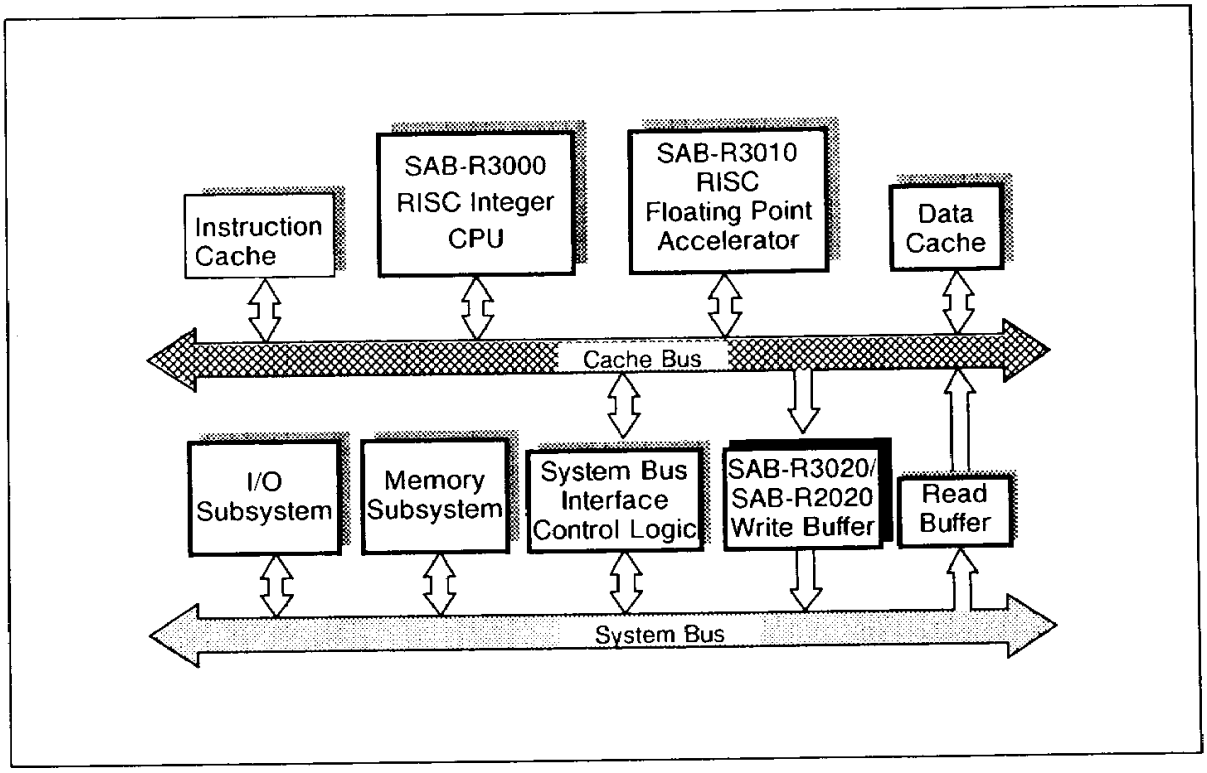
SIEMENS

SAB-R3020/SAB-R2020
Write Buffer

Advance Information

- Temporary storage buffers to enhance the performance of the SAB-R3000 (SAB-R2000A) RISC microprocessor.
- Decouples the SAB-R3000 (SAB-R2000A) from slow main memory.
- Enables write operations from the SAB-R3000 (SAB-R2000A) to occur at processor speeds.
- Supports big endian and little endian byte-order addressing.
- Each Write Buffer has four locations to handle an 8-bit address slice and a 9-bit data slice (including a parity bit)
- Fully pin- and functionally compatible to all R3020/R2020 write buffers from other manufacturers.
- Plastic Package PL-CC-68



Ordering Information

Type	Ordering code	Package	Function
SAB-R2020-16-N	Q67120-C556	PL-CC-68	Write Buffer, 16.67 MHz
SAB-R3020-25-N	Q67120-C557	PL-CC-68	Write Buffer, 25 MHz

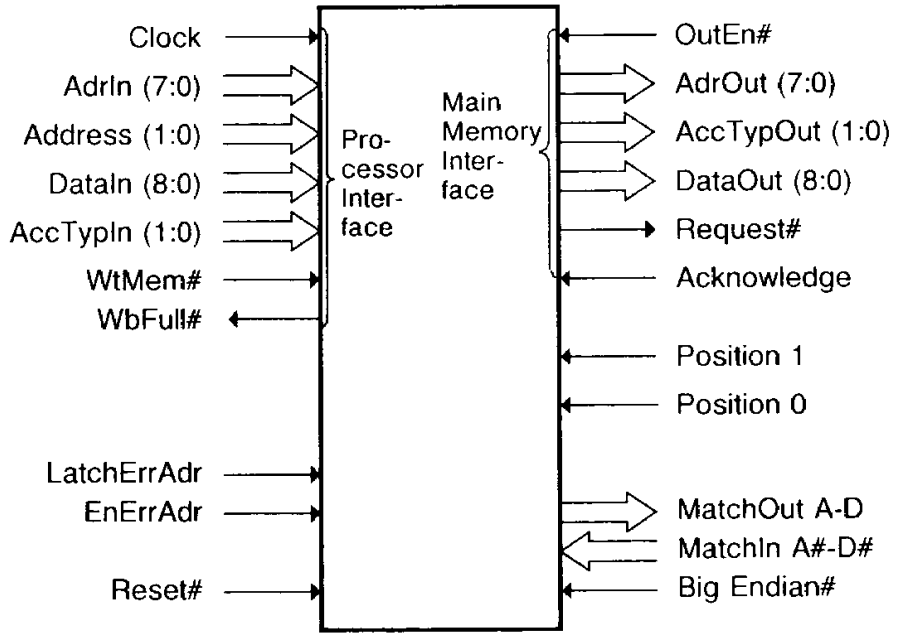
Introduction

The SAB-R3020/SAB-R2020 write buffer enhances performance of MIPS architecture based systems by allowing the processor to perform write operations during run cycles instead of stalling the pipeline. Each device handles an 8-bit slice of address and a 9-bit slice of data (one parity bit per byte). Four write buffers are used per system to provide four-deep buffering of 32 bits of address and 36 bits of data and parity.

