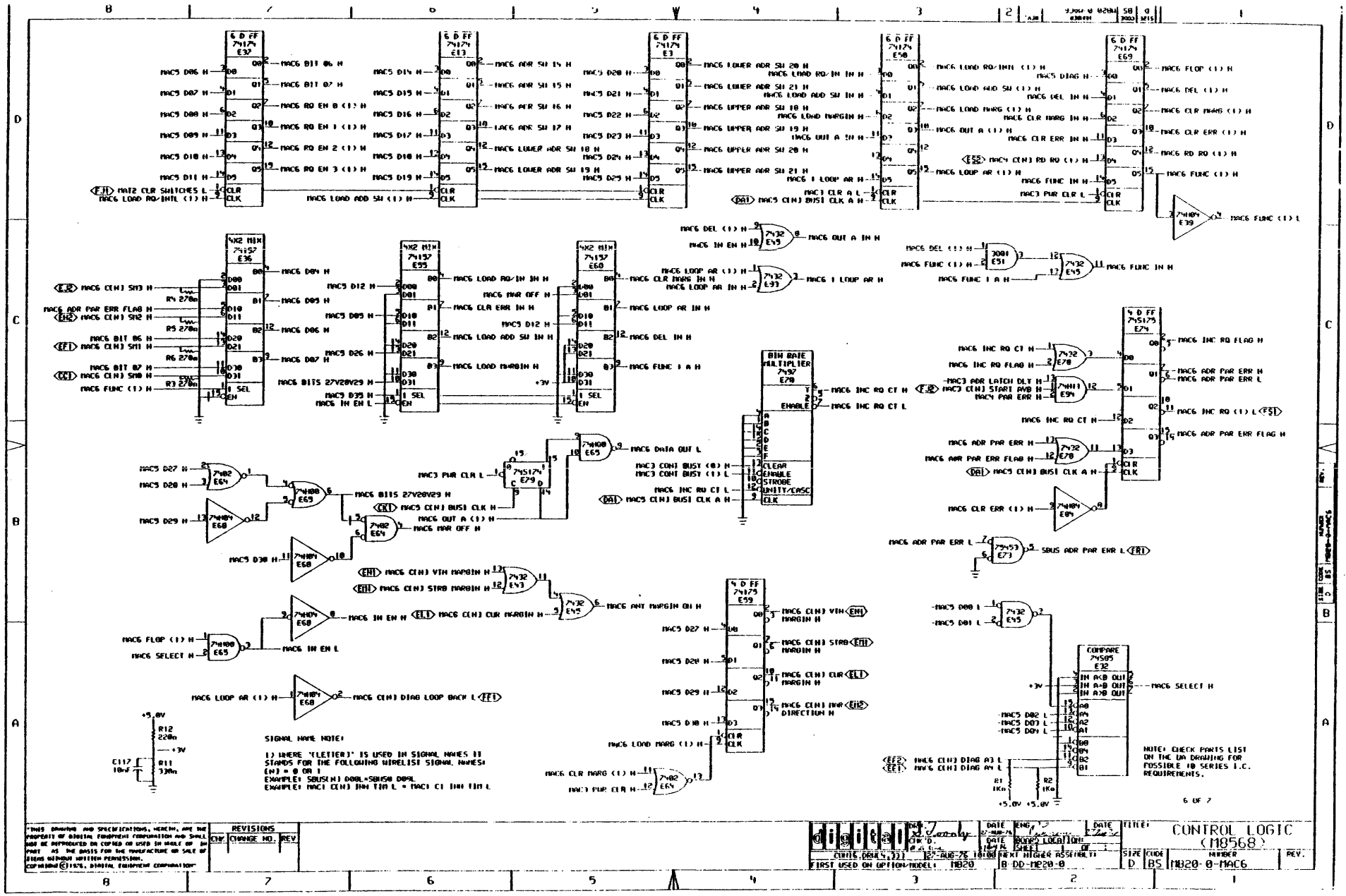
REVISIONS

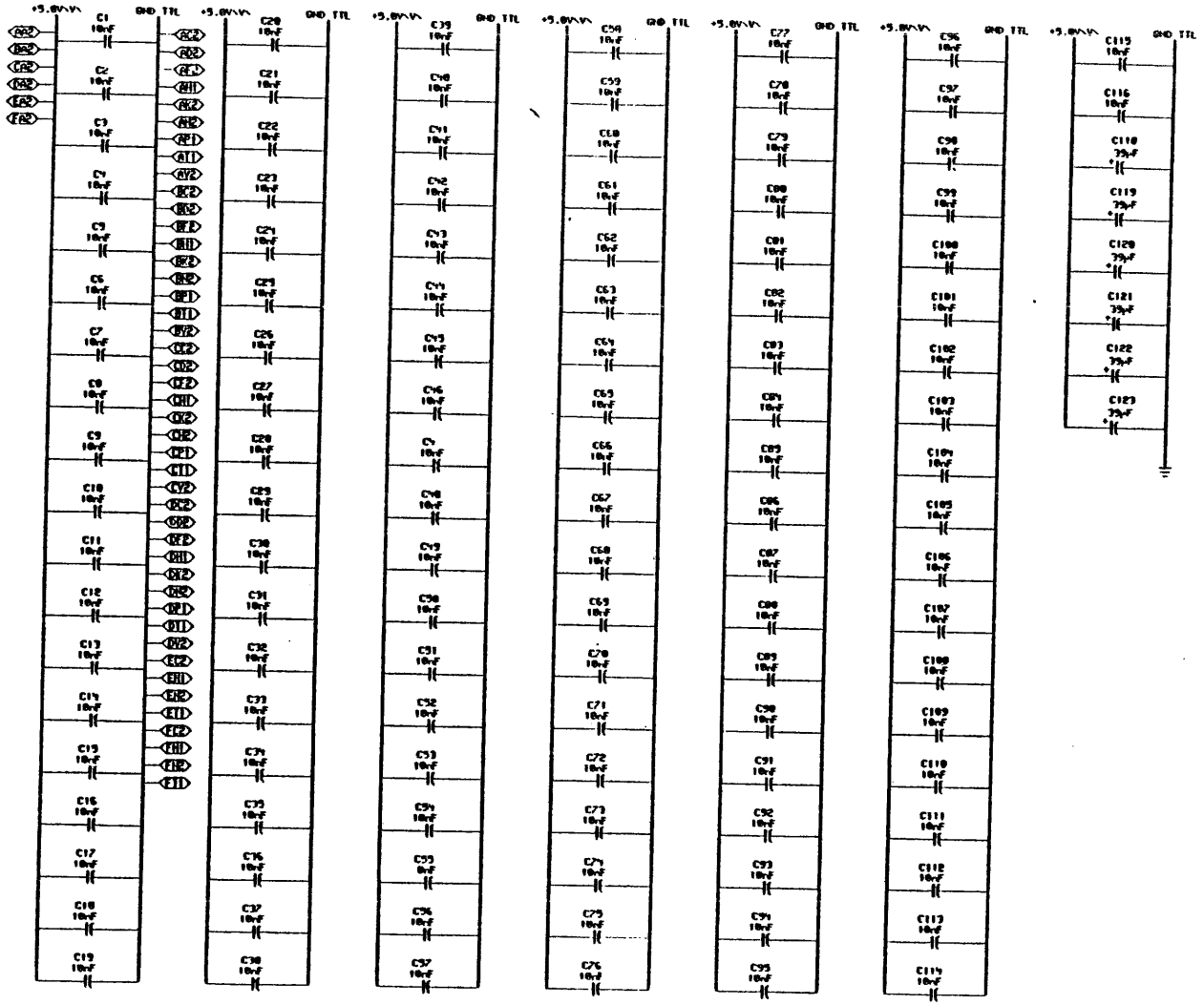
CHK	CHANGE NO.	REV

CONTS.DRA(5,72) III-AUG-76 14:16 NEXT HIGHER ASSEMBLY: B-DD-M820-0
FIRST USED ON OPTION-MODEL: M820

TITLE:	CONTROL LOGIC (M8568)	
SIZE	CODE	NUMBER
D	BS	M820-0-MAC5

REV.


SIGNAL NAME NOTE:
1) WHERE "(LETTER)" IS USED IN SIGNAL NAMES IT STANDS FOR THE FOLLOWING WIRELIST SIGNAL NAMES: (N) = 0 OR 1
EXAMPLE: SBUS(N) D00L=SBUS0 D00L
EXAMPLE: MAC1 CI(N) INH TIM L = MAC1 CI INH TIM L
NOTE: CHECK PARTS LIST ON THE DA DRAWING FOR POSSIBLE 10 SERIES I.C. REQUIREMENTS.
6 OF 7
CONTROL LOGIC (M8568)
M8568
M820-0-MAC6
REVISIONS
CHANGE NO.
digital
M820
SIZE D
CODE BS



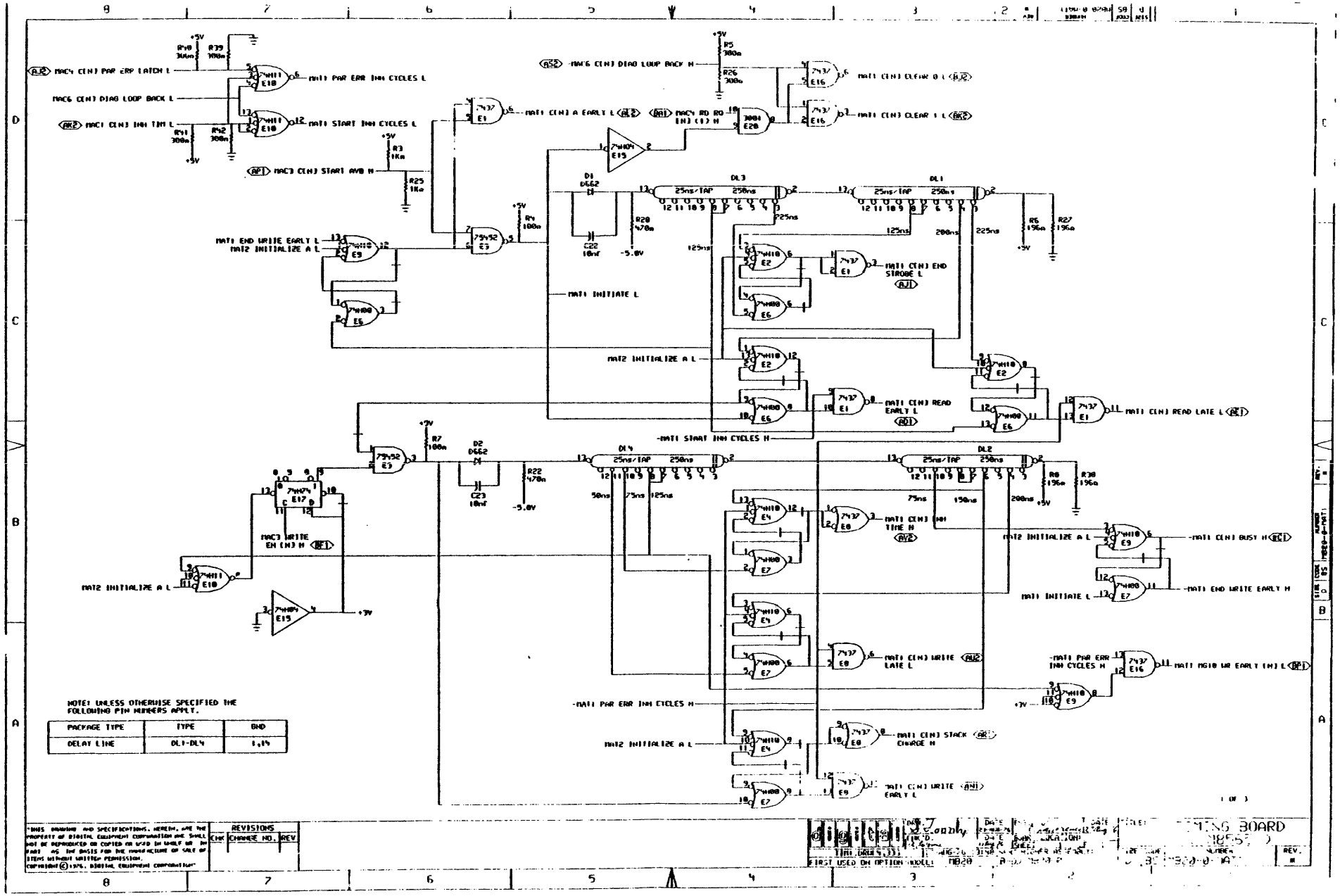
NOTE: CHECK PARTS LIST ON THE UA DRAWING FOR POSSIBLE 10 SERIES I.C. REQUIREMENTS.

