# 4.3

# Interfacing to Cypress VMEbus Interface Controllers

Previously, interfacing the VIC068A to the VMEbus required a significant number of LSI and MSI devices. With the advent of 64-bit VMEbus block transfers and the VIC64, the external discrete device count for a full functional interface has expanded. The CY7C964 has been developed to combat this problem by incorporating the functions of much of this external logic into a single package. Using the CY7C964 shortens system design, debug, and manufacturing cycle times. Design engineers are relieved from the burden of performing critical or worst-case timing analysis on the VMEbus and VIC buffer control signals. Local control signals other than those directly interfaceable to the interface controllers have been kept to a minimum.

A full function D64 VMEbus interface implemented using the CY7C964, VIC64, and all VMEbus interface local support logic includes the following features:

- block transfer support for D16, D32, and D64 VMEbus transfers
- dual-path address operation (allowing single-cycle transfers during master block transfer interleave periods)
- slave block transfers during master block transfer interleave periods
- fully programmable slave VMEbus address decoding
- write posting
- the VIC mail box interrupt message support

The interface can be broken into five functional sections for the purpose of discussion. These sections are:

- VMEbus signal group
- Buffer control signal group
- CY7C964 local signal group
- CY7C964 address comparator and local signal group
- Local data swap buffer logic

The focus of this section is the CY7C964. Each of the interface functional blocks are examined from this perspective. For additional information on the signals described within

this section, consult the VMEbus Specification (IEEE 1014) and/or the section of this book that describes the controller you are connecting.

## 4.3.1 VMEbus Signal Group

This group includes the VMEbus address and data signals. Each CY7C964 provides support for 8 bits of VMEbus address and data. Three CY7C964s are necessary for 32-bit interface applications when used with the VIC068A or VIC64. Four CY7C964s are used with CY7C960 and CY7C961 for full-function slave interfaces. The A[7:0] and D[7:0] transceivers on the CY7C964 furnish a high drive strength, allowing direct connection to the respective address and data signals on the VMEbus backplane. With any of Cypress's family of VMEbus interface controllers generating the control information, the CY7C964s meet VMEbus worst-case timing and drive-strength requirements for all forms of data transfers.

**Table 4−1. VMEbus Signals**

| Signal | Description | Interface Comments |
|---|---|---|
| D[7:0] | VMEbus compatible data signals | Directly connect to VMEbus P1 and P2 connectors |
| A[7:0] | VMEbus compatible address signals | Directly connect to VMEbus P1 and P2 connectors |

## 4.3.2 Buffer Control Signal Group

This group includes all of the address and data buffer control signals. A major design time savings is realized using the CY7C964s as all of these signals directly connect to the chosen controller or are wired to a steady state value. The buffer control interface is simple and straightforward with a few minor exceptions. *Table 4−2* specifies these connections for the VIC068A and VIC64 controllers.

**Table 4−2. Buffer Control Signals Connection for VIC068A and VIC64**

| Signal | Description | Interface Comments |
|---|---|---|
| LADO | Latch address out | Directly connect to VIC LADO on all CY7C964s |
| LADI | Latch address in | Directly connect to VIC LADI on all CY7C964s |
| LEDO | Latch enable data out | Directly connect to VIC LEDO on all CY7C964s |

**Table 4−2. Buffer Control Signals Connection for VIC068A and VIC64** (continued)

| Signal | Description | Interface Comments |
|---|---|---|
| LEDI | Latch enable data in | Directly connect to VIC LEDI on all CY7C964s |
| ABEN* | VMEbus address bus enable | Directly connect to VIC ABEN* on all CY7C964s |
| DENO* | Data enable output | Directly connect to VIC DENO* on all CY7C964s |
| D64 | D64 block transfer mode enable | Directly connect to VIC64 SCON/D64 pin on all CY7C964s. Tie this input LOW on all CY7C964s when using VIC068A. |
| BLT* | Block transfer enable | Directly connect to VIC BLT* on all CY7C964s |
| LAEN | Local address enable | Directly connect to VIC LAEN on all CY7C964s |
| DENIN* | Primary data enable in signal | Directly connect to VIC UWDENIN* on NMSB and MSB CY7C964s, directly connect to VIC LWDENIN* on LSB CY7C964 |
| DENIN1* | Companion data enable in signal | Directly connect to VIC LWDENIN* on NMSB and MSB CY7C964s, directly connect to VIC UWDENIN* on LSB CY7C964 |

## 4.3.3 CY7C964 Local Signal Group

The CY7C964 local signal group consists of the VMEbus and local block-transfer counter count-enable daisy-chains. These signals enable the local and VMEbus higher-order address counters; two local address counters (a master block transfer and a slave block transfer) and a single VMEbus address counter. The local address counters share the LCIN*/LCOUT* count-enable daisy-chain. These signals are multiplexed within the CY7C964 and enable counting for the proper counter depending on the current state of the interface. The VCIN*/VCOUT* daisy-chain is dedicated to the VMEbus address counter on the device. When the VCIN* or LCIN* are held Low, counting is enabled for the appropriate counters withing the device. The VCIN* or LCIN* signals do not advance the counters, they just enable counting. The counters increment when these signals are active and the proper increment count control logic sequence occurs. The controller advances the address counters at the proper time during VMEbus and local DMA block transfer operations. For further information on the counter advance control sequence, refer to Chapter 4.5, CY7C964 Operation.