Pin Names

Clock	System Clock
AdrIn (7:0)	Address Bus In
Address (1:0)	Byte Address
DataIn (8:0)	Data Bus In (includes one parity bit)
AccTypIn (1:0)	Access Type In
WtMem#	Main Memory Write
WbFull#	Write Buffer Full
LatchErrAdr	Latch Error Address
EnErrAdr	Enable Error Address
Reset#	Initializiation
OutEn#	Enable Tristate Outputs
AdrOut (7:0)	Address Bus Out
AccTypOut (1:0)	Access Type Out
DataOut (8:0)	Data Bus Out (including one parity bit)
Request#	Request access to Main Memory
Acknowledge	Acknowledge capture of Data
Position 1, Position 0	Position of Bytes
MatchOut A-D	Match Out to avoid Read/Write Conflicts
MatchIn A#-D#	Match In for Byte Gathering
Big Endian#	Big/Little-Endian Selection

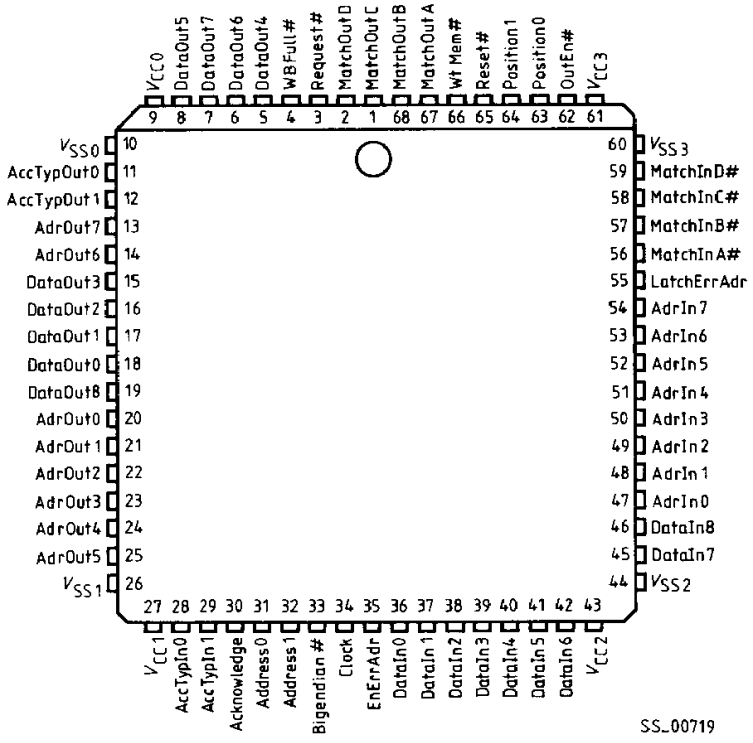
Figure 1
Logic Symbol



Note: "#" signifies an active low signal.

Pin Configuration

Figure 2
PL-CC-68 (Top View)



Pin Definitions and Functions

Symbol	Pin Number	Input (I) Output (O)	Function
Clock	34	I	An inverted version of the SysOut# signal from the CPU that synchronizes data transfers. The write buffer uses the trailing edge of Clock to latch the contents of the AdrLo bus, and the leading edge to latch the contents of the Data and Tag buses.
DataIn (8:0)	46, 45, 42, 41, 40, 39, 38, 37, 36	I	Nine input data lines from the processor's data Bus (eight bits of data and one bit of parity).
AdrIn (7:0)	54, 53, 52, 51, 50, 49, 48, 47	I	Eight input address lines from the CPU. The address lines are taken from the AdrLo and Tag buses.
Address (1:0)	32, 31	I	The least two significant address bits from the CPU. These two address bits must be connected to all four write buffers and are used in conjunction with the Access Type signals, the Position signals, and the BigEndian signal to determine which byte(s) in a word are being written into a particular write buffer.
AccTypIn (1:0)	29, 28	I	The access type signals from the CPU specifying the size of data transfer: word, tri-byte, half-word, or byte.
WtMem#	66	I	This input is connected to the MemWr# signal of the processor, which is asserted whenever the CPU performs a store operation.
Request#	3	O	This signal is used to request access to main memory. Request# may also be tied to the CpCond0 input of the CPU. Since Request# is asserted only when the write buffer contains data, software can determine if a previous write operation (for example, to an I/O device) has been completed before initiating a read to that device.

Pin Definitions and Functions (cont'd)

Symbol	Pin Number	Input (I) Output (O)	Function
WbFull#	4	O	The write buffer asserts this signal to the CPU WrBusy# input when it cannot accept more data. The processor performs a write busy stall if data must be stored while WbFull# is asserted.
AdrOut (7:0)	13, 14, 25, 24, 23, 22, 21, 20	O*)	Eight address lines output from each write buffer.
DataOut (8:0)	19, 7, 6, 8, 5, 15, 16, 17, 18	O*)	Nine output data lines from each write buffer (one bit parity).
AccTypOut (1:0)	12, 11	O*)	The access type signals from the write buffer specifying the size of a data access: word, tri-byte, half-word, or byte.
OutEn#	62	I	The memory controller asserts this write buffer input to enable the tri-state outputs of the write buffer address and data signals.
Acknowledge	30	I	The main memory system asserts this signal when it has captured the data presented by the write buffer on the DataOut lines.
Position1, Position0	64, 63	I	These signals (in conjunction with Big Endian#) determine how each write buffer decodes Address I/O and AccTyp I/O to ensure proper storage of the data inputs.
Big Endian#	33	I	When asserted, byte 0 is the leftmost, most significant byte (big-endian). When deasserted, byte 0 is the rightmost, least significant byte (little-endian).
MatchOutA-D	67, 68, 1, 2	O	Outputs to main memory controller logic used to resolve conflicts when the processor reads or writes to memory.

*) = Tri-state Output
= Active Low

Pin Definitions and Functions (cont'd)

Symbol	Pin Number	Input (I) Output (O)	Function
MatchInA#-D#	56, 57, 58, 59	I	These four active low inputs are used by internal byte gathering logic to combine bytes common to a word, thus increasing memory bandwidth.
EnErrAdr	35	I	When asserted, the contents of an internal error latch are output to the DataOut bus.
LatchErrAdr	55	I	When asserted, the value currently available to the address outputs of the write buffer is latched into an internal error latch.
Reset#	65	I	Used to initialize the write buffer to a known state and clear the contents of its registers.
V_{CC} (3:0)	61, 43, 27, 9	I	Power Supply
V_{SS} (3:0)	60, 44, 26, 10	I	Ground