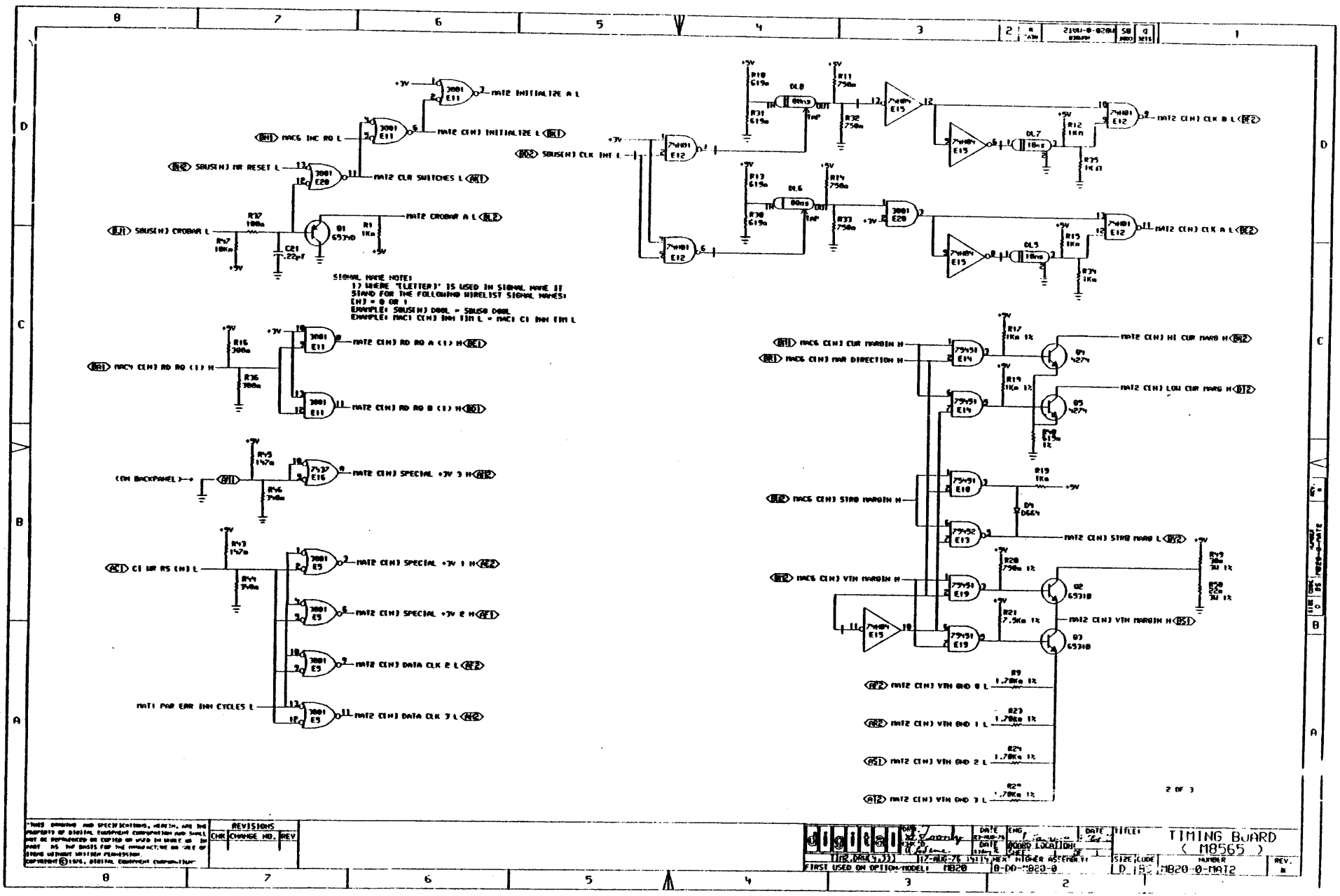
7 OF 7

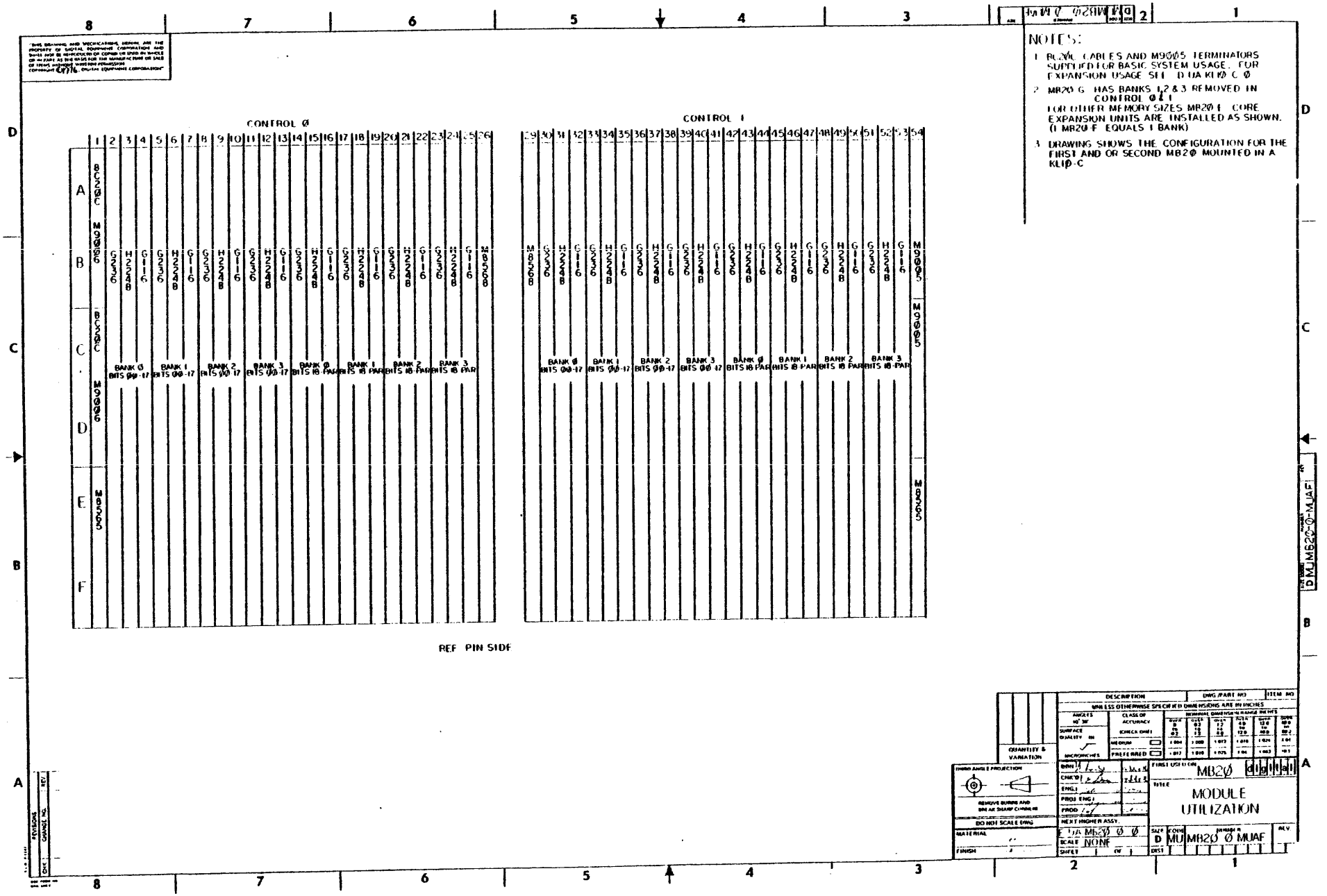
REVISIONS		
CHK	CHANGE NO.	REV

digital

CONTR DRWG NO. 7231 | 11-AUG-75 | APPROVED BY: | DATE | FSD | DATE | TITLE: CONTROL LOGIC (M8568)
FIRST USED ON OPTION-MODEL: M 20 | BOARD LOCATION: SHEET 1 OF | NEXT HIGHER ASSEMBLY: B-DD-M820-0 | SIZE D | CODE BS | NUMBER M820-0-MAC7 | REV.


MAC4 CEN] PAR ERR LATCH L
MAC6 CEN] DIAG LOOP BACK L
MAC1 CEN] INH TIM L
MAT1 PAR ERR INH CYCLES L
MAT1 START INH CYCLES L
MAC3 CEN] START AMB H
MAT1 CEN] A EARLY L
MAC6 CEN] DIAG LOOP BACK H
MAC4 RD RD INH (1) H
MAT1 CEN] CLEAR 0 L
MAT1 CEN] CLEAR 1 L
MAT1 END WRITE EARLY L
MAT2 INITIALIZE A L
MAT1 INITIATE L
MAT1 CEN] END STROBE L
MAT1 CEN] READ EARLY L
MAT1 CEN] READ LATE L
-MAT1 START INH CYCLES H
MAC3 WRITE EN (N) H
MAT1 CEN] TIME H
MAT1 CEN] BUST H
-MAT1 END WRITE EARLY H
MAT1 CEN] WRITE LATE L
-MAT1 PAR ERR INH CYCLES H
MAT1 CEN] STACK CHARGE H
MAT1 CEN] WRITE EARLY L
DL1
DL2
DL3
DL4
25ns/TAP
250ns
NOTE: UNLESS OTHERWISE SPECIFIED THE FOLLOWING PIN NUMBERS APPLY.
PACKAGE TYPE
TYPE
PIN
DELAY LINE
DL1-DL4
14
REVISIONS
CHK
CHANGE NO.
REV
FIRST USED ON OPTION MODEL

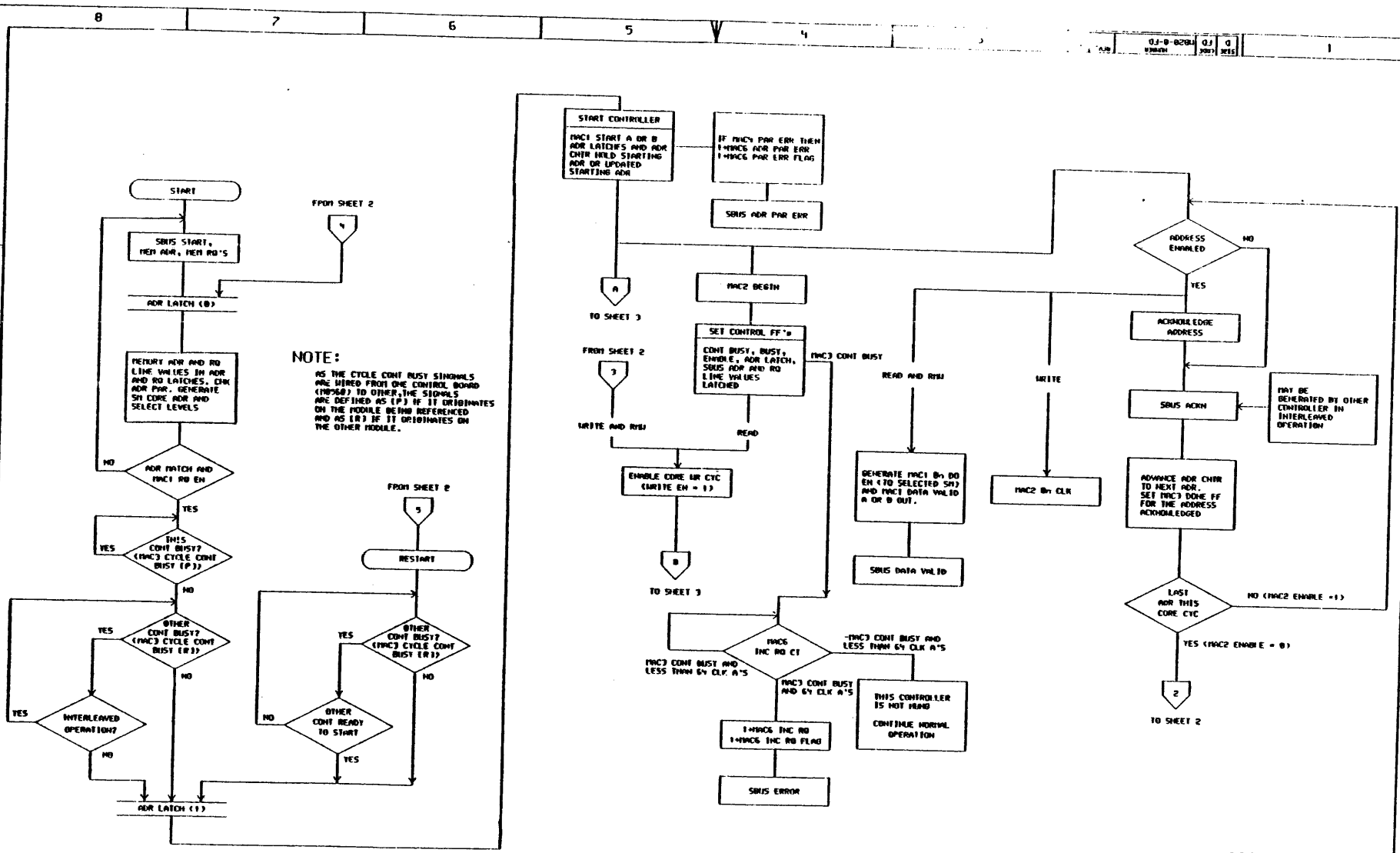

MAT2 INITIALIZE A L
MAT2 CINJ INITIALIZE L (BK1)
MAC6 INC RO L (BH1)
SBUSINJ MR RESET L (BH2)
MAT2 CLR SWITCHES L (AK1)
SBUSINJ CROBAR L (EJ1)
MAT2 CROBAR A L (BL2)
SBUSINJ CLK INT L (BH2)
MAT2 CINJ CLK B L (BF2)
MAT2 CINJ CLK A L (BC2)
SIGNAL NAME NOTE:
1) WHERE "LETTER" IS USED IN SIGNAL NAME IT
STAND FOR THE FOLLOWING WIRELIST SIGNAL NAMES:
(N) = 0 OR 1
EXAMPLE: SBUSINJ DOOL = SBUS0 DOOL
EXAMPLE: MAC1 CINJ INH (N) L = MAC1 CI INH EN L
MAC4 CINJ RO RO (1) H (BB1)
MAT2 CINJ RO RO A (1) H (BE1)
MAT2 CINJ RO RO B (1) H (BG1)
(ON BACKPANEL)
MAT2 CINJ SPECIAL +3V 3 H (AH2)
CI WR RS (N) L (AC1)
MAT2 CINJ SPECIAL +3V 1 H (AE2)
MAT2 CINJ SPECIAL +3V 2 H (AF1)
MAT2 CINJ DATA CLK 2 L (AF2)
MAT1 PAR ERR INH CYCLES L
MAT2 CINJ DATA CLK 3 L (AH2)
MAC6 CINJ CUR MARGIN H (BT1)
MAC6 CINJ MAR DIRECTION H (BR1)
MAT2 CINJ HI CUR MARG H (BH2)
MAT2 CINJ LOW CUR MARG H (BT2)
MAC6 CINJ STRB MARGIN H (BL2)
MAT2 CINJ STRB MARG L (BY2)
MAC6 CINJ VTH MARGIN H (BF2)
MAT2 CINJ VTH MARGIN H (BS1)
MAT2 CINJ VTH GND 0 L (AP2)
MAT2 CINJ VTH GND 1 L (AR2)
MAT2 CINJ VTH GND 2 L (AS1)
MAT2 CINJ VTH GND 3 L (AT2)
2 OF 3
REVISIONS
CHK CHANGE NO. REV
digital
TIMING BOARD
(M8565)
M820-0-MAT2
M8565

