

## UT1553B BCRT True Dual-port Memory Interface

### INTRODUCTION

The UTMC UT1553B BCRT is a monolithic CMOS integrated circuit that provides comprehensive MIL-STD-1553B Bus Controller and Remote Terminal functions. The BCRT design reduces the overhead placed on the host computer by automatically executing message transfers, providing interrupts, and generating status information. The BCRT off-loads the host processor with built-in memory management functions designed specifically for MIL-STD-1553B applications. This means that the host need only establish the necessary data and/or control parameters in memory so the BCRT can access the information as required and, therefore, provide the requisite MIL-STD-1553B bus functions. UTMC variants of the BCRT are the BCRTM, a BCRT with monitor functions, and the BCRTMP, a BCRT which operates in a wide variety of 1553 protocols.

This note will discuss a true dual-port (TDP) interface configuration. The design uses a 2K x 16 dual-port memory device available from several manufacturers. If additional memory is needed, the designer can incorporate more than one memory chip. Some manufacturers also offer single package dual-port RAM assemblies in configurations up to 8K x 16. This interface is applicable to the BCRT, the BCRTM, and the BCRTMP.

### TRUE DUAL-PORT CONFIGURATION

The TDP configuration's main advantage over both the DMA (direct memory access) and PDP (pseudo dual-port) configurations is that it generates minimal impact on the host processor's operations. In the DMA configuration, the processor must be put on "hold" during any BCRT memory access. Using the PDP method, the processor must wait for the BCRT whenever the BCRT is accessing common memory. The only time the processor must wait in the TDP configuration is when the processor tries to access the exact same memory location to which the BCRT already has access, or when the host processor needs to access one of the BCRT's internal registers and the BCRT is in a memory cycle.

The TDP configuration's main disadvantage is cost. The price of dual-port devices is significantly higher than that of standard memory, and the density of the memory devices is lower. Of course, in a system where throughput is the driving issue, the TDP method is the most effective solution.

### CIRCUIT OPERATION - MEMORY ARBITRATION

Figure 1 shows the circuit's basic configuration. Figure 2 details operation of the PLD. The PLD is designed as a Mealy machine to improve response time and simplify state transitions. The dual-port RAM contains two signals:  $\overline{\text{BUSYL}}$  and  $\overline{\text{BUSYR}}$ . When both the host and BCRT access the same memory locations at the same time, one of these signals will assert. Generally, the rules outlined in table 1 are followed.

**Table 1. Simultaneous Memory Arbitration**

CONDITION	"WINNING" SIDE	NOTES
left Xns before right	left	1
right Xns before left	right	1
left/right within Xns of each other	arbitration	1, 2

**Notes:**

1. X = detection limit, this value is between 2 and 10 ns depending upon the device used.
2. Arbitration algorithm depends on device. Examples: always give to left, give to side which didn't have it last, random, etc.

The TDP configuration outlined in this note uses  $\overline{\text{BUSYL}}$ ,  $\overline{\text{BUSYR}}$ ,  $\overline{\text{DMAR}}$ ,  $\overline{\text{DMAG}}$ ,  $\overline{\text{MEMCSI}}$ ,  $\overline{\text{BCRTCS}}$ , and  $\overline{\text{READY}}$  to properly arbitrate RAM use.

When the BCRT needs to read or write to dual-port memory, it asserts  $\overline{\text{DMAR}}$ . If (or when) the host access to memory is inactive (i.e.,  $\overline{\text{MEMCSI}}$  is deasserted),  $\overline{\text{DMAG}}$  will be asserted and the BCRT will begin its memory cycle. Note that in this case, the memory location being accessed does not affect the arbitration decision. This would seem to defeat the purpose of the dual-port. However, the BCRT-side arbitration must work this way for the following combination of reasons:

1. In order to determine a "busy" condition in the dual-port RAM, all addresses must be known.
2. Once the BCRT receives  $\overline{\text{DMAG}}$ , there is no way to delay completion of its memory cycle.

Thus, if we try to give the BCRT a grant and use  $\overline{\text{BUSYR}}$  for a collision, there is no way to use  $\overline{\text{BUSYR}}$  to delay the BCRT when an address match occurs. This is really not a problem; since the BCRT's "throughput" is not very high, it can afford to wait for a host memory cycle to complete. Figure 3 shows BCRT-memory access timing.

Note: Most host access times are small (1/10) compared to the BCRT request-to-grant requirement of 1.9 $\mu\text{s}$  at 12MHz. However, when designing a system, this requirement should not be overlooked.