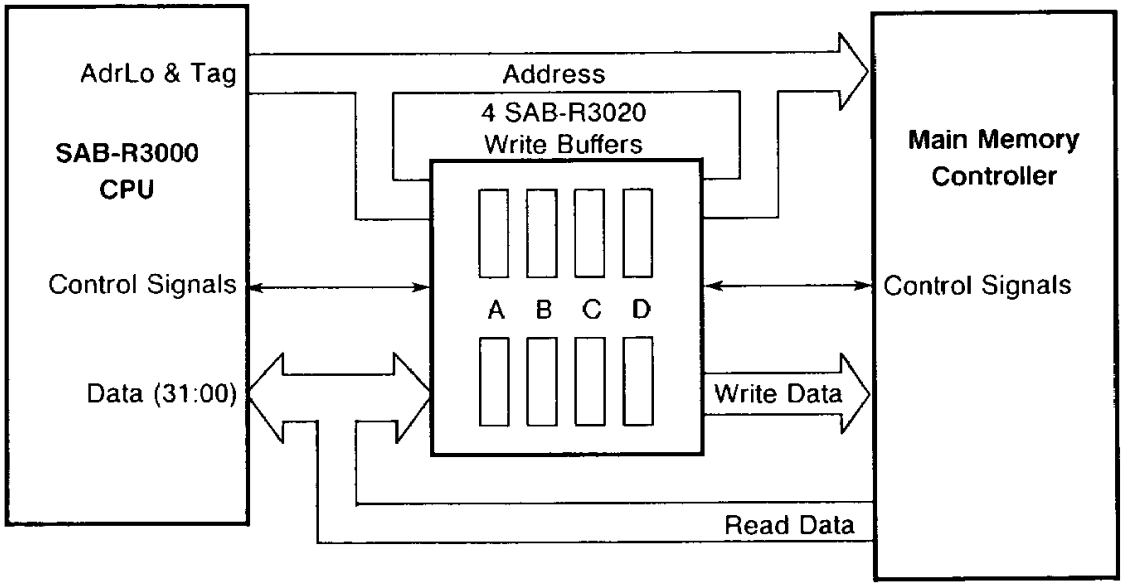
Operation

When the SAB-R3000 (SAB-R2000A) performs a write operation, the write buffer captures the output data and its address (including the access type bits). The write buffer can hold up to four data-address sets while it waits to pass the data on to main memory. Transfers from the processor to the write buffers occur synchronously at the cycle rate of the processor and the write buffer signals the processor if it is unable to accept data. The write buffer also provides a set of handshake signals to communicate with a main memory controller to coordinate the transfer of write data to main memory.

System Interface

Figure 3 shows the functional position of the write buffer in a SAB-R3000 (SAB-R2000A) based system.

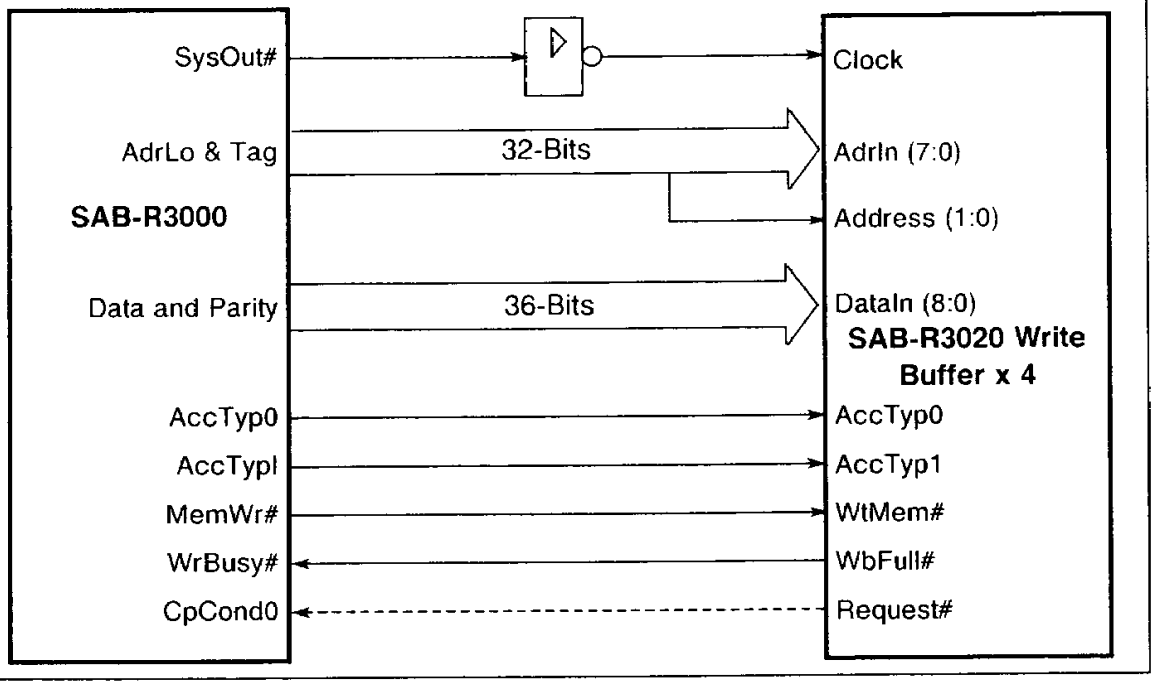
Figure 3
The SAB-R3020/SAB-R2020 in an SAB-R3000 (SAB-R2000A) System



Write Buffer-CPU Interface

Figure 4 details the interface between the CPU and the write buffer. The description assumes that four write buffers will be used to implement a 32-bit, buffered interface. The AdrLo bus and Tag bus bits from the CPU are both connected to form the 32-bit physical address that is captured by the write buffer. Thirty-two bits of data, four bits of parity, and two access type bits are also captured by the write buffer.

Figure 4
SAB-R3020/SAB-R3000 Interface (SAB-R2020/SAB-R2000A)