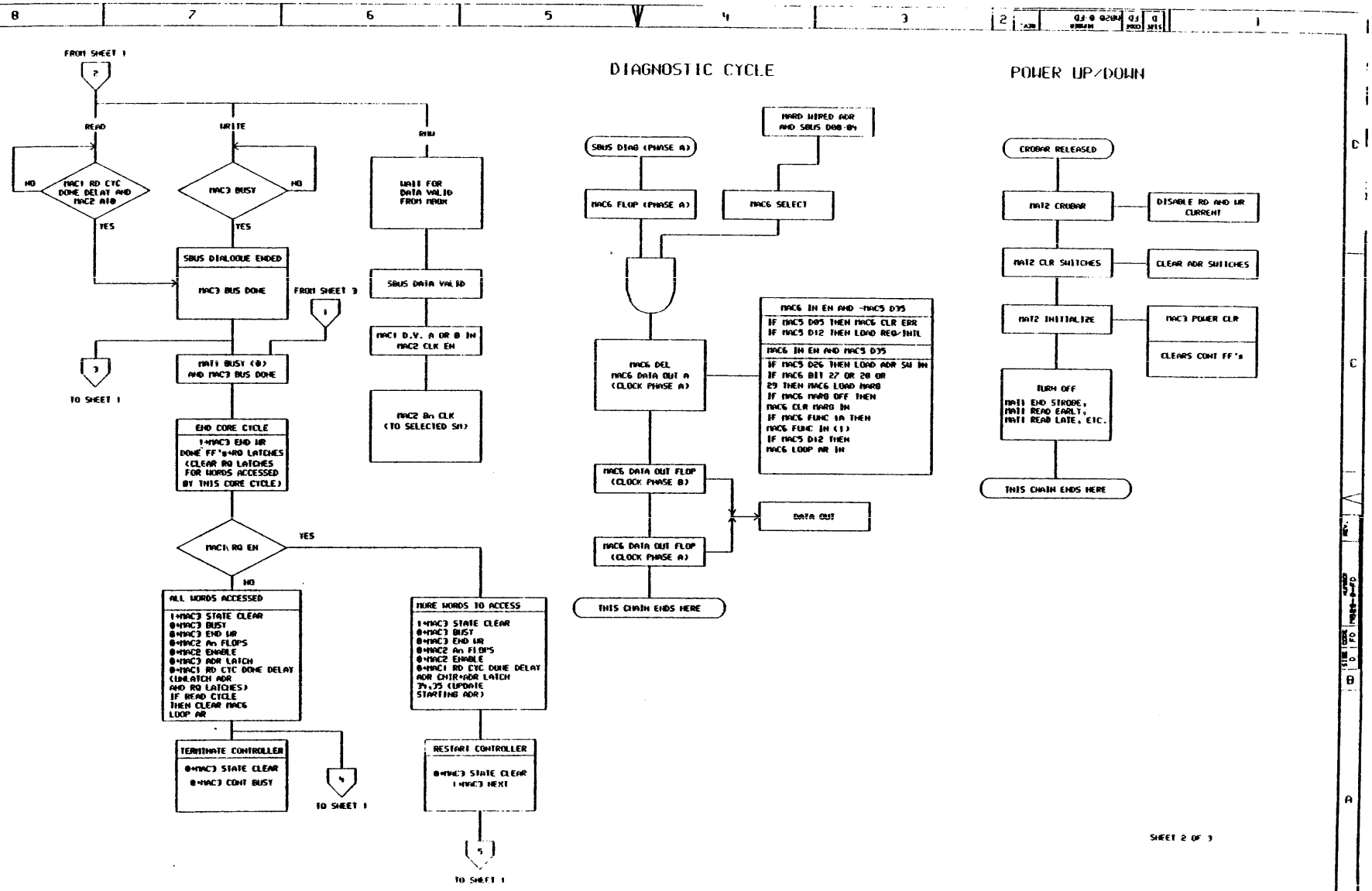


NOTES:

1. BC70L CABLES AND M9005 TERMINATORS SUPPLIED FOR BASIC SYSTEM USAGE. FOR EXPANSION USAGE SEE D UA KL10 C Ø
2. MB20 G HAS BANKS 1,2 & 3 REMOVED IN CONTROL Ø & 1
   FOR OTHER MEMORY SIZES MB20 E CORE EXPANSION UNITS ARE INSTALLED AS SHOWN. (1 MB20 F EQUALS 1 BANK)
3. DRAWING SHOWS THE CONFIGURATION FOR THE FIRST AND OR SECOND MB20 MOUNTED IN A KL10-C

FIRST USED ON	MB20
TITLE	MODULE UTILIZATION
SIZE	D
CODE	MU
NUMBER	MB20 Ø MUAF
REV	
DIST	
SCALE	NONE
SHEET	1 OF 1
NEXT HIGHER ASSY	D UA MB20 Ø Ø
DRN	
CHK'D	
ENG'G	
PROJ ENG'G	
PROD	
MATERIAL	
FINISH	
DO NOT SCALE DWG	
REMOVE BURRS AND BREAK SHARP CORNERS	
THIRD ANGLE PROJECTION	
QUANTITY & VARIATION	
DESCRIPTION	
DWG / PART NO	
ITEM NO	
UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES	
ANGLES ±0° 30'	
SURFACE QUALITY ✓	
MICROINCHES	
CLASS OF ACCURACY (CHECK ONE)	
MEDIUM	
PREFERRED	
NOMINAL DIMENSION RANGE INCHES	
REVISIONS	
DWG	
CHG	
ORIGINATOR	
DATE ISSUED	D MUMB20-Ø-MUAF
SIZE	B


START
SBUS START, MEM ADR, MEM RQ'S
FROM SHEET 2
4
ADR LATCH (0)
MEMORY ADR AND RQ LINE VALUES IN ADR AND RQ LATCHES, CHK ADR PAR, GENERATE SM CORE ADR AND SELECT LEVELS
NO
ADR MATCH AND MAC1 RQ EN
YES
YES
THIS CONT BUSY? (MAC) CYCLE CONT BUSY (P))
NO
YES
OTHER CONT BUSY? (MAC) CYCLE CONT BUSY (B))
NO
YES
INTERLEAVED OPERATION?
NO
FROM SHEET 2
5
RESTART
YES
OTHER CONT BUSY? (MAC) CYCLE CONT BUSY (A))
NO
NO
OTHER CONT READY TO START
YES
ADR LATCH (1)
NOTE:
AS THE CYCLE CONT BUSY SIGNALS ARE WIRED FROM ONE CONTROL BOARD (MB960) TO OTHER, THE SIGNALS ARE DEFINED AS (P) IF IT ORIGINATES ON THE MODULE BEING REFERENCED AND AS (B) IF IT ORIGINATES ON THE OTHER MODULE.
START CONTROLLER
MAC1 START A OR B ADR LATCHES AND ADR CNTR HOLD STARTING ADR OR UPDATED STARTING ADR
IF MAC5 PAR ERR THEN 1=MAC5 ADR PAR ERR 1=MAC5 PAR ERR FLAG
SBUS ADR PAR ERR
A
TO SHEET 3
MAC2 BEGIN
SET CONTROL FF's
CONT BUSY, BUSY, ENABLE, ADR LATCH, SBUS ADR AND RQ LINE VALUES LATCHED
MAC3 CONT BUSY
FROM SHEET 2
3
WRITE AND RMW
READ
ENABLE CORE WR CYC (WRITE EN = 1)
B
TO SHEET 3
READ AND RMW
WRITE
GENERATE MAC1 Bn DO EN (TO SELECTED SM) AND MAC1 DATA VALID A OR B OUT.
MAC2 Bn CLR
SBUS DATA VALID
MAC5 INC RQ CT
-MAC3 CONT BUSY AND LESS THAN 64 CLK A'S
MAC3 CONT BUSY AND LESS THAN 64 CLK A'S
MAC3 CONT BUSY AND 64 CLK A'S
THIS CONTROLLER IS NOT HUNG
CONTINUE NORMAL OPERATION
1=MAC5 INC RQ 1=MAC5 INC RQ FLAG
SBUS ERROR
ADDRESS ENABLED
NO
YES
ACKNOWLEDGE ADDRESS
SBUS ACKN
MAY BE GENERATED BY OTHER CONTROLLER IN INTERLEAVED OPERATION
ADVANCE ADR CNTR TO NEXT ADR. SET MAC3 DONE FF FOR THE ADDRESS ACKNOWLEDGED
LAST ADR THIS CORE CYC
NO (MAC2 ENABLE =1)
YES (MAC2 ENABLE = 0)
2
TO SHEET 2
SHEET 1 OF 3
REVISIONS
ENG CHANGE NO. REV
digital
TITLE: MB20 FLOW DIAGRAM
SIZE CODE D
FD MB20-0-ED
REV.



REVISIONS

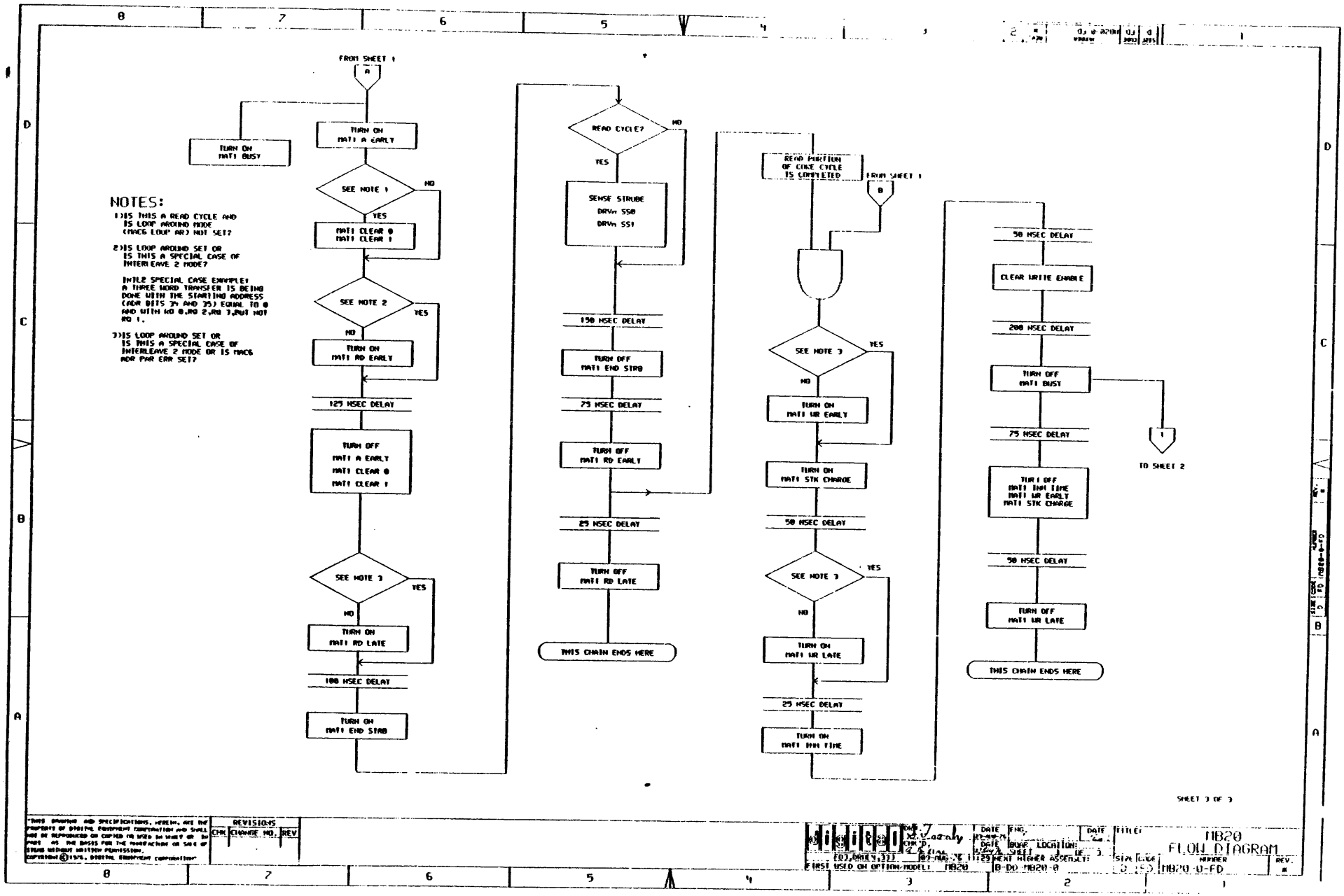
CHK	CHANGE NO.	REV

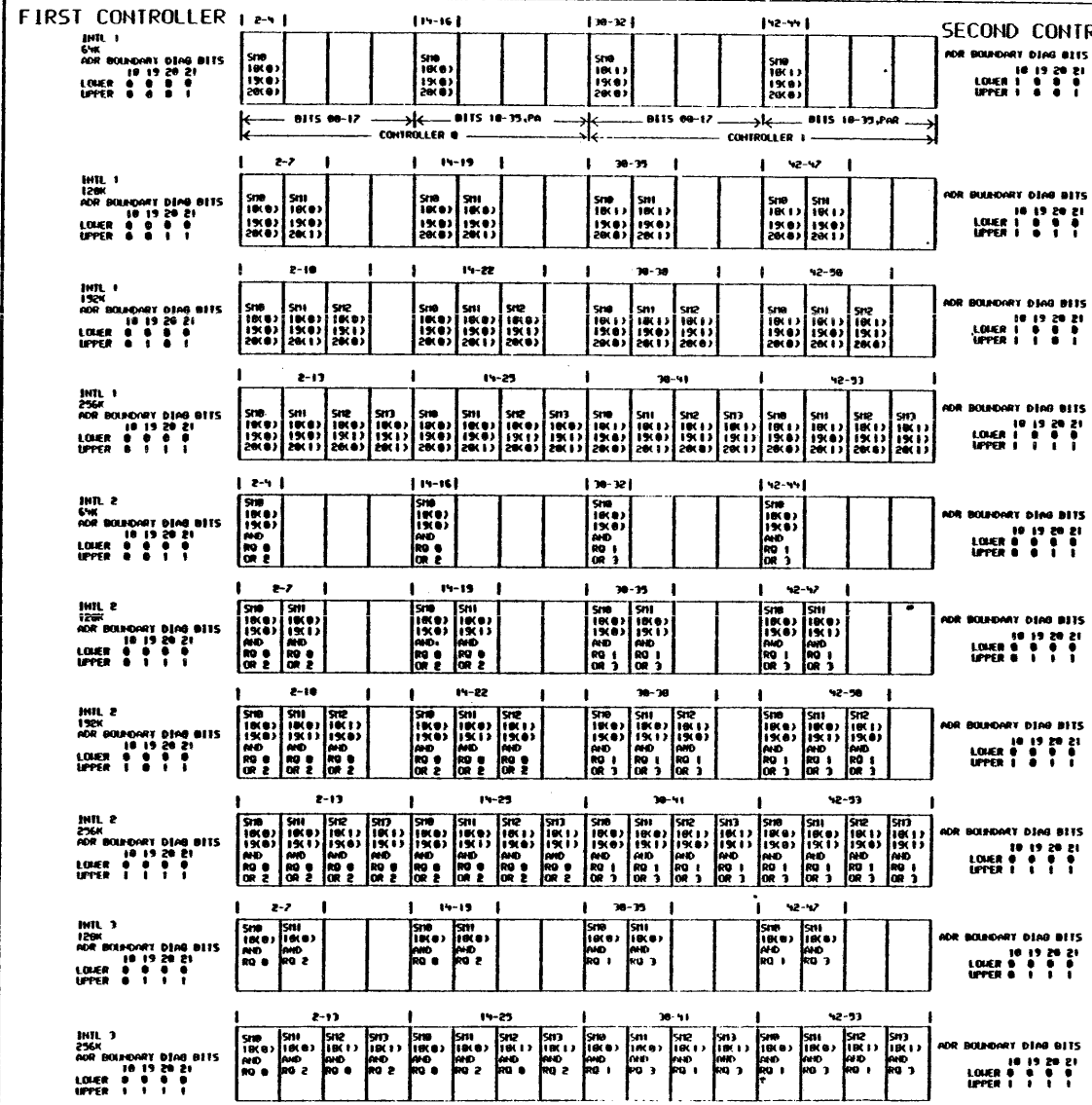
digital

DRAWN	DATE	ENG
CHECKED	DATE	MGR
FDZ_DRLEV_333	17-AUG-76 13:01	BOARD LOCATION:
FIRST USED ON OPTION-MODEL: M 20		NEXT HIGHER ASSEMBLY: 8-DD-MB20-0