When the host accesses memory, it is only delayed if a collision occurs, thus realizing the advantage of the dual-port RAM. If and only if an address match occurs,  $\overline{\text{BUSYL}}$  will assert and the host will have to wait for the BCRT to complete its cycle. If the host accesses a memory location other than the one the BCRT is using, both the BCRT and host will complete their memory cycles normally. Figure 5 shows the timing diagrams for host-memory accesses both with and without arbitration.

A special arbitration case may also occur in this design. After the BCRT receives a  $\overline{\text{DMAG}}$ , the host may access memory. The BCRT may take up to four MCLK periods before it actually begins the memory access, thus it is possible the host may begin a cycle after  $\overline{\text{DMAG}}$  has asserted but before  $\overline{\text{MEMCSO}}$  on the BCRT asserts. Since the BCRT has already started a cycle, it is too late to stop this access. If the host and BCRT address the same memory location in this situation,  $\overline{\text{BUSYR}}$  will assert. As was already stated, there is no way to stop the BCRT in this situation. To correct this problem, the host will be delayed if  $\overline{\text{BUSYR}}$  asserts. The  $\overline{\text{READY}}$  signal will deassert and the chip select to the host side of memory will be blocked. This clears the  $\overline{\text{BUSYR}}$  condition. The BCRT will complete its cycle. After completion, the host chip select will be reasserted and the host cycle will complete.

The PLD device controls all of the arbitration. It consists of a state machine to control access to the "BCRT bus" and some random logic to generate the  $\overline{\text{READY}}$  signal.

### Special Cautions

Since this document is written for a "generic" processor, some cautions are necessary for direct use of this design:

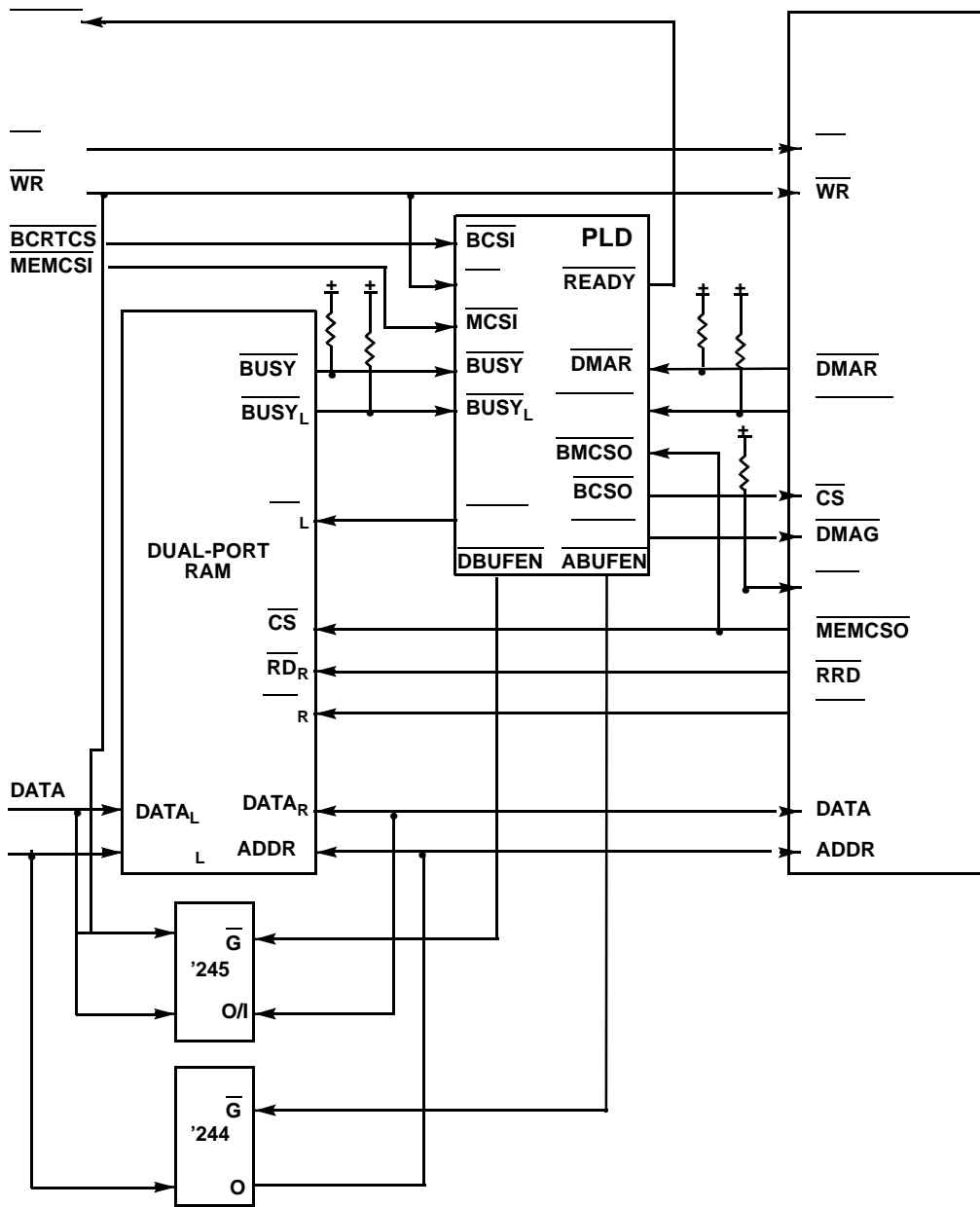
1. To prevent multiple BCRT read/write operations, the  $\overline{\text{BCSI}}$  (BCRT chip select in) signal must be asserted low for no more than one cycle after the falling edge of  $\overline{\text{READY}}$ .
2. The  $\overline{\text{READY}}$  signal stays in an asserted state to facilitate faster memory access times. Thus during a host-to-BCRT cycle, the  $\overline{\text{READY}}$  signal must deassert high first. This deassertion may take as long as two clock cycles.
3. During a host-to-BCRT register read, the data from the BCRT is only valid for one clock cycle after the  $\overline{\text{READY}}$  signal is asserted low.
4. During a host-to-memory cycle, the  $\overline{\text{READY}}$  signal is asynchronous; it may change to a deasserted state at any time during the cycle. As long as the host cycle does not complete while  $\overline{\text{READY}}$  is deasserted, no problems will occur.