Data and Address Connections

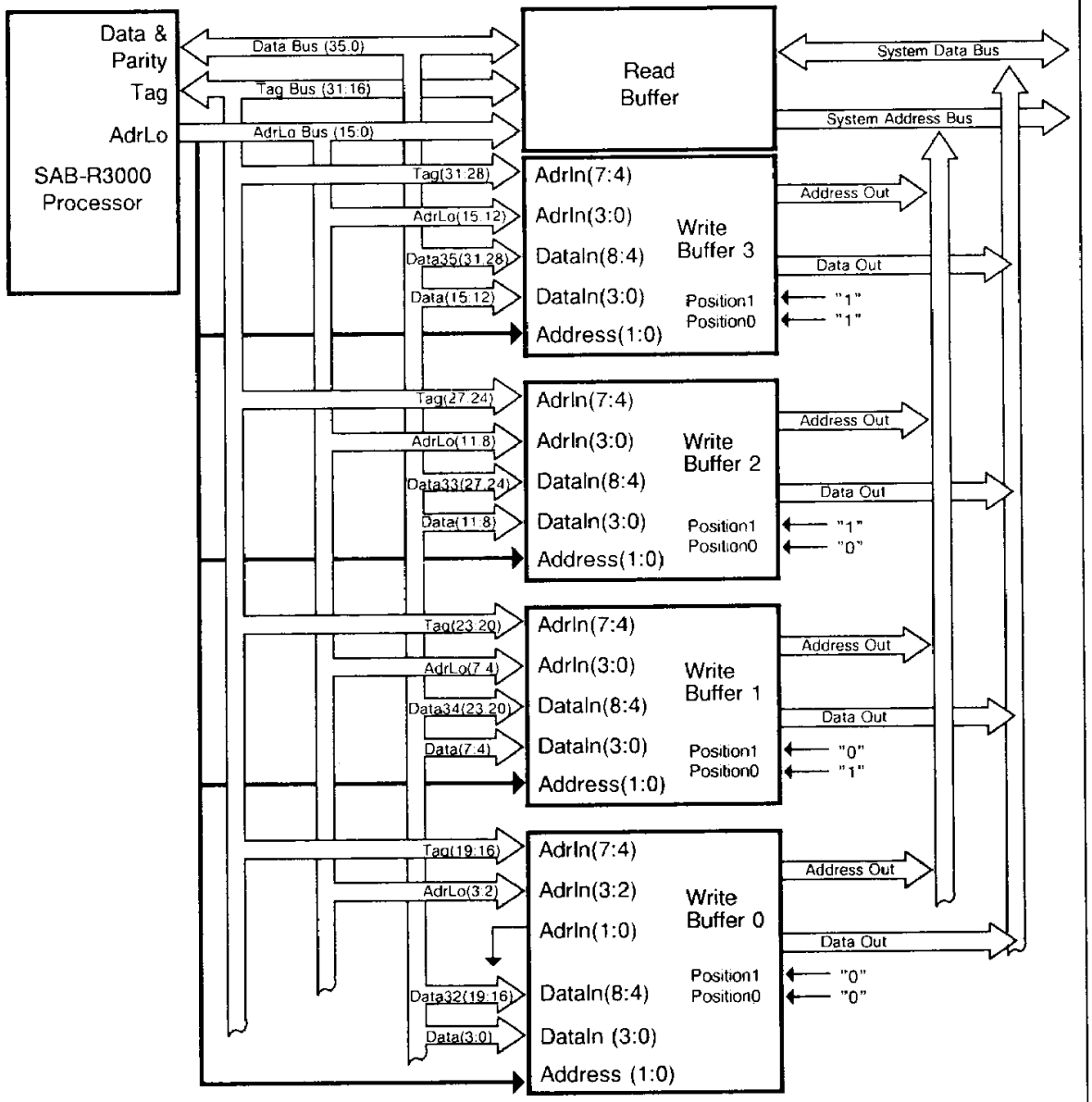
Figure 5 illustrates the address and data connection between four write buffers and the RISC CPU.

Each write buffer has eight address inputs (AdrIn7:0). The four low order bits (AdrIn3:0) are clocked into the device on the trailing edge of the Clock signal and taken from the SAB-R3000 (SAB-R2000A) AdrLo bus. The four high order bits (AdrIn7:4) are clocked into the device on the rising edge of the clock signal and taken from the SAB-R3000 (SAB-R2000A) Tag bus.

Each device also has separate inputs (Address1, Address0) for the two low order bits from the AdrLo bus. These bits act as a byte pointer within the write buffer. Note in figure 5 that the two low order AdrIn inputs (AdrIn1:0) to the write buffer device 0 are connected to ground since the Address1, Address0 inputs already supply these bits to the device.

The write buffer has nine data inputs that are clocked into the device on the leading edge of the clock signal and taken from the SAB-R3000 (SAB-R2000A) data bus. Note that the data bits assigned to each device correspond to the address bits. This arrangement is required since data selection is dependent on a combination of the AccTyp signals and the two low order address bits.

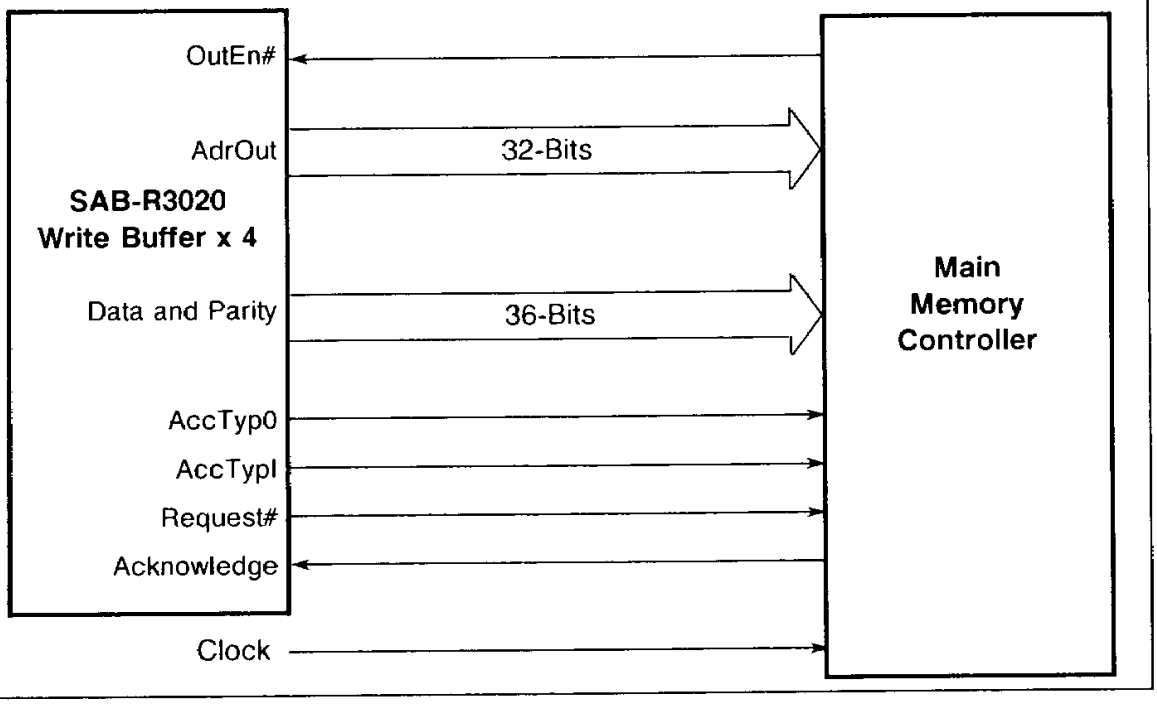
Figure 5
Write Buffer Data and Address Line Connections



Write Buffer-Main Memory Interface

Figure 6 shows the signals comprising the write buffer interface to main memory. The interface is essentially decoupled from the write buffer-processor interface, although some synchronization of the memory interface signals and the clock signal is required. The handshaking signals with main memory have no direct connection with the write buffer-processor interface.

Figure 6
Write Buffer/Main Memory Interface



Byte Gathering

The write buffers perform byte (half-word, tri-byte, and word) gathering to decrease the number of write transfers to the same word location. Sequential writes to the same word address have their data combined into the same address-data pair buffer.