TITLE: MB20 FLOW DIAGRAM

SIZE	CODE	NUMBER	REV.
D	FD	MB20-0-FD	


FROM SHEET 1
A
TURN ON MAT1 BUSY
TURN ON MAT1 A EARLY
SEE NOTE 1
NO
YES
MAT1 CLEAR 0
MAT1 CLEAR 1
SEE NOTE 2
YES
NO
TURN ON MAT1 RD EARLY
125 NSEC DELAY
TURN OFF MAT1 A EARLY MAT1 CLEAR 0 MAT1 CLEAR 1
SEE NOTE 3
YES
NO
TURN ON MAT1 RD LATE
100 NSEC DELAY
TURN ON MAT1 END STRB
READ CYCLE?
NO
YES
SENSE STRUBE DRVn S50 DRVn S51
150 NSEC DELAY
TURN OFF MAT1 END STRB
75 NSEC DELAY
TURN OFF MAT1 RD EARLY
25 NSEC DELAY
TURN OFF MAT1 RD LATE
THIS CHAIN ENDS HERE
READ PORTION OF CASE CYCLE IS COMPLETED
FROM SHEET 1
B
SEE NOTE 3
YES
NO
TURN ON MAT1 WR EARLY
TURN ON MAT1 STK CHARGE
50 NSEC DELAY
SEE NOTE 3
YES
NO
TURN ON MAT1 WR LATE
25 NSEC DELAY
TURN ON MAT1 INH TIME
50 NSEC DELAY
CLEAR WRITE ENABLE
200 NSEC DELAY
TURN OFF MAT1 BUSY
75 NSEC DELAY
TURN OFF MAT1 INH TIME MAT1 WR EARLY MAT1 STK CHARGE
50 NSEC DELAY
TURN OFF MAT1 WR LATE
THIS CHAIN ENDS HERE
1
TO SHEET 2
NOTES:
1)IS THIS A READ CYCLE AND IS LOOP AROUND MODE (MAC6 LOUP AR) NOT SET?
2)IS LOOP AROUND SET OR IS THIS A SPECIAL CASE OF INTERLEAVE 2 MODE?
INTL2 SPECIAL CASE EXAMPLE: A THREE WORD TRANSFER IS BEING DONE WITH THE STARTING ADDRESS (ADR BITS 24 AND 25) EQUAL TO 0 AND WITH WD 0,WD 2,WD 3,BUT NOT WD 1.
3)IS LOOP AROUND SET OR IS THIS A SPECIAL CASE OF INTERLEAVE 2 MODE OR IS MAC6 ADR PAR ERR SET?
SHEET 3 OF 3
REVISIONS
CHK CHANGE NO. REV
MB20
FLOW DIAGRAM
MB20-0-FD



THE CHART AT LEFT SHOWS 'NORMAL' SM INSTALLATION AND ALSO THE SIGNALS NECESSARY TO SELECT A SM FOR ALL SIZE AND INTERLEAVE COMBINATIONS.

HARDWIRED CONTROLLER SELECTION		
	PIN EF2	PIN EE1
C0	GND	GND
C1	GND	—
C2	—	GND
C3	—	—

MB20 DESKEW PROCEDURE

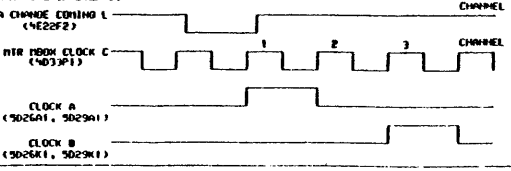
NOTES:
1- USE EQUAL LENGTH PROBES WITH SHORT GROUND CLIPS.
2- SELECT CLOCK RATE 0 ON KL10 (FOR ON LINE DESKEW PROCEDURE).
3- THE M8565 DELAY LINES ARE ARRANGED SO THAT (A) CLOCK IS THE TOP POT AND (B) CLOCK IS THE BOTTOM POT.
4- A 7000 SERIES OR EQUIVALENT SCOPE SHOULD BE USED WHEN DESKEWING THE MB20.
5- THE SCOPE SWEEP RATE SHOULD BE 5ns OR LESS PER Cm.

MB20 DESKEW PROCEDURE (USING A MA20-TA)

1- PUT ALL SWITCHES DOWN.
2- PUT CHANNEL A ON THE MB20-TA AT A23N1 (CONI CLK A H), PUT CHANNEL B ON THE MEMORY AT D26A1.
3- ADJUST THE UPPER DELAY ON THE M8565 IN EF01 UNTIL THE RISING EDGES MATCH OR CROSS AT +1.5V.
4- REPEAT STEPS 1 THROUGH 3 FOR D29A1 AND THE M8565 IN EF54.
5- PUT CHANNEL A ON THE MA20-TA AT A23K1 (CONI CLK B H), PUT CHANNEL B ON THE MEMORY AT D26K1.
6- ADJUST THE LOWER DELAY ON THE M8565 IN EF01 UNTIL THE RISING EDGES MATCH OR CROSS AT 1.5V.
7- REPEAT STEPS 5 THROUGH 6 FOR D29K1 AND THE M8565 IN EF54.

MB20 DESKEW PROCEDURE (ON LINE)

1- SET CLOCK RATE TO 'CR6', CLOCK SOURCE TO 'CS0'. TYPE 'FAI' TO TURN THE CLOCK ON.
2- PLACE CHANNEL 1 PROBE ON PIN 4E22F2 'A CHANGE COMING L'. SYNC ON CHANNEL 1.
3- PLACE CHANNEL 2 PROBE ON PIN 4D33P1 'MTR MBOX CLOCK C'.
4- ALIGN THE RISING EDGE OF THE PULSE DESIGNATED AS 1, SO THAT THE 50% POINT CROSSES THE SECOND VERTICAL DIVISION OF THE SCOPE SCREEN. NOTE THIS POSITION.
5- PLACE CHANNEL 2 PROBE ON PIN 5D26A1. ADJUST THE TOP DELAY ON THE M8565, SLOT 9EF01 SO THAT THE LEADING EDGE 50% POINT CROSSES THE SECOND VERTICAL DIVISION OF THE SCOPE SCREEN.
6- PLACE CHANNEL 2 PROBE ON PIN 5D29A1. ADJUST THE TOP DELAY ON THE M8565, SLOT 9EF54 SO THAT THE LEADING EDGE 50% POINT CROSSES THE SECOND VERTICAL DIVISION OF THE SCOPE SCREEN.
7- REPLACE CHANNEL 2 PROBE ON PIN 4D33P1 'MTR MBOX CLOCK C'.
8- ALIGN THE LEADING EDGE OF THE PULSE DESIGNATED AS 3, SO THAT THE 50% POINT CROSSES THE NEAREST VERTICAL DIVISION MARK OF THE SCOPE SCREEN. NOTE THIS POSITION.
9- PLACE CHANNEL 2 PROBE ON PIN 5D26K1. ADJUST THE BOTTOM DELAY ON THE M8565, SLOT 9EF01 SO THAT THE LEADING EDGE 50% POINT CROSSES THE VERTICAL DIVISION MARK NOTE IN STEP 8.
10- PLACE CHANNEL 2 PROBE ON PIN 5D29K1. ADJUST THE BOTTOM DELAY ON THE M8565, SLOT 9EF54 SO THAT THE LEADING EDGE 50% POINT CROSSES THE VERTICAL DIVISION MARK NOTE IN STEP 8.

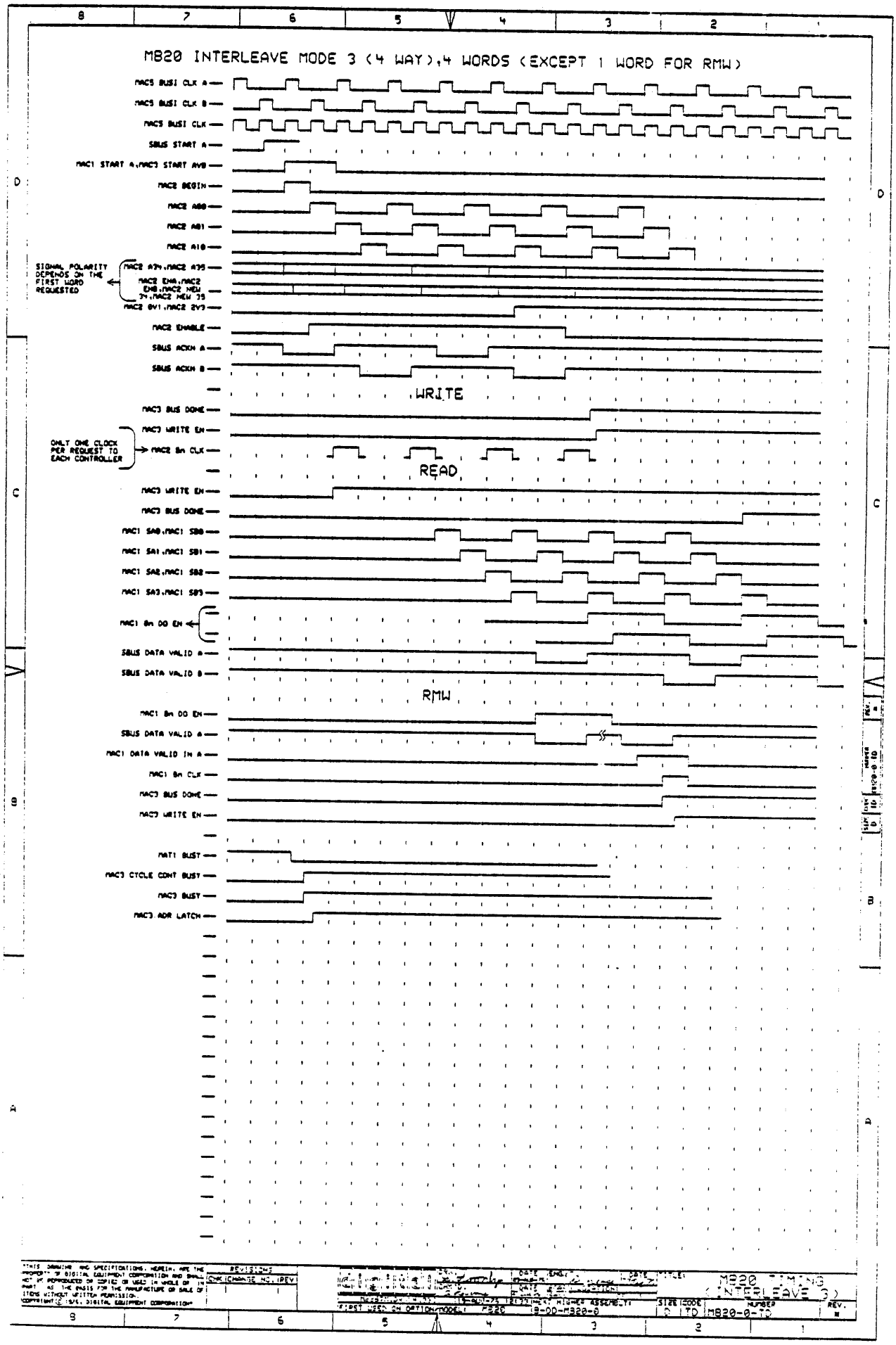


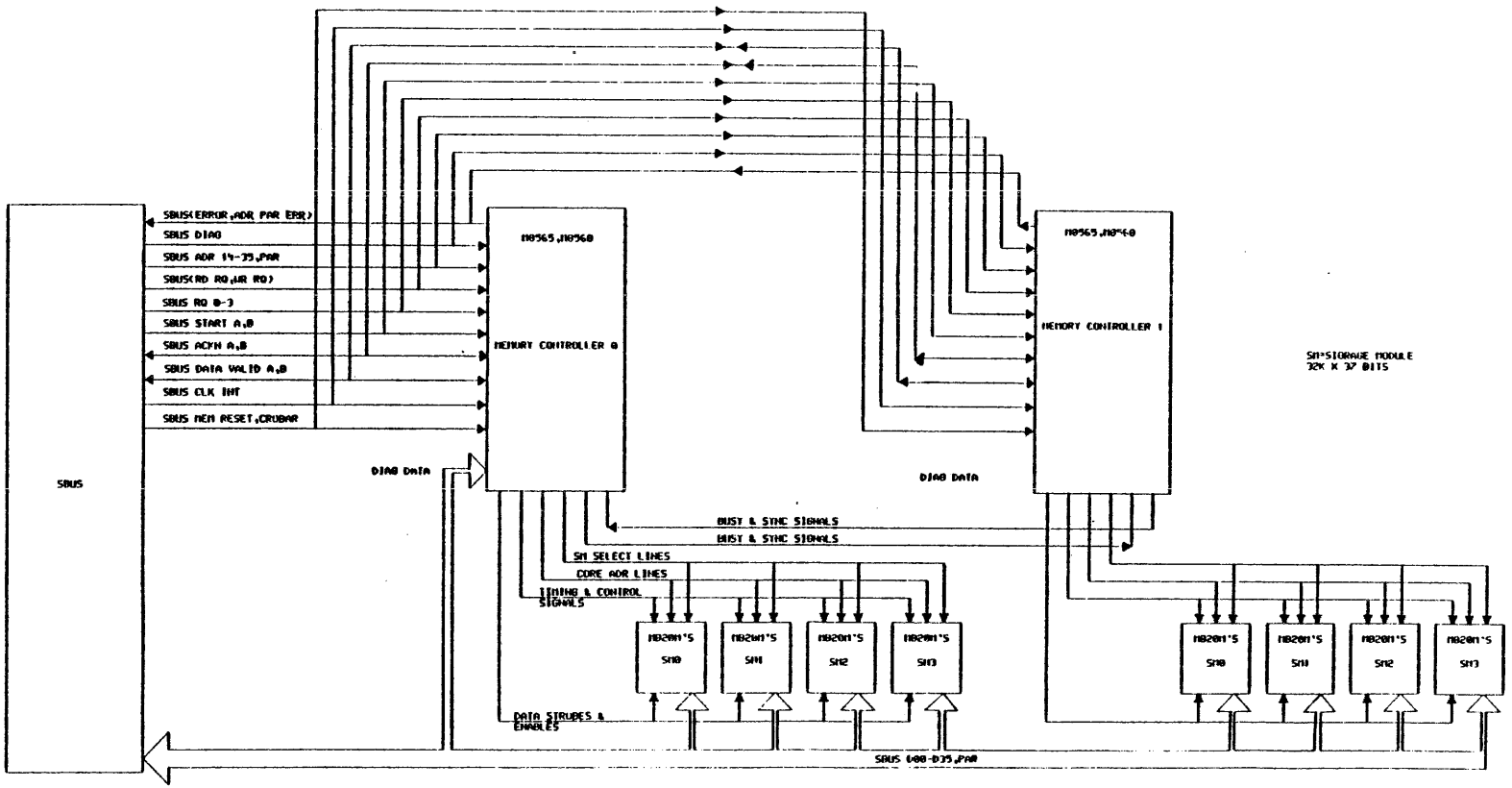
REVISIONS		
CHK	CHANGE NO.	REV

digital

DRN J. Gazzuolo	DATE 22-MAR-76	ENG.
CHK [illegible]	DATE 23-MAR-76	DATE
APVD [illegible]	DATE [illegible]	SIZE D
FIRST USED ON OPTION/MODEL: MB20	NEXT HIGHER ASSEMBLY: 8-DD-MB20-0	CODE BS
		NUMBER MB20-0-INS
		REV. M