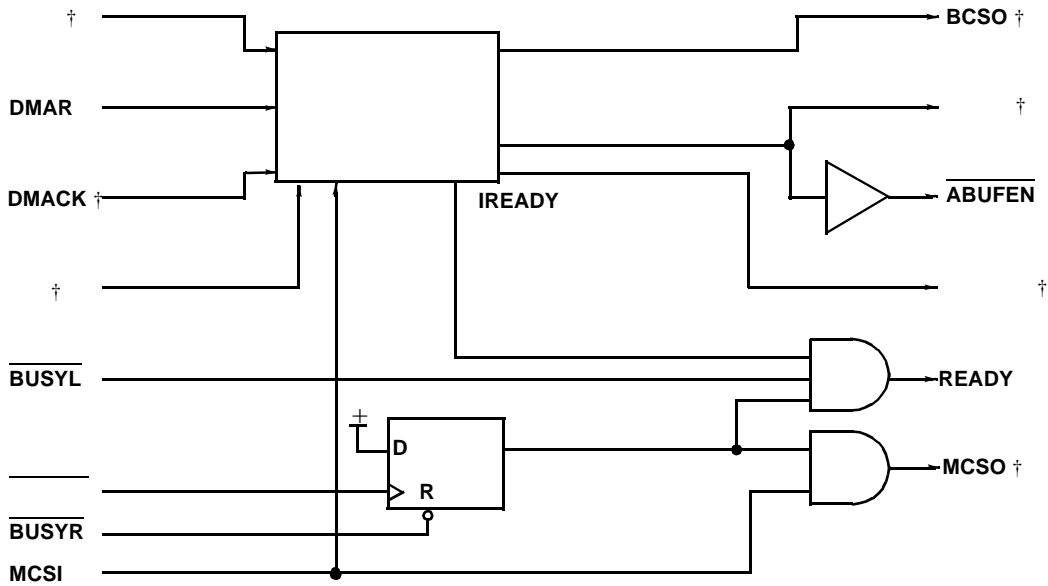
### REGISTER ARBITRATION

The same Address/Data bus is used for both BCRT register access by the host and memory access by the BCRT. Thus, when the host accesses a BCRT internal register, it must wait for the BCRT to complete a current memory cycle. This is accomplished using the host's  $\overline{\text{READY}}$  signal and the BCRT's  $\overline{\text{DMACK}}$  signal. When the BCRT is given a grant,  $\overline{\text{DMACK}}$  is asserted until the BCRT is finished. The assertion of  $\overline{\text{DMACK}}$  will prevent the address/data buffers and the  $\overline{\text{READY}}$  line from being enabled. When the BCRT completes, the buffers are enabled. One clock cycle later, the  $\overline{\text{READY}}$  signal will assert. Figure 4 shows the timing diagrams for host-BCRT accesses with and without arbitration.