Byte gathering is prohibited in the address-data pair that is currently available to the memory controller. Thus the first write into an empty write buffer will not have subsequent writes gathered into it. Writes to the same location (byte) will be overwritten in the write buffer if gathering is not prohibited by the preceding rule.

The write buffers present address-data pairs to the main memory in the sequence in which they were received from the processor except in the case of gathered data, where bytes or half-words can be collected and written to main memory in a single write operation.

If the address-data pair is scheduled to be output, then gathering is inhibited and the buffer contents are presented to the main memory controller. Subsequent writes are then placed in other buffers. No reliance should be placed on byte gathering as it is not readily deterministic. Non-sequential writes to the same address are not gathered.

In some cases gathering may require that two memory controller references be used to empty a single write buffer entry. For example, this may occur if bytes 0 and 3 of a word are sequentially written. Where order in writing is important, such as with I/O controllers, software should avoid sequential access to the same word. In cases where write-read access ordering is important but reading of the write locations is not desired (such as I/O), a write followed by a write to a dummy location will ensure that the first write has occurred before continuing. Alternatively, the REQUEST# signal can be tested to determine that the write buffer is empty.

Configuration Logic

Because of their byte gathering ability, each write buffer internally maintains a record of each valid byte in an address/data pair. To do this, each device must have a way of determining which data bits within a word it is handling. AccType0/1, Address0/1, BigEndian, and Position0/1 determine how the write buffers handle data when it is received. The table below shows the position of bytes within a word based on these signals.

Table 1
Position of Bytes within words

Access Type		Address		Bytes Accessed							
				31 Big-Endian 0				31 Little-Endian 0			
1	1										
(word)		0	0	0	1	2	3	3	2	1	0
1	0	0	0	0	1	2			2	1	0
(triple-byte)		0	1		1	2	3	3	2	1	
0	1	0	0	0	1					1	0
(halfword)		1	0			2	3	3	2		
0	0	0	0	0							0
(byte)		0	1		1					1	
		1	0			2			2		
		1	1				3	3			

Note also that the lower two address bits of the device in position zero (as determined by the two Position inputs) are inhibited. Instead on output, the lower two address bits are generated from the positions of the valid data bytes as determined by the above table.

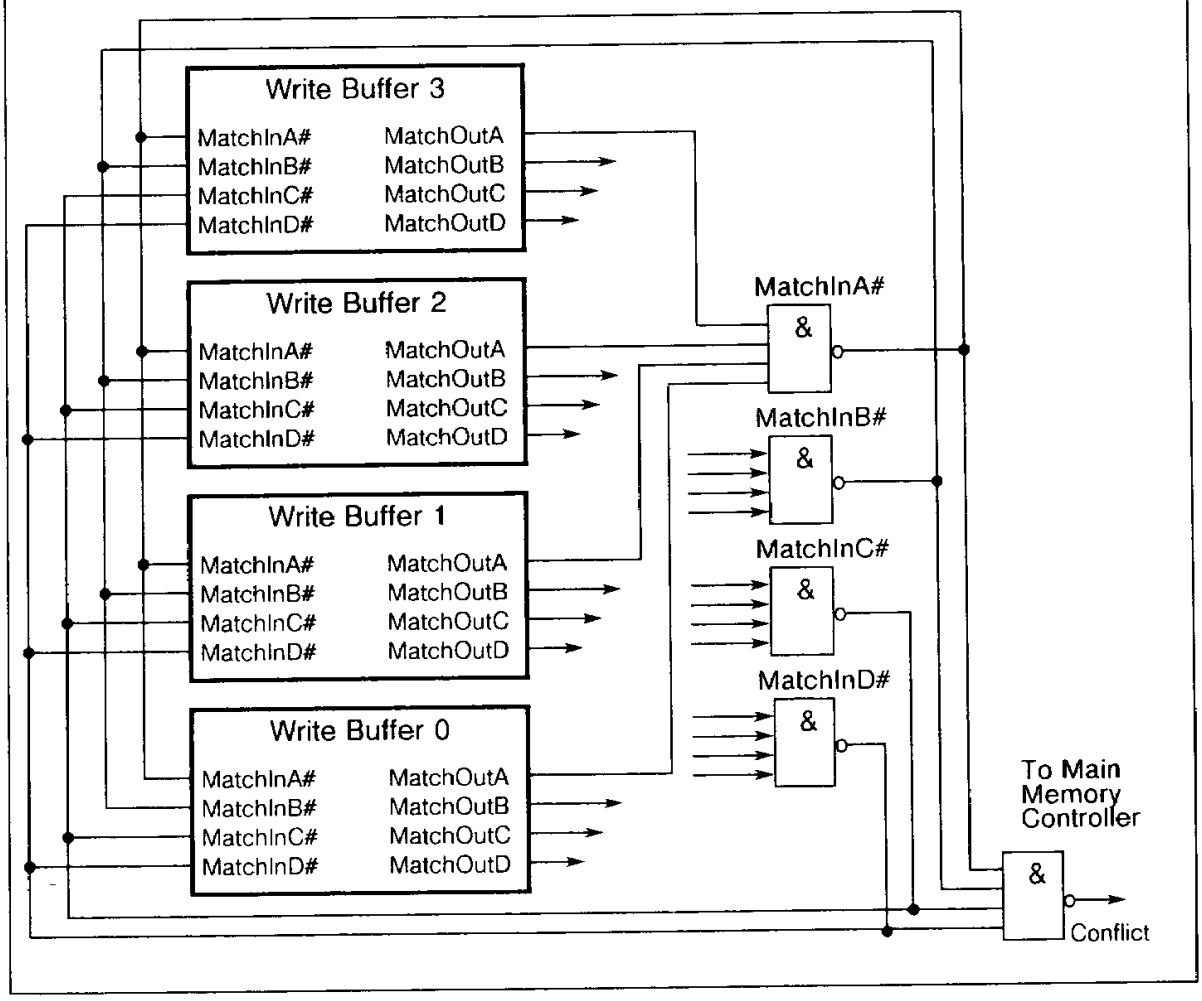
Conflict Resolution

When the SAB-R3000 (SAB-R2000A) references main memory (either a write or a read reference), the write buffers compare the word address from the CPU with the word address stored in the buffers. If any word address matches, the write buffers assert signals that can be used by the main memory controller to ensure that the write buffer is emptied before the read access with the conflicting address has been performed.

Figure 7 illustrates the write buffer signals involved in the address comparison logic. Each write buffer provides four output signals (MatchOutA, B, C, D) which correspond to the four buffer ranks in each device. These MatchOut signals can be externally NANDed to determine if the address being input matches those in any rank of the write buffer.