TITLE: MB20 INSTRUCTION /SETUP CHARTS


MB20 INTERLEAVE MODE 3 (4 WAY),4 WORDS (EXCEPT 1 WORD FOR RMW)
MAC5 BUSI CLK A
MAC5 BUSI CLK B
MAC5 BUSI CLK
SBUS START A
MAC1 START A,MAC3 START AVB
MAC2 BEGIN
MAC2 A00
MAC2 A01
MAC2 A10
SIGNAL POLARITY DEPENDS ON THE FIRST WORD REQUESTED
MAC2 A34,MAC2 A35
MAC2 ENA,MAC2 ENB,MAC2 NEW 34,MAC2 NEW 35
MAC2 0V1,MAC2 2V3
MAC2 ENABLE
SBUS ACKN A
SBUS ACKN B
WRITE
MAC3 BUS DONE
MAC3 WRITE EN
ONLY ONE CLOCK PER REQUEST TO EACH CONTROLLER
MAC2 Bn CLK
READ
MAC3 WRITE EN
MAC3 BUS DONE
MAC1 SA0,MAC1 SB0
MAC1 SA1,MAC1 SB1
MAC1 SA2,MAC1 SB2
MAC1 SA3,MAC1 SB3
MAC1 Bn DO EN
SBUS DATA VALID A
SBUS DATA VALID B
RMW
MAC1 Bn DO EN
SBUS DATA VALID A
MAC1 DATA VALID IN A
MAC1 Bn CLK
MAC3 BUS DONE
MAC3 WRITE EN
MAT1 BUSY
MAC3 CYCLE CONT BUSY
MAC3 BUSY
MAC3 ADR LATCH
MB20 TIMING (INTERLEAVE 3)
MB20-0-TD



(BUS DETAIL)

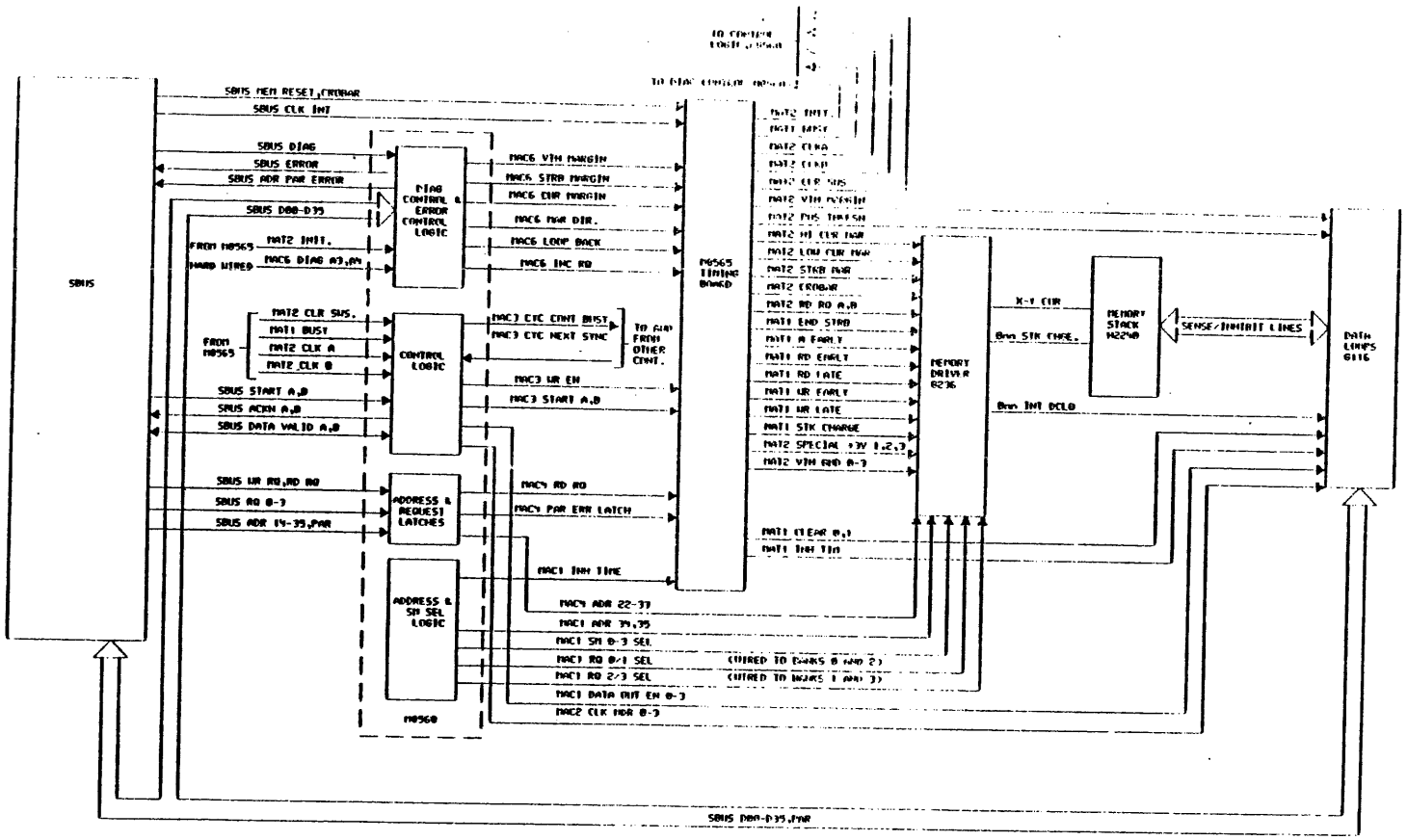
REVISIONS		
ECO	CHANGE NO.	REV

DRN D		DATE 27-MAR-76
CHK D		DATE 27-MAR-76
ENG.		DATE
MBD DATE 27-MAR-76 10:01		NEXT HIGHER ASSEMBLY
FIRST USED ON OPTION/MODEL: M820		B-DD-M820-0

SIZE D | CODE BD | NUMBER M820-0-1BD | REV.

TITLE: INTERCONNECTION BLOCK DIAGRAM

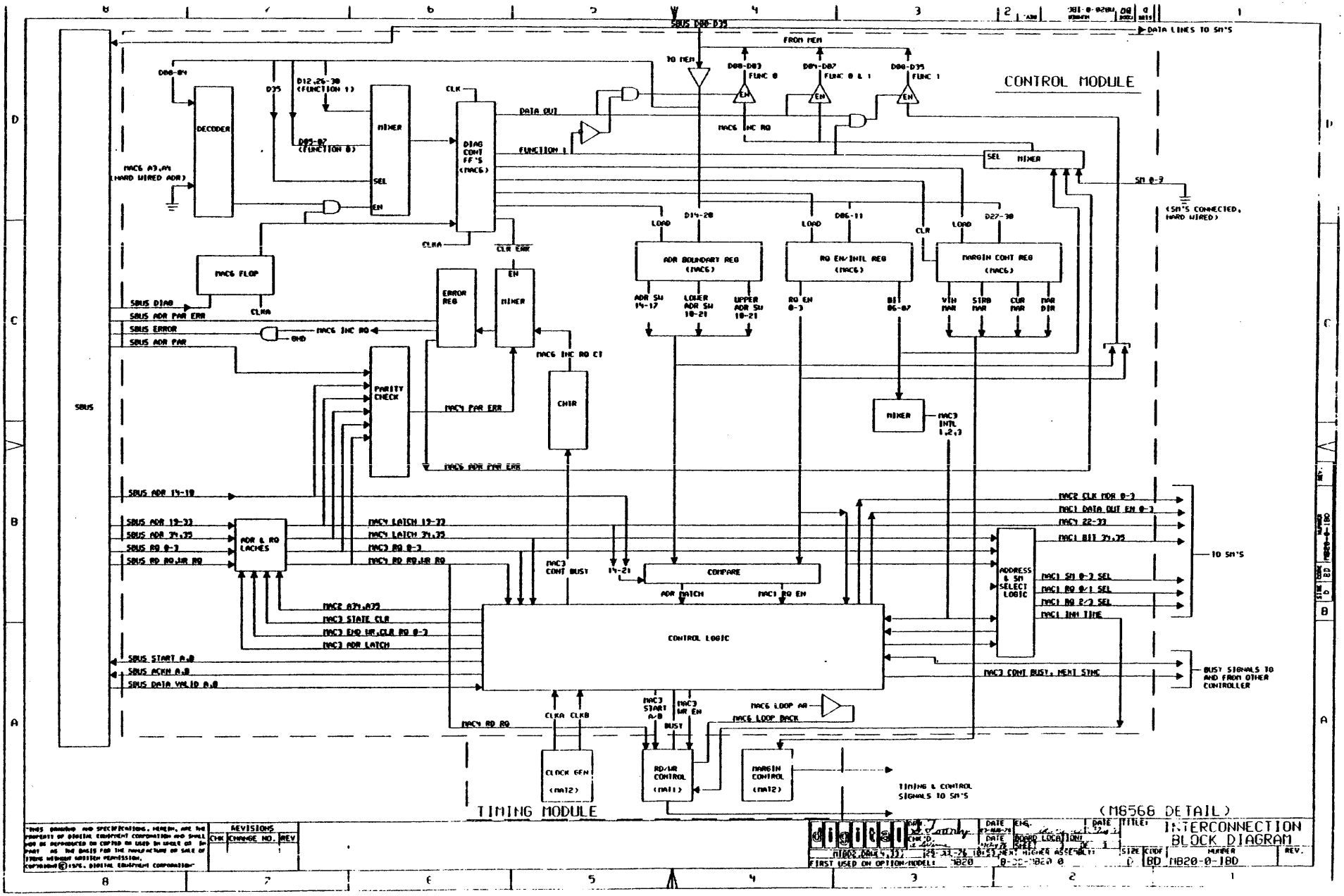
SHEET 1 OF 1



(STORAGE MODULE DETAIL)