### FURTHER ASSISTANCE

Due to the variety of processors to which the BCRT can be connected, this document cannot address all the possibilities. Contact UTMC applications support for additional assistance.





To avoid confusion, the PLD design is implemented in positive (active high) logic. It will be necessary to add inverters to the design where

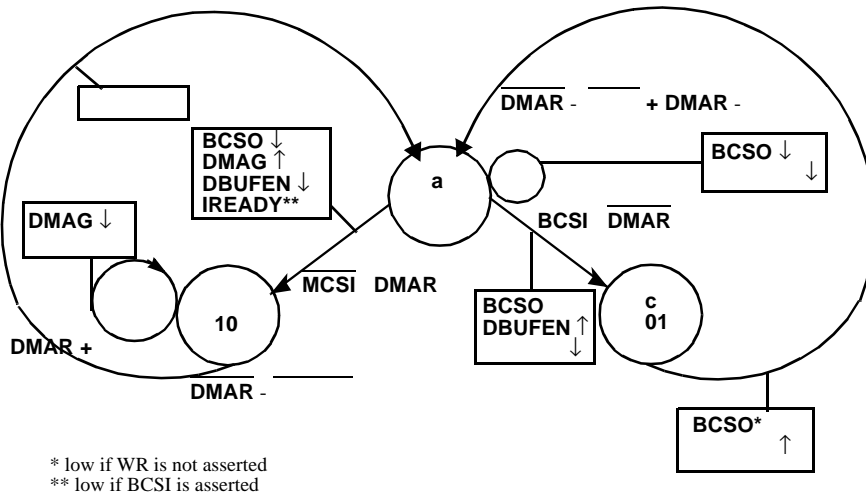


Figure 2. PLD Description

INPUT VARIABLES					STATE				NEXT		OUTPUT VARIABLES				
WR	MCSI	DMAR	BCSI	DMACK	S		NS <sub>0</sub>				BCSO	DMAG	DBUFEN	IREADY	
					0	1			0	1					
X	X	L	L	X	0	0	a	→	a	0	0	L	L	L	H
X	H	H	X	X	0	0	a	→	a	0	0	L	L	L	H
X	L	H	L	X	0	0	a	→	b	1	0	L	H	L	H
X	L	H	H	X	0	0	a	→	b	1	0	L	H	L	L
X	X	L	H	X	0	0	a	→	c	0	1	H	L	H	L
X	X	H	L	X	1	0	b	→	b	1	0	L	L	L	H
X	X	H	H	X	1	0	b	→	b	1	0	L	L	L	L
X	X	X	L	H	1	0	b	→	b	1	0	L	L	L	H
X	X	X	H	H	1	0	b	→	b	1	0	L	L	L	L
X	X	L	L	L	1	0	b	→	a	0	0	L	L	L	H
X	X	L	H	L	1	0	b	→	a	0	0	L	L	L	L
H	X	X	X	X	0	1	c	→	a	0	0	L	L	H	H
L	X	X	X	X	0	1	c	→	a	0	0	H	L	H	H

PLD state machine equations:

$$\dagger BCSO := \overline{DMAR} - BCSI - \overline{S0} - \overline{S1} + \overline{S0} - S1 - \overline{WR}$$

$$\dagger DMAG := \overline{MCSI} - DMAR - \overline{S0} - \overline{S1}$$

$$\dagger DBUFEN := \overline{DMAR} \quad \overline{S0} - \overline{S1} + \overline{S0}$$

$$IREADY := \overline{DMAR} \quad \overline{S0} - \overline{S1} + DMAR - \overline{MCSI} \quad \overline{S0} - \overline{S1} - BCSI + S0 - \overline{S1}$$

$$NS0 := DMAR @ S0 @ \overline{S1} \quad \overline{S1} + DMAR @ \overline{S0} @ \overline{S1}$$