The outputs of the NAND gates are fed back into the write buffers via MatchA#, B#, C# and D# and are used within each device as part of the byte gathering logic. The NAND gate outputs can be NANDed together as shown with the resultant signal (in conjunction with the CPU's MEMRD signal) to alert the main memory controller logic that there is a pending buffered write that conflicts with a just issued read. The main memory controller can then delay the read access until the Request is deasserted indicating that the write buffer has been emptied.

Figure 7
Write Buffer MatchIn/MatchOut Logic

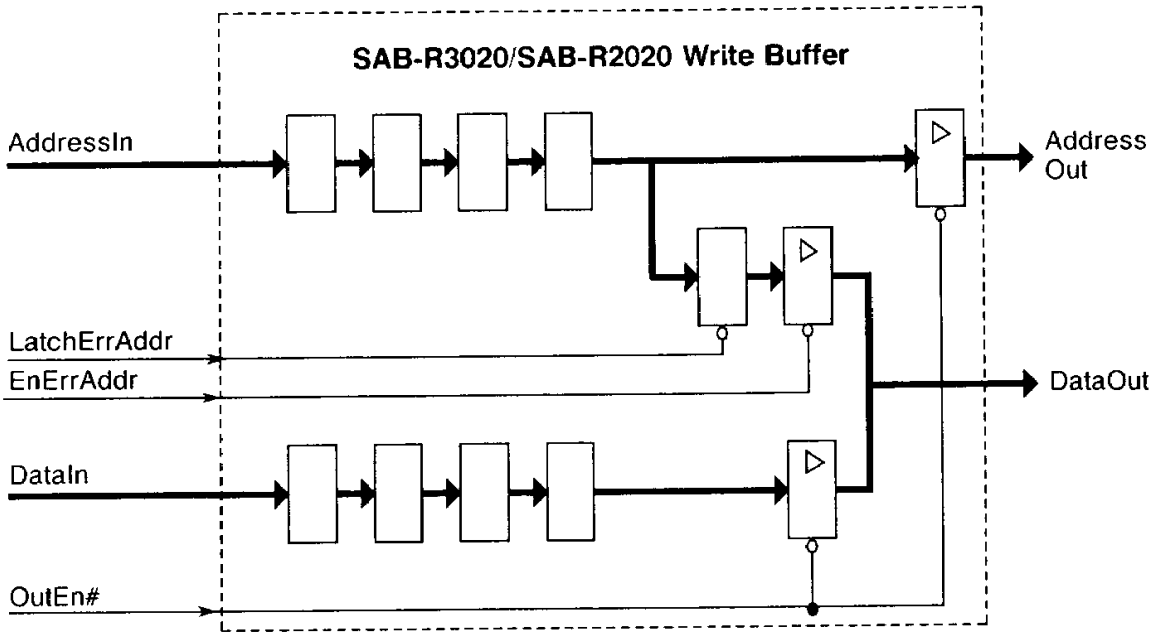


Error Address Latch

The write buffer incorporates an internal latch that can be loaded with one of the buffered addresses and subsequently enabled out onto the data lines. This feature can be used by error handling to read an address back from the write buffer and analyze, or recover from, certain bus errors. Figure 8 shows the signals involved in the operation of this address latch.

When the LatchErrAdr signal is asserted, the address currently available to the address outputs of the write buffer is latched into the internal latch. This address can then be output on the DataOut lines by asserting the EnErrAdr signal so that the processor can read the address in as data.

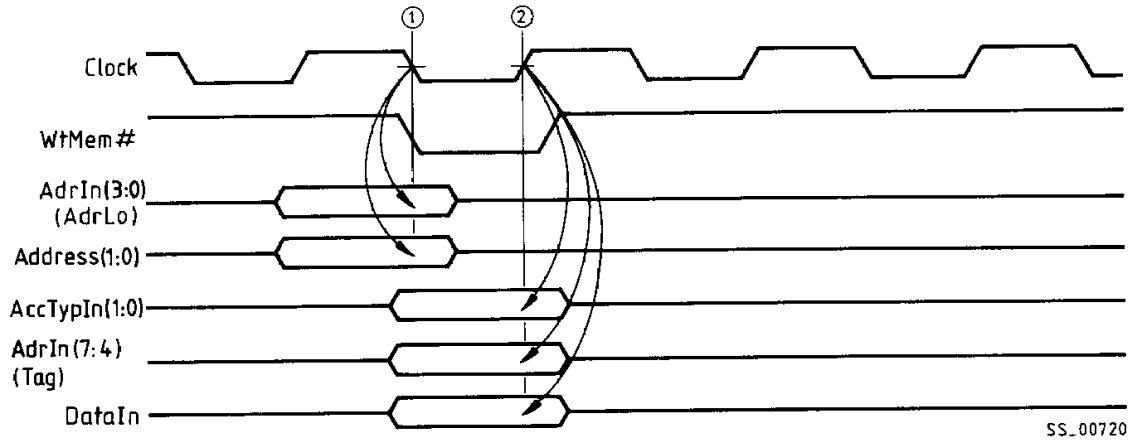
Figure 8
Write Buffer Error Address Latch



AC Timing

Transfers between the processor and the write buffers occur synchronously: the clock signal from the processor is input to the write buffers and used to clock the address and data information into the write buffer's latches. The relative timing is shown in figure 9.

Figure 9
AC Timing Diagram



When the WtMem# signal is asserted, the low-order address bits, the Address (1:0) inputs, and the access type inputs are latched on the trailing edge of the clock signal①. The rising edge of Clock② is used to latch the high-order address bits and the contents of the data bus.

Figure 10 illustrates the timing for transfer of data from the write buffer to the main memory system. The sequence is as follows:

① When the write buffer has a data-address pair for transfer to the memory system, it asserts the Request# signal.

② When the memory system is ready to handle the write buffer data, it asserts the OutEn# signal to enable the write buffers' address and data outputs onto the system bus.

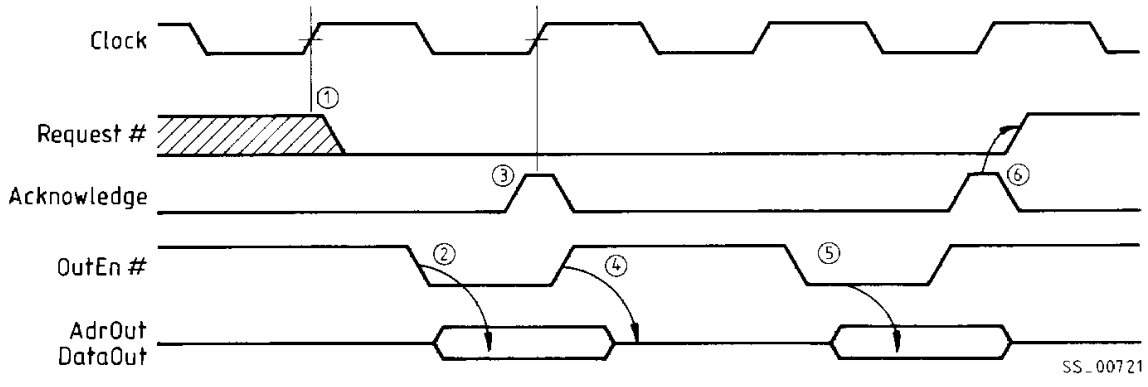
③ When the memory system no longer requires the write buffer address and data outputs, it asserts the Acknowledge signal. The write buffer responds to this signal by discarding the address-data pair that was just output.