REVISIONS

ECN	CHANGE NO.	REV

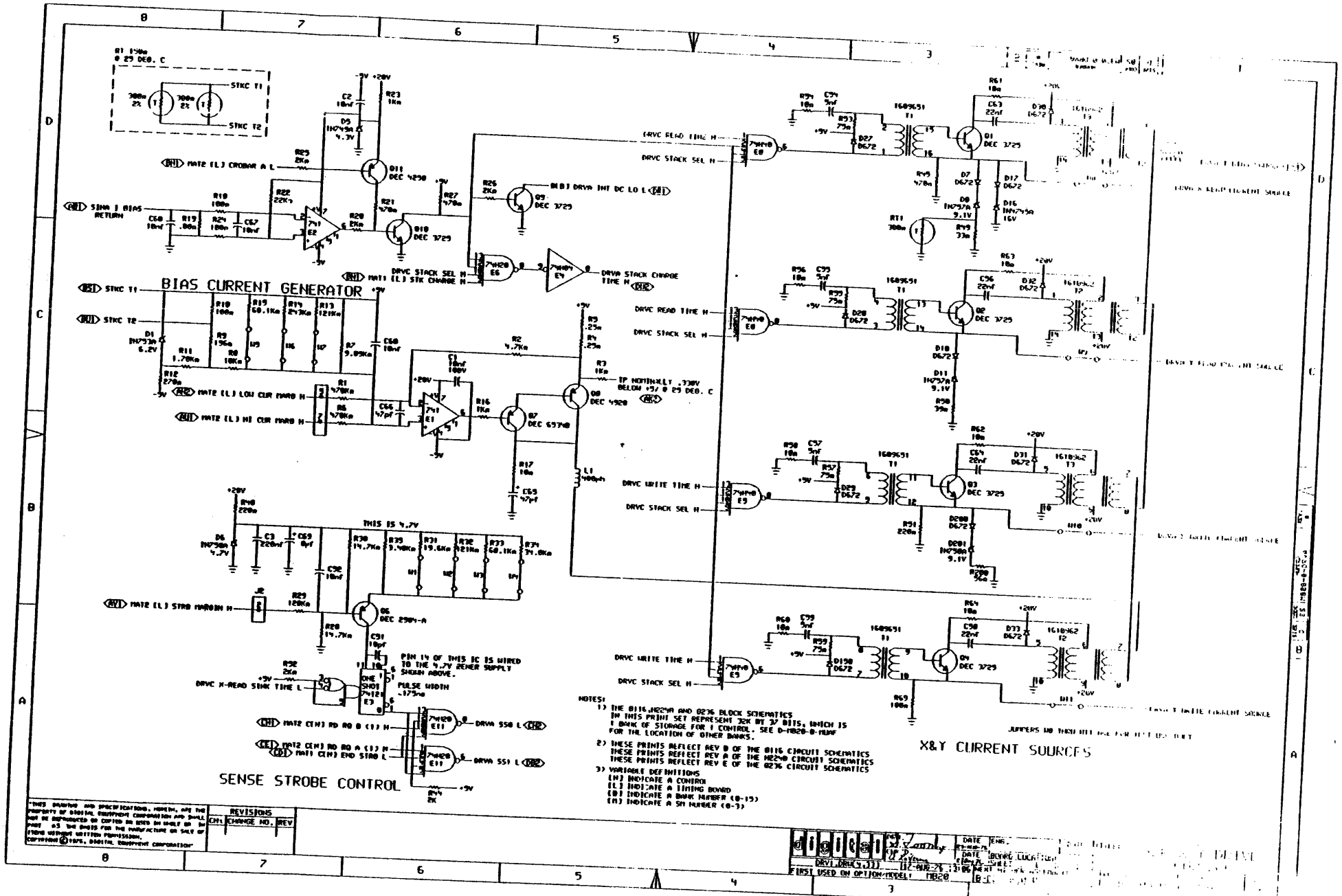


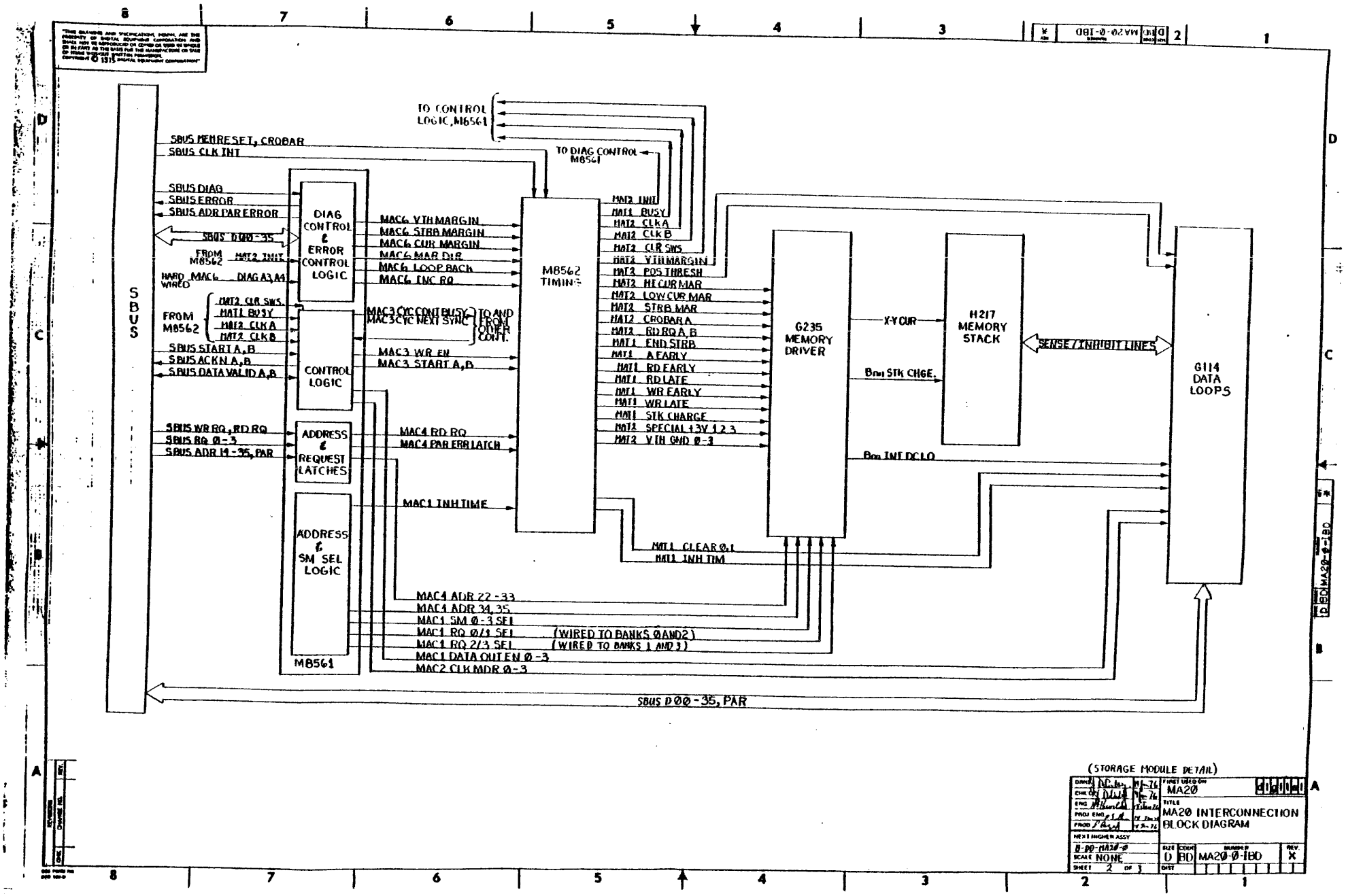
THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION. COPYRIGHT © 1975, DIGITAL EQUIPMENT CORPORATION

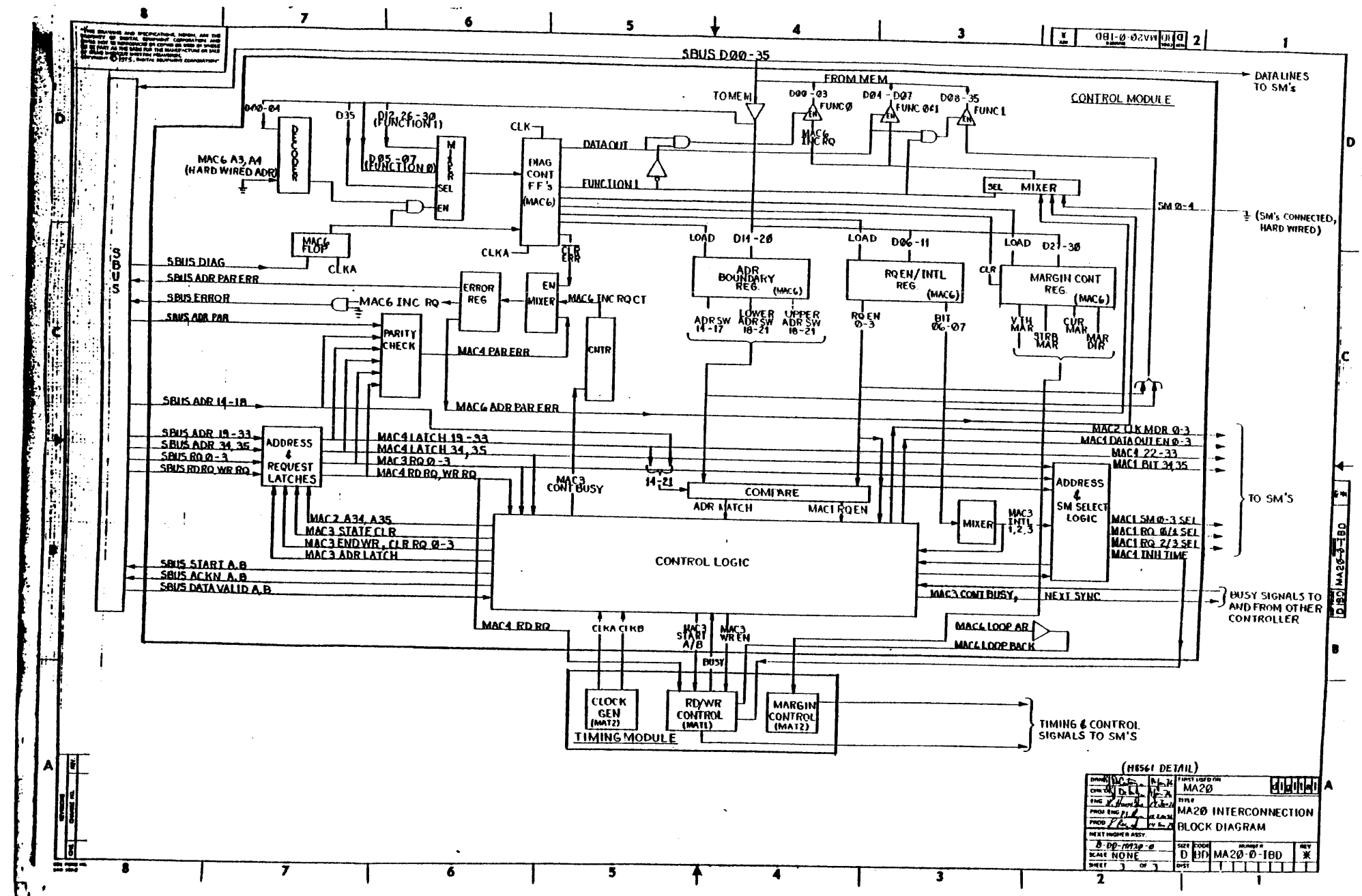
REVISIONS		
CHK	CHANGE NO.	REV

DRAWN: J. Carney 25-JUN-75
CHK'D:
ENG:
BOARD LOCATION:
SHEET 1 OF 1
FIRST USED ON OPTION-MODEL: M820

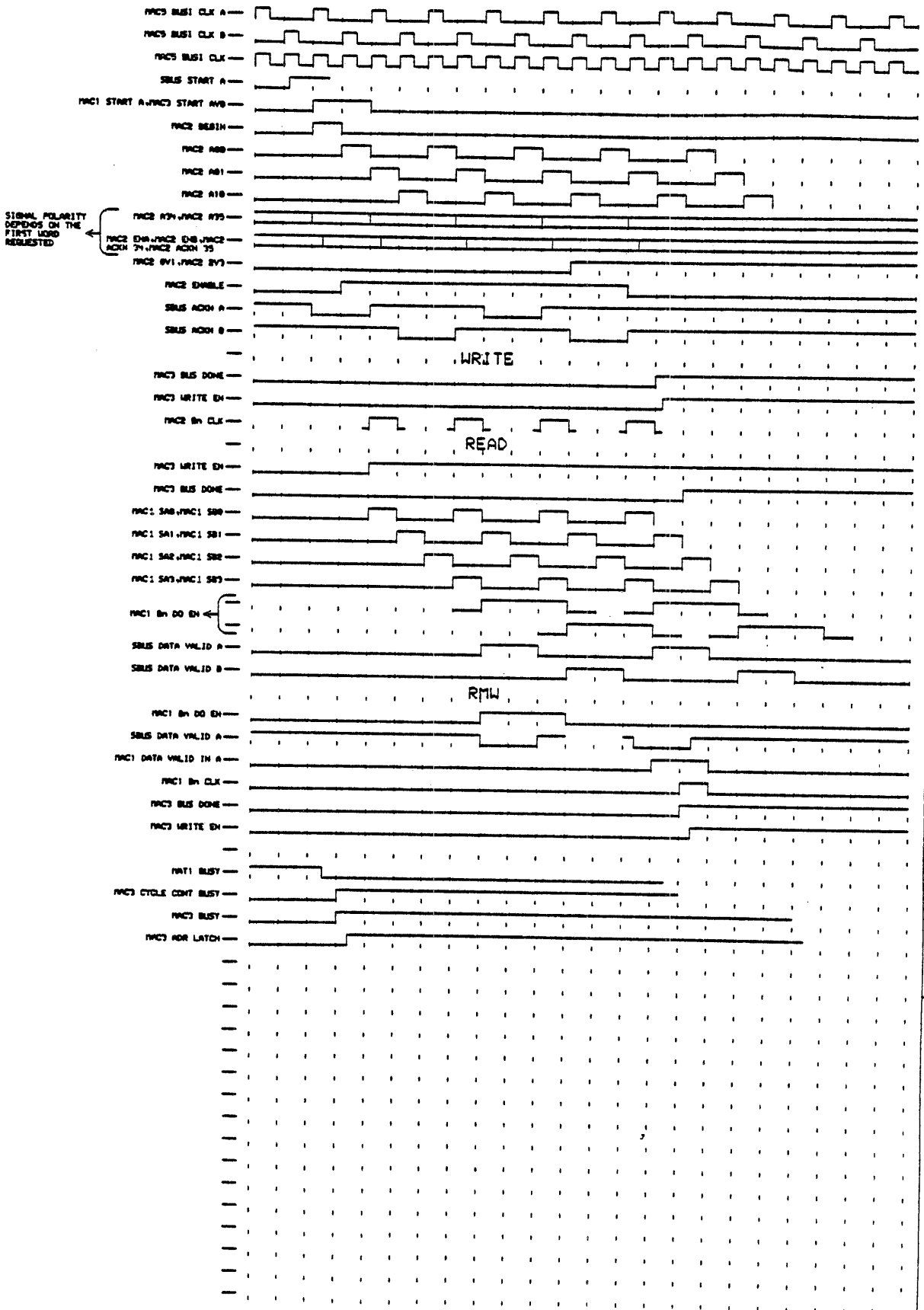
TITLE: INTERCONNECTION BLOCK DIAGRAM
SIZE D | CODE | NUMBER M820-0-1BD | REV.


BIAS CURRENT GENERATOR
SENSE STROBE CONTROL
X&Y CURRENT SOURCES
DRVI READ TIME H
DRVI STACK SEL H
DRVI WRITE TIME H
DRVA STACK CHANGE TIME H
DRVA SS0 L
DRVA SS1 L
NOTES:
1) THE 8116,M22YA AND 8236 BLOCK SCHEMATICS IN THIS PRINT SET REPRESENT 32K BY 32 BITS, WHICH IS 1 BANK OF STORAGE FOR 1 CONTROL. SEE D-MB20-0-MAP FOR THE LOCATION OF OTHER BANKS.
2) THESE PRINTS REFLECT REV D OF THE 8116 CIRCUIT SCHEMATICS
THESE PRINTS REFLECT REV A OF THE M22A0 CIRCUIT SCHEMATICS
THESE PRINTS REFLECT REV E OF THE 8236 CIRCUIT SCHEMATICS
3) VARIABLE DEFINITIONS
(N) INDICATE A CONTROL
(L) INDICATE A TIMING BOARD
(B) INDICATE A BANK NUMBER (0-15)
(M) INDICATE A SM NUMBER (0-3)


TO CONTROL LOGIC, M8561
TO DIAG CONTROL M8561
SBUS MEM RESET, CROBAR
SBUS CLK INT
SBUS DIAG
SBUS ERROR
SBUS ADR PAR ERROR
SBUS D00-35
FROM M8562 MAT2 INIT
HARD WIRED MAC6 DIAG A3, A4
FROM M8562
MAT2 CLR SWS
MAT1 BUSY
MAT2 CLK A
MAT2 CLK B
SBUS START A, B
SBUS ACK N A, B
SBUS DATA VALID A, B
SBUS WR RQ, RD RQ
SBUS RQ 0-3
SBUS ADR 14-35, PAR
SBUS
DIAG CONTROL & ERROR CONTROL LOGIC
CONTROL LOGIC
ADDRESS & REQUEST LATCHES
ADDRESS & SM SEL LOGIC
M8561
MAC6 VTH MARGIN
MAC6 STRB MARGIN
MAC6 CUR MARGIN
MAC6 MAR DIR
MAC6 LOOP BACK
MAC6 INC RQ
MAC3 CYC CONT BUSY
MAC3 CYC NEXT SYNC
TO AND FROM OTHER CONT.
MAC3 WR EN
MAC3 START A, B
MAC4 RD RQ
MAC4 PAR ERR LATCH
MAC1 INH TIME
M8562 TIMING
MAT2 INIT
MAT1 BUSY
MAT2 CLK A
MAT2 CLK B
MAT2 CLR SWS
MAT2 VTH MARGIN
MAT2 POS THRESH
MAT2 HI CUR MAR
MAT2 LOW CUR MAR
MAT2 STRB MAR
MAT2 CROBAR A
MAT2 RD RQ A, B
MAT1 END STRB
MAT1 A EARLY
MAT1 RD EARLY
MAT1 RD LATE
MAT1 WR EARLY
MAT1 WR LATE
MAT1 STK CHARGE
MAT2 SPECIAL +3V 1 2 3
MAT2 VTH GND 0-3
G235 MEMORY DRIVER
X-Y CUR
Bmx STK CHGE
Bmx INH DCLO
H217 MEMORY STACK
SENSE/INHIBIT LINES
G114 DATA LOOPS
MAT1 CLEAR 0,1
MAT1 INH TIM
MAC4 ADR 22-33
MAC4 ADR 34, 35
MAC1 SM 0-3 SEL
MAC1 RQ 0/1 SEL (WIRED TO BANKS 0 AND 2)
MAC1 RQ 2/3 SEL (WIRED TO BANKS 1 AND 3)
MAC1 DATA OUT EN 0-3
MAC2 CLK MDR 0-3
SBUS D00-35, PAR
(STORAGE MODULE DETAIL)
MA20
MA20 INTERCONNECTION BLOCK DIAGRAM
D-BD-MA20-0
SCALE NONE
SHEET 2 OF 3
SIZE D
CODE BD
MA20-0-IBD
REV X