$$NS1 := \overline{S1} @ \overline{S1} \quad \overline{DMAR} @ BCSI$$

x = don't care

† = outputs which will probably need inversion

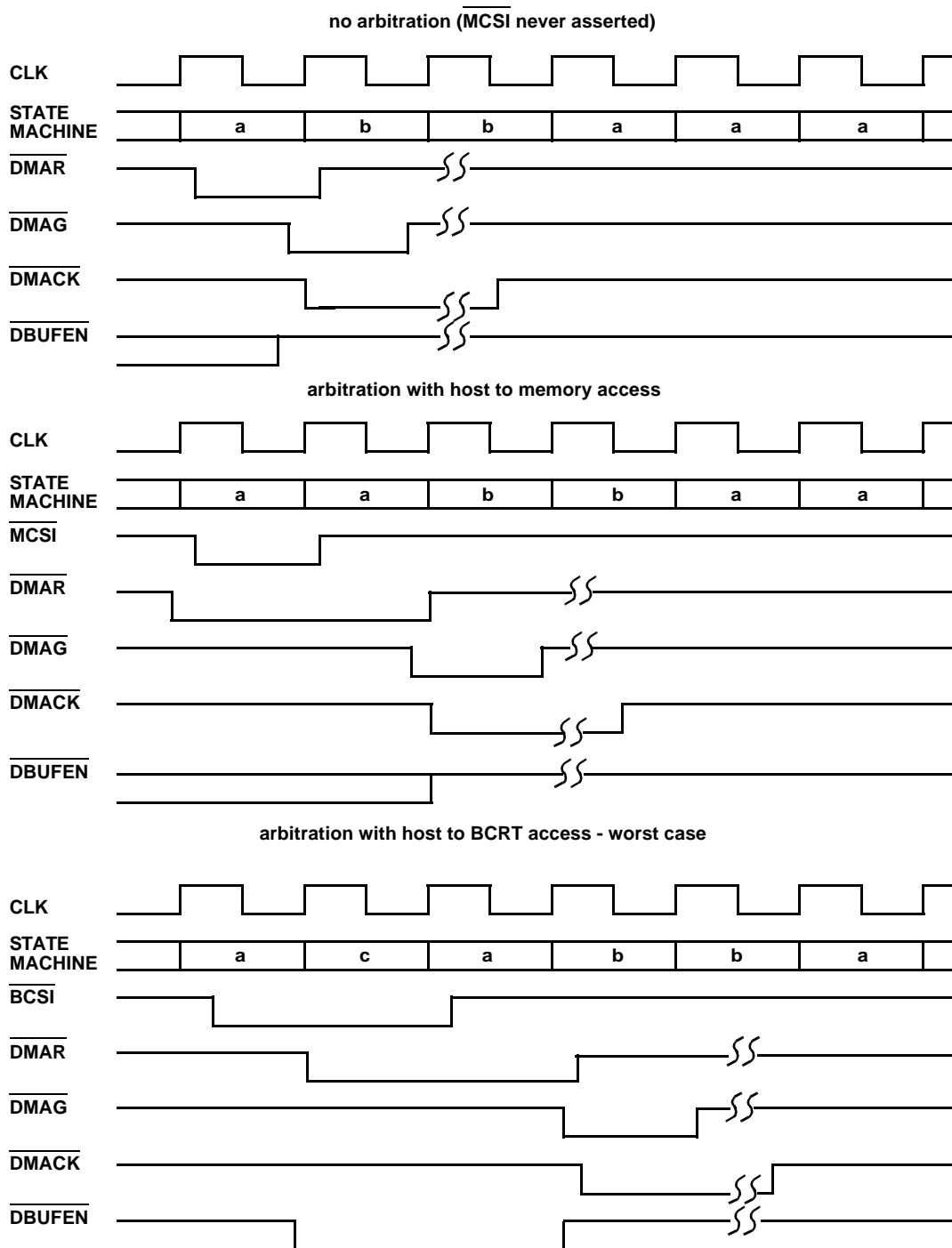


Figure 3. BCRT-Memory Access (Single Word)