④ The memory system can deassert the OutEn# signal to return the write buffers' address and data outputs to their tri-state condition.

⑤ Since the Request# signal remains asserted, the memory system asserts the OutEn# signal again to enable the next address-data pair onto the system buses.

⑥ When the memory system has accepted the second address-pair it again asserts the Acknowledge signal. If the write buffer is now empty, it responds to this signal by deasserting the Request# signal.

Figure 10
Write Buffer/Main Memory Interface Timing



Note that the write buffer's interface to main memory is not completely asynchronous; assertion of the Request# signal by the write buffer is synchronized with the rising edge of Clock, and the Acknowledge signal input by main memory has a minimum set up and hold time in relation to the Clock signal.

Absolute Maximum Ratings

Ambient temperature under bias (T_A)	0 to +70 °C
Storage temperature (T_{ST})	– 40 to +125 °C
Lead temperature (T_L)	300 °C
Supply voltage (V_{CC})	– 0.5 to +7.0 V
Input voltage (V_{IN})	– 0.5 to +7.0 V

Note: *Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.*
Not more than one output should be shorted at a time. Duration of the short should not exceed 30 seconds.

DC Characteristics

T_A = 0 to +70 °C; V_{CC} = 5 V ± 5 %

Parameter	Symbol	Limit values				Unit	Test condition
		SAB-R2020 16.67 MHz		SAB-R3020 25 MHz			
		min.	max.	min.	max.		
Operating Parameters							
Output HIGH voltage	V_{OH}	2.4	–	2.4	–	V	V_{CC} = min. I_{OH} = – 4 mA
Output LOW voltage	V_{OL}	–	0.4	–	0.4	V	V_{CC} = min. I_{OL} = 4 mA
Input HIGH voltage	V_{IH}	2	V_{CC} +0.25	2	V_{CC} +0.25	V	
Input LOW voltage	V_{IL}	– 0.5[1]	0.8	– 0.5[1]	0.8	V	
Input Leakage	I_{IN}	– 80	+80	– 80	+80	µA	V_{IN} = V_{DD} or GND
Output Leakage	I_{OZ}	– 40	+40	– 40	+40	µA	V_{OUT} = V_{DD} or GND
Operating current	I_{CC}	–	70	–	70	mA	V_{CC} = 5.25 V
Load capacitance	C_{Ld} [2]	–	50	–	50	pF	

1) V_{IL} min. = – 3.0 V for pulse width less than 15 ns

2) Operation above the C_{Ld} maximum may impair the useful life of the device.

AC Characteristics

$T_A = 0$ to $+70$ °C; $V_{CC} = 5$ V ± 5 %

Parameter	Symbol	Limit values				Unit
		SAB-R2020 16.67 MHz		SAB-R3020 25 MHz		
		min.	max.	min.	max.	
AdrIn (3:0) to clock falling setup	t_{01}	2	–	6	–	ns
AdrIn (3:0) from clock falling hold	t_{02}	5	–	4	–	ns
Address (1:0) to clock falling setup	t_{03}	5	–	6	–	ns
Address (1:0) from clock falling hold	t_{04}	4	–	6	–	ns
AccessType (1:0) to clock rising setup	t_{05}	4	–	6	–	ns
AccessType (1:0) from clock rising hold	t_{06}	3	–	4	–	ns
AdrIn (7:4) to clock rising setup	t_{07}	5	–	4	–	ns
AdrIn (7:4) from clock rising hold	t_{08}	3	–	3	–	ns
DataIn (8:0) to clock rising setup	t_{09}	4	–	4	–	ns
DataIn (8:0) from clock rising hold	t_{10}	3	–	3	–	ns
WtMem# to clock rising setup	t_{11}	10	–	7	–	ns
WtMem# from clock rising hold	t_{12}	5	–	4	–	ns
Request from clock rising	t_{13}	–	25	–	22	ns
Acknowledge to clock rising setup	t_{14}	10	–	6	–	ns
Acknowledge from clock rising hold	t_{15}	6	–	6	–	ns
LatchErrAdr to clock rising setup	t_{16}	2	–	5	–	ns
WbFull# active from clock rising	t_{17}	–	21	–	17.3	ns
WbFull# inactive from clock rising	t_{18}	–	21	–	11	ns
OutEn to AdrOut (7:0), DataOut(8:0) valid	t_{19}	–	16	–	16	ns
OutEn to AdrOut (7:0), DataOut(8:0) Tri-State	t_{20}	–	15	–	15	ns
MatchOut (A:D) from clock rising	t_{21}	–	24	–	22	ns
MatchIn (A#:D#) to clock rising setup	t_{22}	10	–	10	–	ns
MatchIn (A#:D#) from clock rising hold	t_{23}	0	–	3	–	ns
EnErrAdr to data (error latch) valid	t_{24}	–	16	–	23	ns

AC Characteristics (cont'd)

Parameter	Symbol	Limit values				Unit
		SAB-R2020 16.67 MHz		SAB-R3020 25 MHz		
		min.	max.	min.	max.	
EnErrAdr to data (error latch) Tri-state	t_{25}	–	11	–	15	ns
Address/DataOut from clock rising	t_{26}	–	30	–	34	ns
Reset# to clock rising setup	t_{27}	10	–	10	–	ns
Reset# from clock rising hold	t_{28}	1	–	1	–	ns
Reset# low pulse width	t_{29}	8	–	8	–	ns
WbFull# high from Clk Rsg (after Reset#)	t_{30}	–	14	–	20	ns
Request# high from Reset# Low	t_{31}	–	17	–	18	ns
AccTypIn (1:0) low from Reset# Low	t_{32}	–	21	–	25	ns
MatchOut (A:D) low from Reset# Low	t_{33}	–	19	–	20	ns
Address/DataOut from Reset# Low	t_{34}	–	32	–	27	ns
AccTypeOut (0:1) from clock rising	t_{35}	–	32	–	27	ns