SBUS D00-35
DATA LINES TO SM's
FROM MEM
TO MEM
D00-03
D04-D07
D08-35
FUNC 0
FUNC 01
FUNC 1
CONTROL MODULE
MAC6 INC RQ
D00-04
DECODE
D35
D12,26-30 (FUNCTION 1)
D05-07 (FUNCTION 0)
MAC6 A3, A4 (HARD WIRED ADR)
MASK SEL EN
CLK
MAG CONT FF's (MAC6)
DATA OUT
FUNCTION 1
SEL MIXER
SM 0-4
¼ (SM's CONNECTED, HARD WIRED)
MAC6 FLOP
CLKA
CLR ERR
LOAD D14-20
LOAD D00-11
LOAD D21-30
ADR BOUNDARY REG. (MAC6)
RQ EN/INTL REG. (MAC6)
MARGIN CONT REG. (MAC6)
CLR
ADR SW 14-17
LOWER ADR SW 18-21
UPPER ADR SW 18-21
RQ EN 0-3
BIT 06-07
VTH MAR
STRB MAR
CUR MAR
MAR DIR
SBUS
SBUS DIAG
SBUS ADR PAR ERR
SBUS ERROR
SBUS ADR PAR
ERROR REG
EN MIXER
MAC6 INC RQ CT
MAC6 INC RQ
PARITY CHECK
MAC4 PAR ERR
CNTR
SBUS ADR 14-18
MAC6 ADR PAR ERR
SBUS ADR 19-33
SBUS ADR 34, 35
SBUS RQ 0-3
SBUS RD RQ, WR RQ
ADDRESS & REQUEST LATCHES
MAC4 LATCH 19-33
MAC4 LATCH 34, 35
MAC3 RQ 0-3
MAC4 RD RQ, WR RQ
MAC3 CONT BUSY
14-21
COMPARE
ADR MATCH
MAC1 RQ EN
MAC2 CLK MDR 0-3
MAC1 DATA OUT EN 0-3
MAC1 22-33
MAC1 BIT 34,35
ADDRESS & SM SELECT LOGIC
MAC1 SM 0-3 SEL
MAC1 RQ 0/1 SEL
MAC1 RQ 2/3 SEL
MAC1 INH TIME
TO SM'S
MIXER
MAC3 INTL 1,2,3
MAC2 A34, A35
MAC3 STATE CLR
MAC3 END WR, CLR RQ 0-3
MAC3 ADR LATCH
CONTROL LOGIC
SBUS START A,B
SBUS ACKN A,B
SBUS DATA VALID A,B
MAC3 CONT BUSY, NEXT SYNC
BUSY SIGNALS TO AND FROM OTHER CONTROLLER
MAC4 RD RQ
CLKA CLKB
MAC3 START A/B
MAC3 WR EN
BUSY
MAC4 LOOP AR
MAC4 LOOP BACK
CLOCK GEN (MAT2)
RD/WR CONTROL (MAT1)
MARGIN CONTROL (MAT2)
TIMING MODULE
TIMING & CONTROL SIGNALS TO SM'S
(H8561 DETAIL)
MA20
MA20 INTERCONNECTION BLOCK DIAGRAM
SCALE NONE
SIZE D
CODE 80
MA20-0-1B0
REV *
SHEET 3 OF 3

MA20 INTERLEAVE MODE 3 (4 WAY),4 WORDS (EXCEPT 1 WORD FOR RMW)



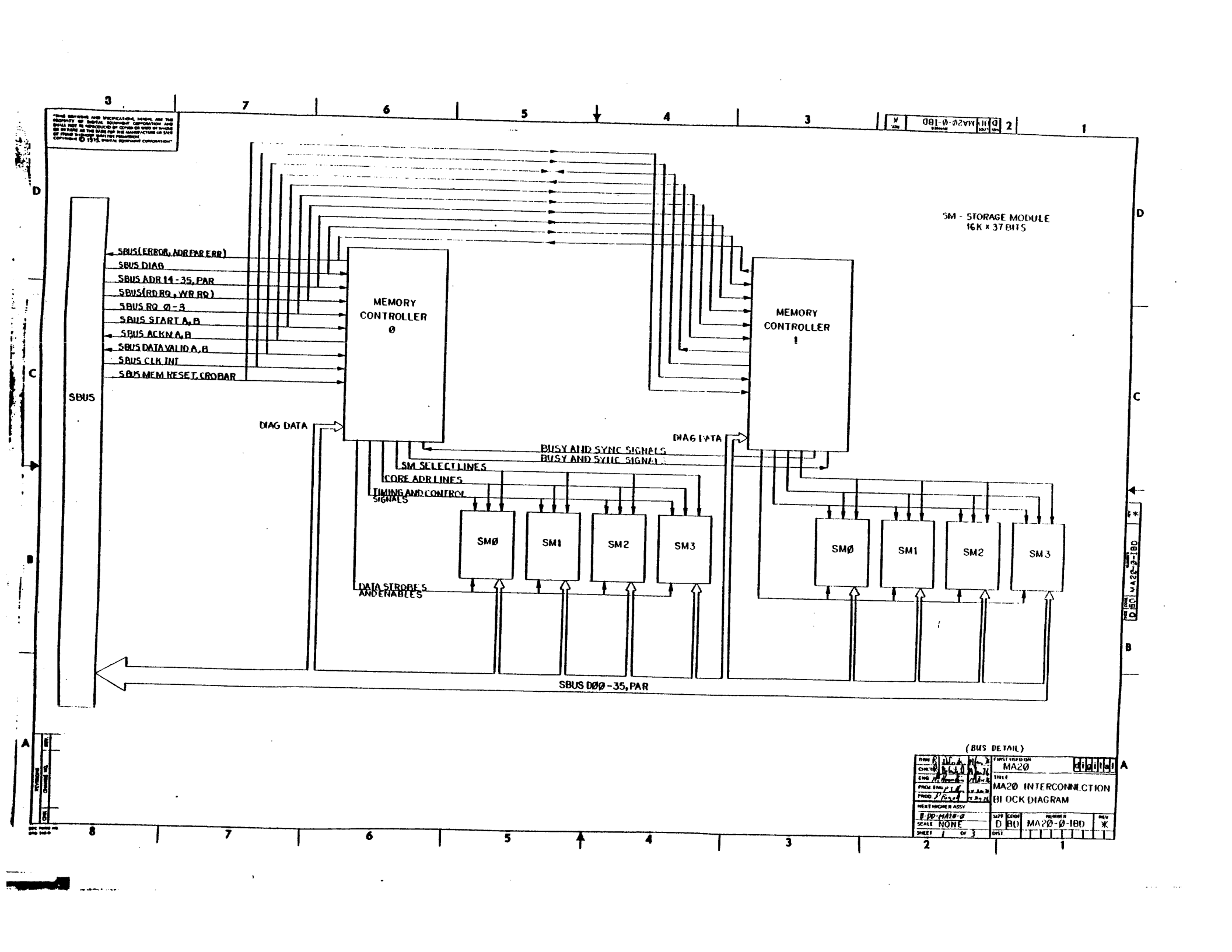
REVISIONS			
OK	CHANGE NO.	REV

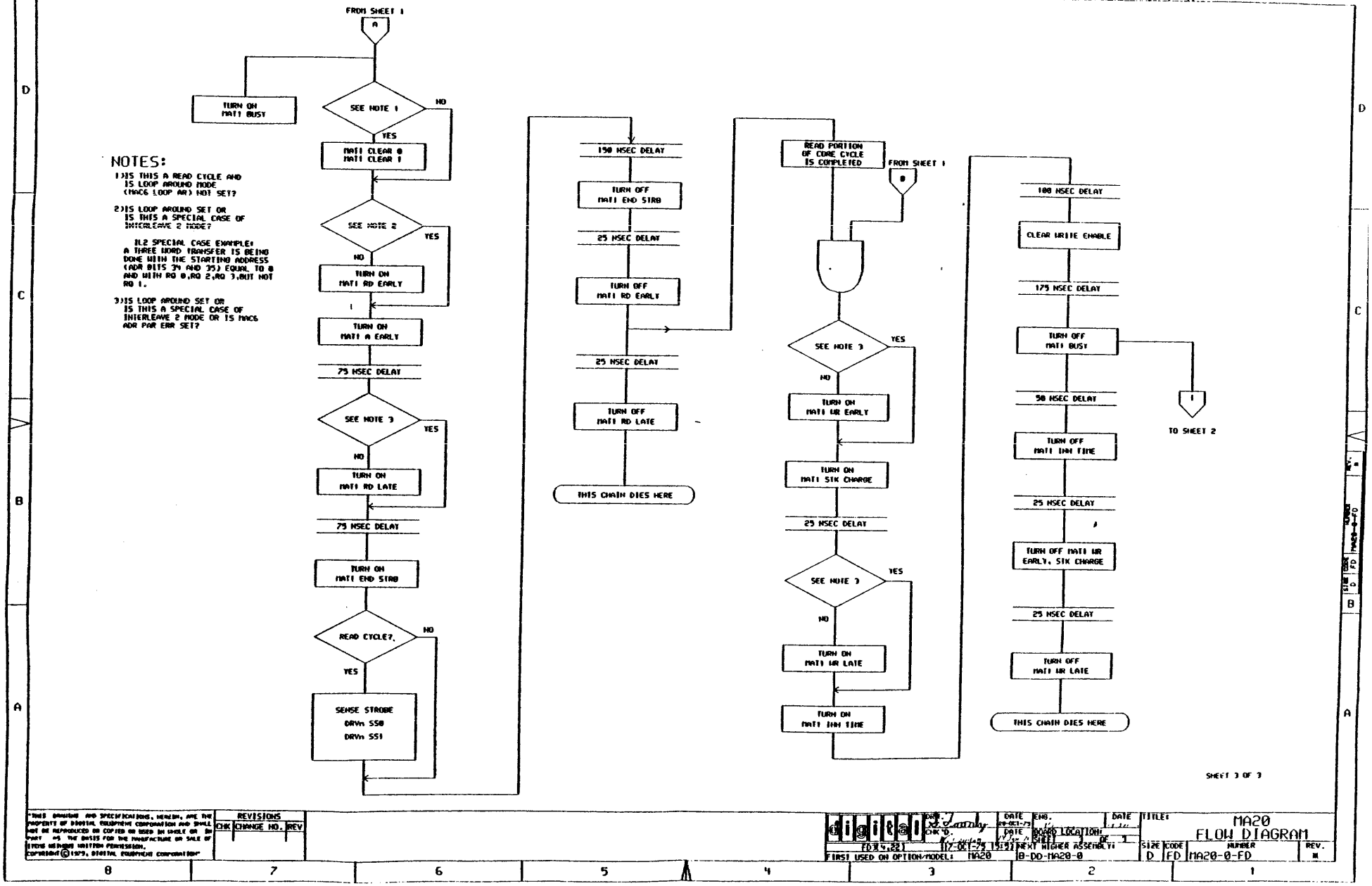
digital

DRAWN	DATE	ENG.	DATE
CHK'D	DATE	BOARD LOCATION:	
DESIGN APPROVED	27-OCT-75	NEXT HIGHER ASSEMBLY:	
FIRST USED ON PT	MA20	B-DD-MA20-0	

TITLE: MA20 TIMING (INTERLEAVE 3)

SIZE	CODE	NUMBER	REV.
D	TD	MA20-0-TD	


SM - STORAGE MODULE
16K x 37 BITS
SBUS(ERROR, ADR PAR ERR)
SBUS DIAG
SBUS ADR 14-35, PAR
SBUS(RD RQ, WR RQ)
SBUS RQ Ø-3
SBUS START A, B
SBUS ACK N A, B
SBUS DATA VALID A, B
SBUS CLK INT
SBUS MEM RESET, CROBAR
SBUS
MEMORY CONTROLLER Ø
MEMORY CONTROLLER 1
DIAG DATA
DIAG DATA
BUSY AND SYNC SIGNALS
BUSY AND SYNC SIGNALS
SM SELECT LINES
CORE ADR LINES
TIMING AND CONTROL SIGNALS
SMØ
SM1
SM2
SM3
SMØ
SM1
SM2
SM3
DATA STROBES AND ENABLES
SBUS DØØ-35, PAR
(BUS DETAIL)
MA2Ø
MA2Ø INTERCONNECTION BLOCK DIAGRAM
MA2Ø-Ø-1BD
D
BD
NONE



NOTES:

1)IS THIS A READ CYCLE AND IS LOOP AROUND MODE (MACG LOOP AR) NOT SET?

2)IS LOOP AROUND SET OR IS THIS A SPECIAL CASE OF INTERLEAVE 2 MODE?

IL2 SPECIAL CASE EXAMPLE: A THREE WORD TRANSFER IS BEING DONE WITH THE STARTING ADDRESS (ADR BITS 34 AND 35) EQUAL TO 0 AND WITH RQ 0,RQ 2,RQ 3,BUT NOT RQ 1.

3)IS LOOP AROUND SET OR IS THIS A SPECIAL CASE OF INTERLEAVE 2 MODE OR IS MACG ADR PAR ERR SET?