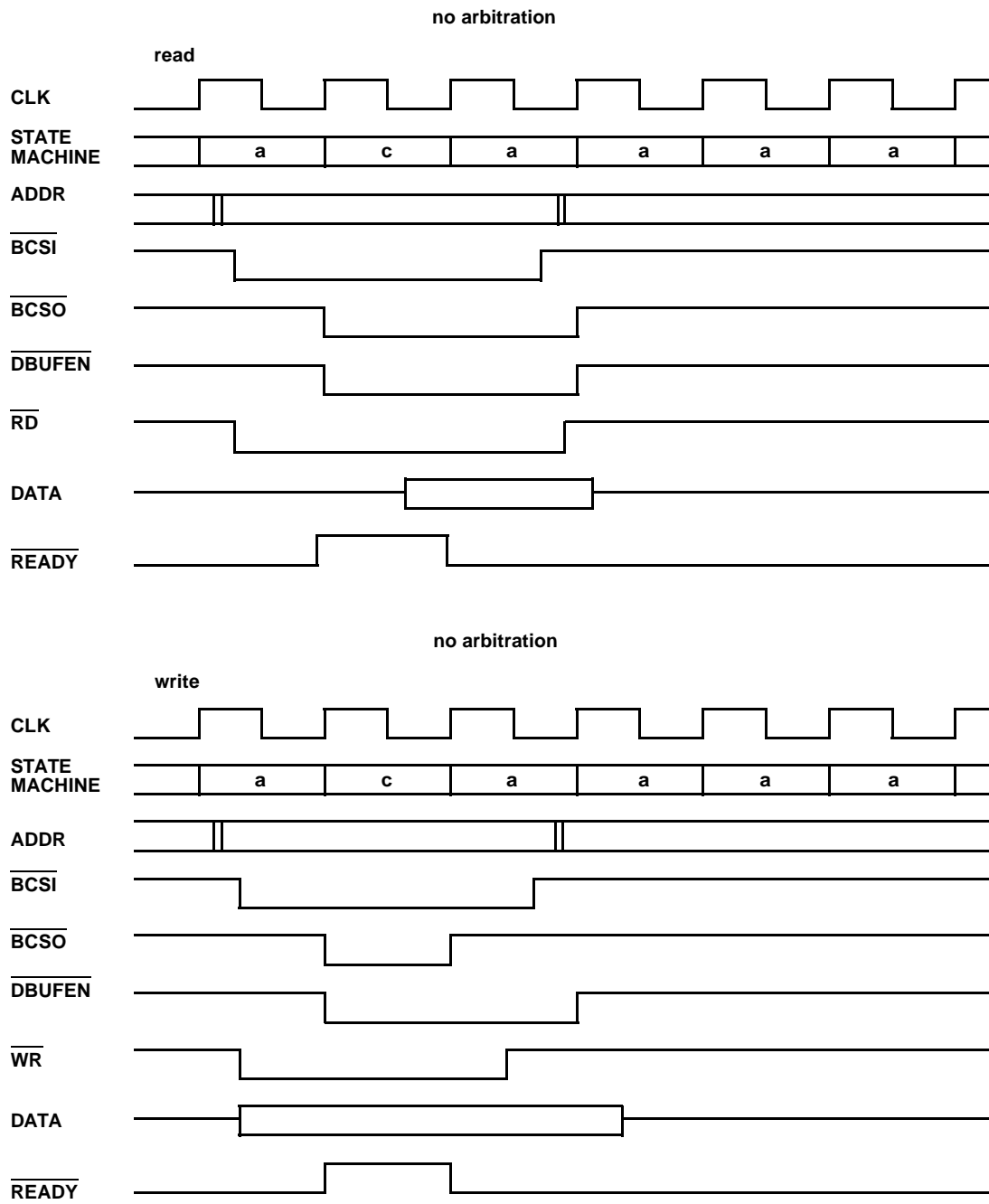
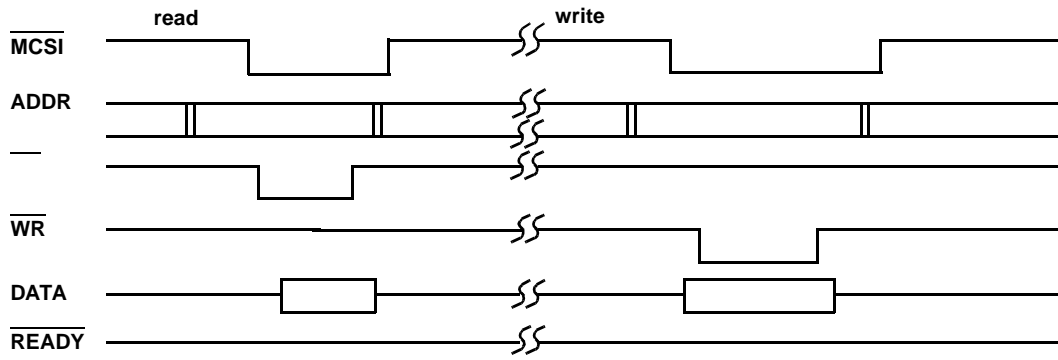


Figure 4. Host-BCRT Access



arbitration