Figure 11
Write Buffer Timing Specifications

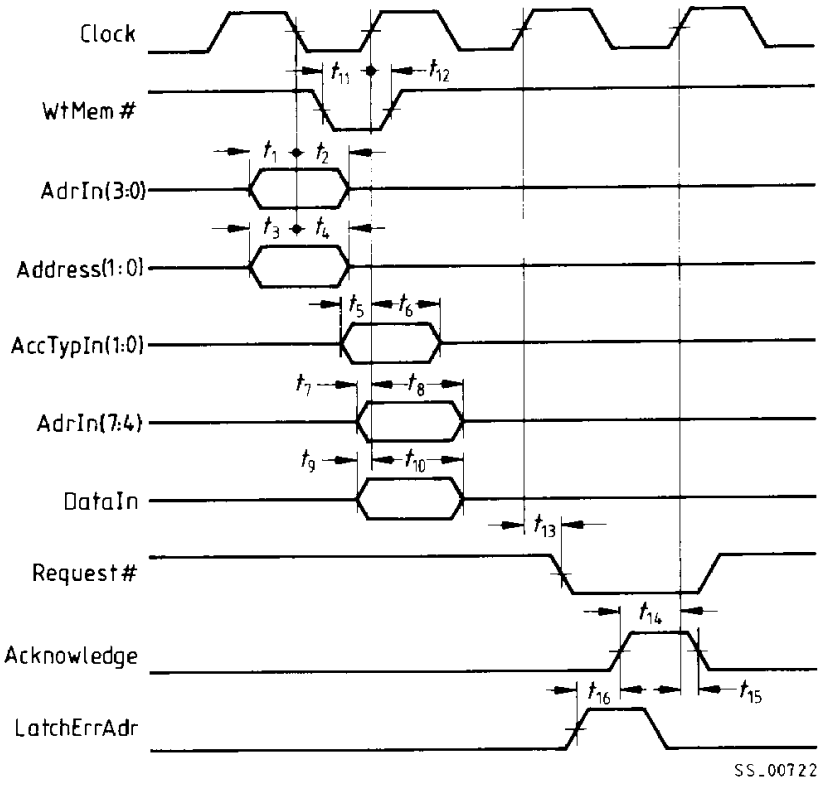


Figure 12
WbFull# Timing Specification

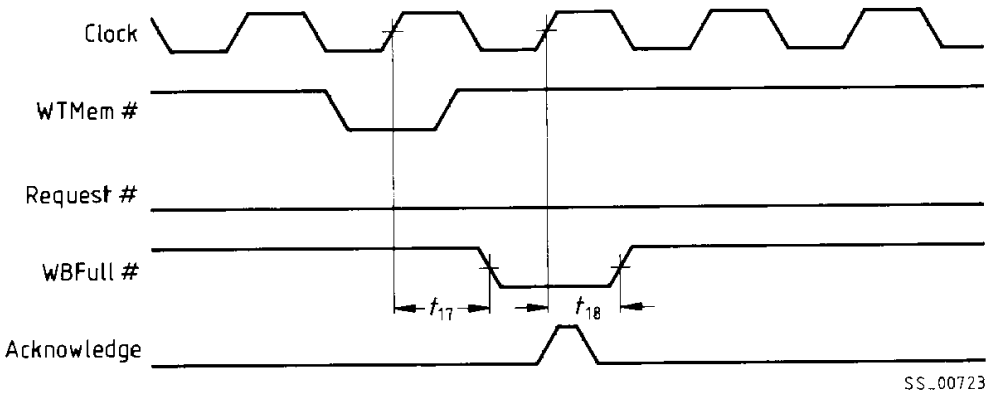


Figure 13
OutEn# Timing Specification

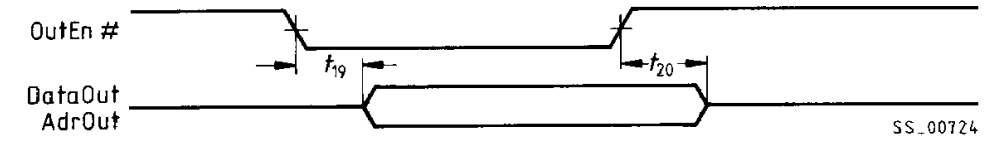


Figure 14
Match and Error Latch Timing Specifications

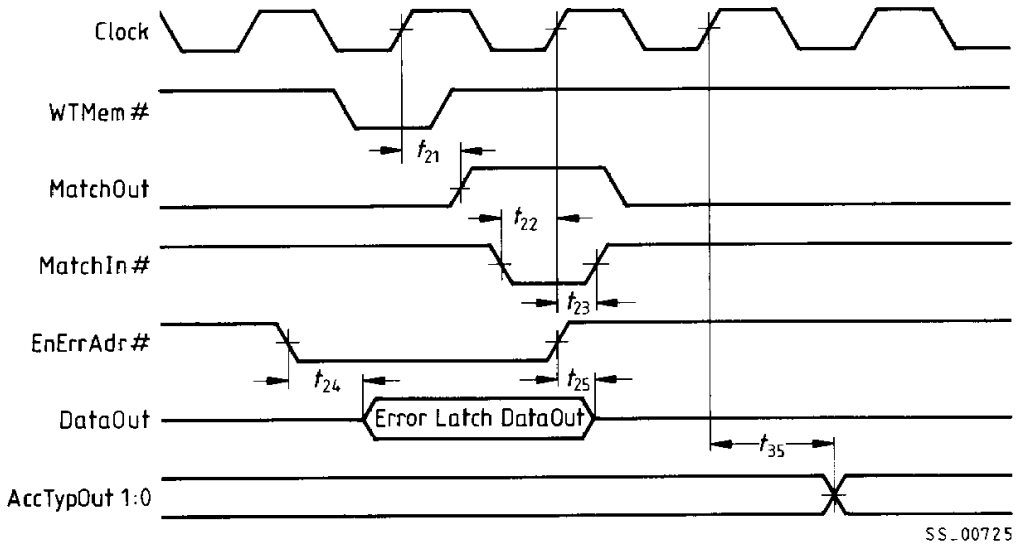


Figure 15
Address/DataOut Timing Specification

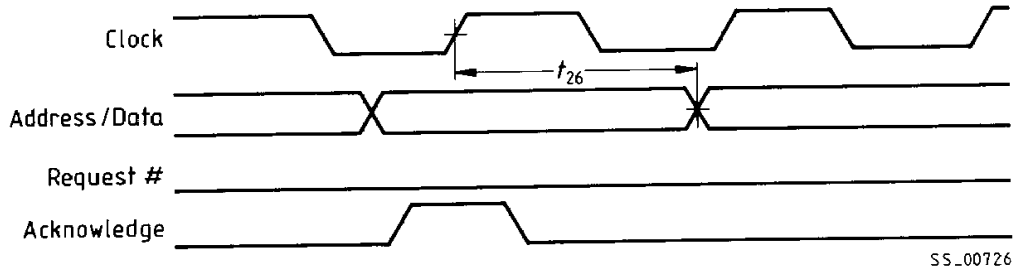


Figure 16
Timing Specifications