SHEET 3 OF 3

REVISIONS

CHK	CHANGE NO.	REV

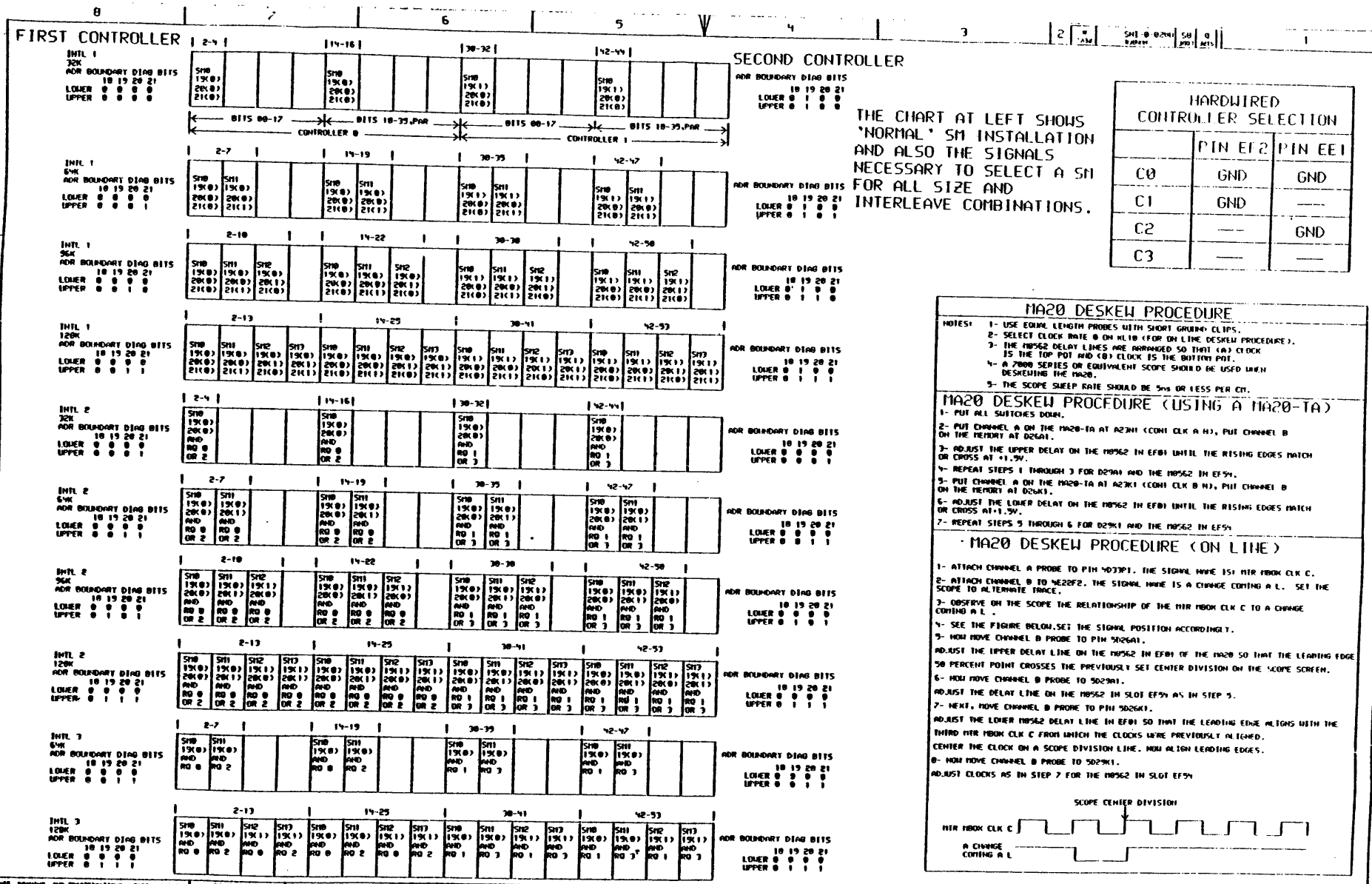
digital

DRN	CHK'D	DATE
FD34,221	17-OCT-79 15:51	
FIRST USED ON OPTION/MODEL: MA20		

ENG. DATE 29-OCT-79
BOARD LOCATION
SHEET 3 OF 3
NEXT HIGHER ASSEMBLY: B-DD-MA20-0

TITLE: MA20 FLOW DIAGRAM

SIZE	CODE	NUMBER	REV.
D	FD	MA20-0-FD	H



THE CHART AT LEFT SHOWS 'NORMAL' SM INSTALLATION AND ALSO THE SIGNALS NECESSARY TO SELECT A SM FOR ALL SIZE AND INTERLEAVE COMBINATIONS.

HARDWIRED CONTROLLER SELECTION

	PIN EF2	PIN EE1
C0	GND	GND
C1	GND	—
C2	—	GND
C3	—	—

MA20 DESKEW PROCEDURE

NOTES:
1- USE EQUAL LENGTH PROBES WITH SHORT GROUND CLIPS.
2- SELECT CLOCK RATE 0 ON KL10 (FOR ON LINE DESKEW PROCEDURE).
3- THE M9562 DELAY LINES ARE ARRANGED SO THAT (A) CLOCK IS THE TOP POT AND (B) CLOCK IS THE BOTTOM POT.
4- A 7000 SERIES OR EQUIVALENT SCOPE SHOULD BE USED WHEN DESKEWING THE MA20.
5- THE SCOPE SWEEP RATE SHOULD BE 5ns OR LESS PER CM.

MA20 DESKEW PROCEDURE (USING A MA20-TA)

1- PUT ALL SWITCHES DOWN.
2- PUT CHANNEL A ON THE MA20-TA AT A23N1 (CONT CLK A H), PUT CHANNEL B ON THE MEMORY AT D26A1.
3- ADJUST THE UPPER DELAY ON THE M9562 IN EF01 UNTIL THE RISING EDGES MATCH OR CROSS AT +1.5V.
4- REPEAT STEPS 1 THROUGH 3 FOR D29A1 AND THE M9562 IN EF54.
5- PUT CHANNEL A ON THE MA20-TA AT A23K1 (CONT CLK B H), PUT CHANNEL B ON THE MEMORY AT D26K1.
6- ADJUST THE LOWER DELAY ON THE M9562 IN EF01 UNTIL THE RISING EDGES MATCH OR CROSS AT+1.5V.
7- REPEAT STEPS 5 THROUGH 6 FOR D29K1 AND THE M9562 IN EF54

MA20 DESKEW PROCEDURE (ON LINE)

1- ATTACH CHANNEL A PROBE TO PIN 4D33P1. THE SIGNAL NAME IS: MIR MBOX CLK C.
2- ATTACH CHANNEL B TO 4E22F2. THE SIGNAL NAME IS A CHANGE COMING A L. SET THE SCOPE TO ALTERNATE TRACE.
3- OBSERVE ON THE SCOPE THE RELATIONSHIP OF THE MIR MBOX CLK C TO A CHANGE COMING A L .
4- SEE THE FIGURE BELOW.SET THE SIGNAL POSITION ACCORDINGLY.
5- NOW MOVE CHANNEL B PROBE TO PIN 5026A1.
ADJUST THE UPPER DELAY LINE ON THE M9562 IN EF01 OF THE MA20 SO THAT THE LEADING EDGE 50 PERCENT POINT CROSSES THE PREVIOUSLY SET CENTER DIVISION ON THE SCOPE SCREEN.
6- NOW MOVE CHANNEL B PROBE TO 5029A1.
ADJUST THE DELAY LINE ON THE M9562 IN SLOT EF54 AS IN STEP 5.
7- NEXT, MOVE CHANNEL B PROBE TO PIN 5026K1.
ADJUST THE LOWER M9562 DELAY LINE IN EF01 SO THAT THE LEADING EDGE ALIGNS WITH THE THIRD MIR MBOX CLK C FROM WHICH THE CLOCKS WERE PREVIOUSLY ALIGNED.
CENTER THE CLOCK ON A SCOPE DIVISION LINE. NOW ALIGN LEADING EDGES.
8- NOW MOVE CHANNEL B PROBE TO 5029K1.
ADJUST CLOCKS AS IN STEP 7 FOR THE M9562 IN SLOT EF54

REVISIONS

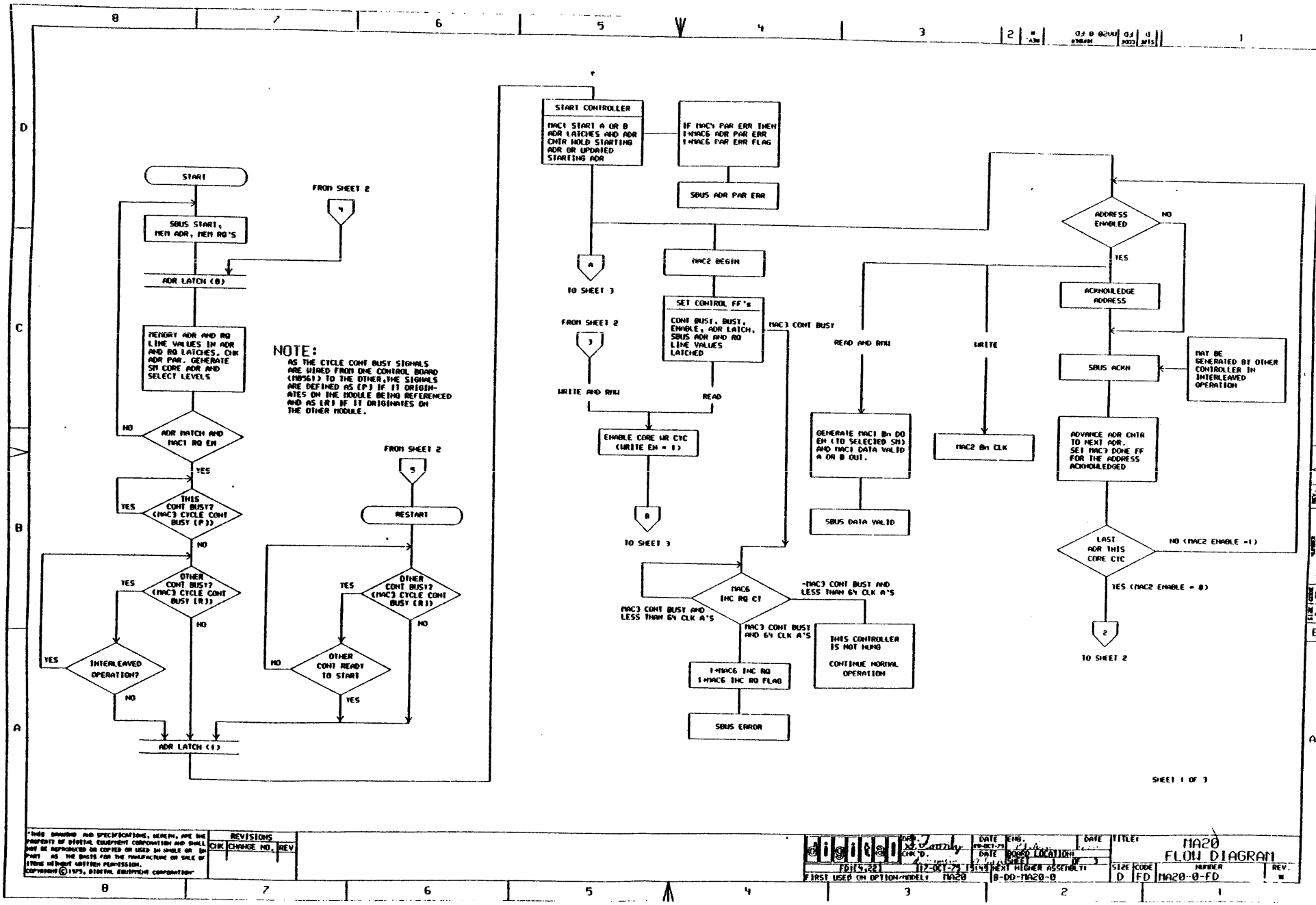
CHR	CHANGE NO.	REV.

digital — Maynard

DRAWN BY	DATE	ENG.	DATE
CHK'D	DATE	BOARD LOCATION	
INT.DRW 5,510	19-DEC-79	SHEET 1 OF	
FIRST USED ON OPTION/MODEL	MA20	NEXT HIGHER ASSEMBLY	B-DD-MA20-0

TITLE: MA20 INSTRUCTION /SETUP CHARTS

SIZE	CODE	NUMBER	REV.
D	BS	MA20-0-INS	H



REVISIONS

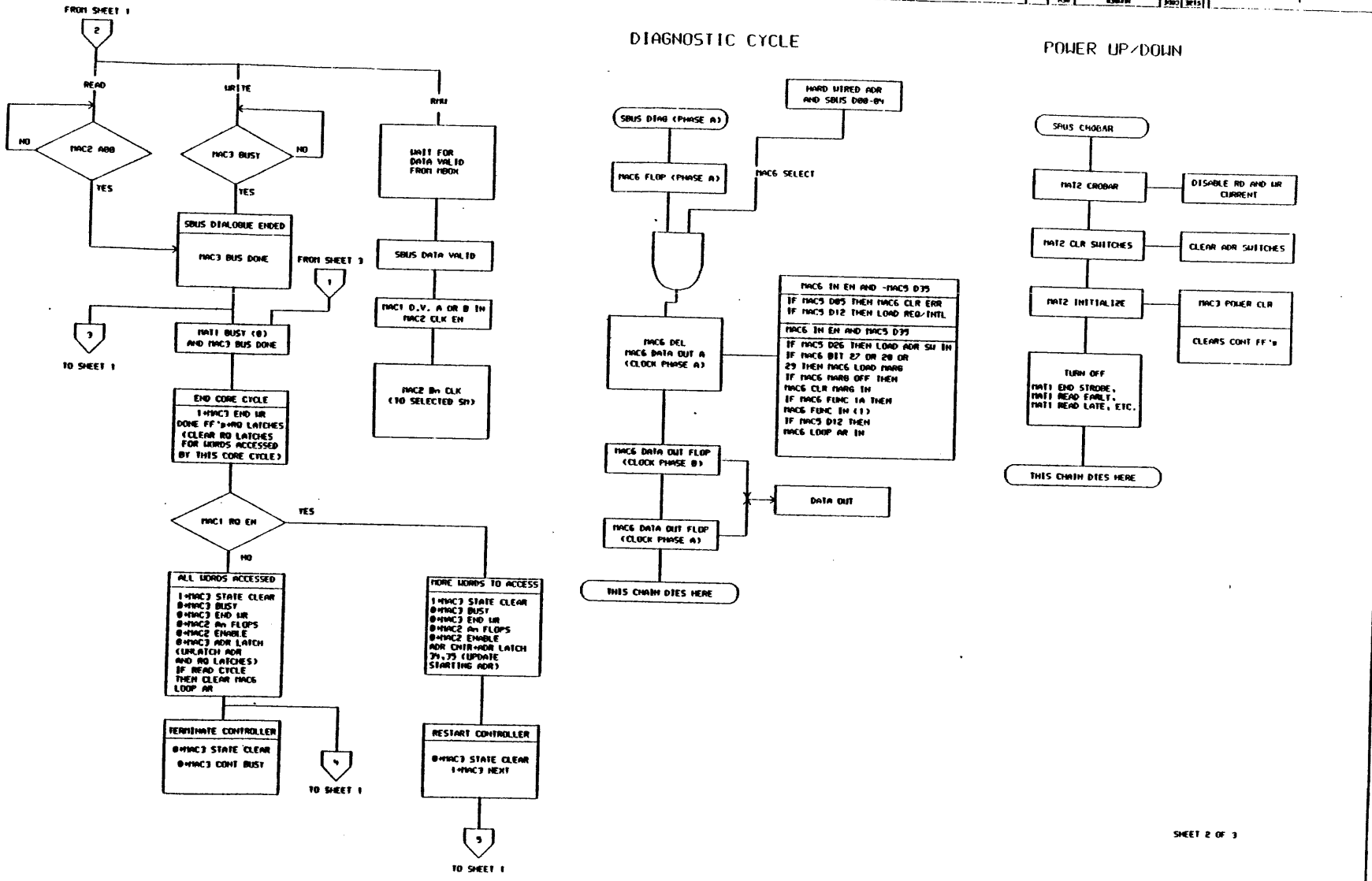
CHK	CHANGE NO.	REV

digital

DRN. J. Zamby
CHK'D.
DATE 12-OCT-75
DATE
DATE
FDI(5,22)
FIRST USED ON OPTION/MODEL: MA20
NEXT HIGHER ASSEMBLY: B-DD-MA20-0
BOARD LOCATION: SHEET 1 OF 3

TITLE: MA20 FLOW DIAGRAM

SIZE	CODE	NUMBER	REV
D	FD	MA20-0-FD	



REVISIONS		
CHK	CHANGE NO.	REV

CHK: J. Landry	DATE 20-OCT-75	END:	DATE
CHK: D.	DATE	BOARD LOCATION:	
FD2(1,22)	17-OCT-75 15:51	NEXT HIGHER ASSEMBLY:	
FIRST USED ON OPTION/MODEL: MA20		8-DD-MA20-0	

TITLE: MA20 FLOW DIAGRAM

SIZE	CODE	NUMBER	REV.
D	FD	MA20-0-FD	N