**Table 4−3. CY7C964 Local Signals for VIC068A and VIC64**

| Signal | Description | Interface Comments |
|---|---|---|
| LCIN* | Local address counter count enable | On LSB CY7C964, tie this input Low. On the NMSB device, directly connect to the LCOUT* of the LSB device. For the MSB CY7C964, connect to the LCOUT* of the NMSB device. |
| LCOUT* | Local address counter carry out | On LSB CY7C964, connect this output to the LCIN* input. On the NMSB CY7C964, connect this output to the MSB LCIN* input. For the MSB device do not connect this output. |
| VCIN* | VMEbus address counter count enable | On LSB CY7C964, tie this input Low. On the NMSB device, directly connect to the VCOUT* of the LSB device. For the MSB CY7C964, connect to the VCOUT* of the NMSB device. |
| VCOUT* | VMEbus address counter carry out | On LSB CY7C964, connect this output to the VCIN* input. On the NMSB CY7C964, connect this output to the MSB VCIN* input. For the MSB device do not connect this output. |

## 4.3.4 CY7C964 Address Comparison and Local Signal Group

The implementation of this group of CY7C964 signals is application specific. The MWB* signal and FC1 signal have been included in this section because they are locally generated signals required by the VIC and output directly from the CY7C961. These two signals differ slightly on the VIC; MWB* is an input only, while FC1 is a bidirectional signal that can be driven by the VIC. On the CY7C964 the MWB* and FC1 signals are inputs. These signals should be directly connected to the respective local signals on VIC or CY7C961. For CY7C960 designs, MWB* should be connected to VCC and FC1 should be connected to GND.

**Table 4−4. CY7C964 Address Comparison and Local Signals for VIC068A and VIC64**

| Signal | Description | Interface Comments |
|---|---|---|
| FC1 | Function code 1 signal | Directly connect on all CY7C964s to the same local signal that drives the FC1 signal on the VIC. |
| MWB* | Module wants VMEbus | Directly connect on all CY7C964s to the same local signal that drives the MWB* signal on the VIC. |

**Table 4−4. CY7C964 Address Comparison and Local Signals for VIC068A and VIC64** (continued)

| Signal | Description | Interface Comments |
|---|---|---|
| LDS | Load register select signal | Should be directly connected to LA2 for VIC systems with 32-bit local bus. Refer to text below for further information. |
| STROBE | Latch register control signal | Chip select−like signal for CY7C964 internal comparator mask and comparison registers. See text below for further information. |
| VCOMP* | VMEbus address comparator output | Needs a small amount of external glue logic to validate an combine signals in a parallel high-performance fashion. |

The CY7C964s contain a high-performance programmable VMEbus address equality comparator. The comparator is controlled by two internal, write-only registers: a mask and a compare register. The mask register enables and disables bits of the comparator, and the compare register stores the data pattern that inputs are compared against. VCOMP* is the active-Low comparator match output signal. VCOMP* is driven Low by the CY7C964 when the bit pattern on pins A[7:0] match enabled bits of the compare register. Setting mask register bits to 0 enables the corresponding bits of the compare register. Loading bits of the mask register with 1s places bits of the compare register in don't care or "match anything" state. Loading the mask register with all 0s forces the compare register to match all bits of the pattern on A[7:0]. Setting the mask register to all 1s effectively disables the on-board comparator. VCOMP* will always be Low.

These registers are loaded by supplying the proper data on LD[7:0] and the register address on MWB* and LDS signals. The STROBE input is used to qualify the address and latch the data into the proper internal register. *Figure 4−1* and *Figure 4−2* show the waveforms needed to load the compare and mask registers.

This load cycle operates as follows:

1. The state of LDS and MWB* are latched on the falling edge of STROBE.
2. The data is loaded into the selected register on the rising edge of the STROBE signal.
3. MWB* must be held inactive (High): the state of LDS selects the register to load.
4. If LDS is High at the falling edge of STROBE, the compare register will be loaded; if LDS is Low the data is written to the mask register.

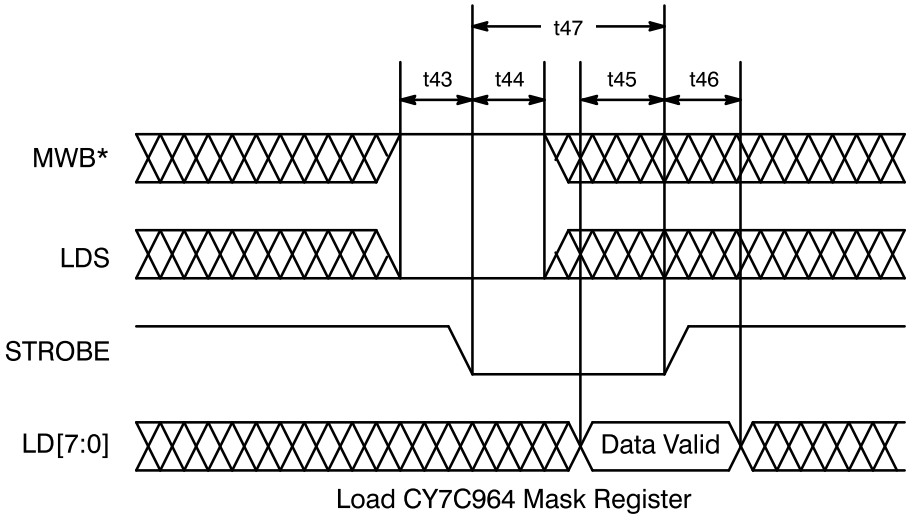


**Figure 4−1. Mask Register Load Timing**

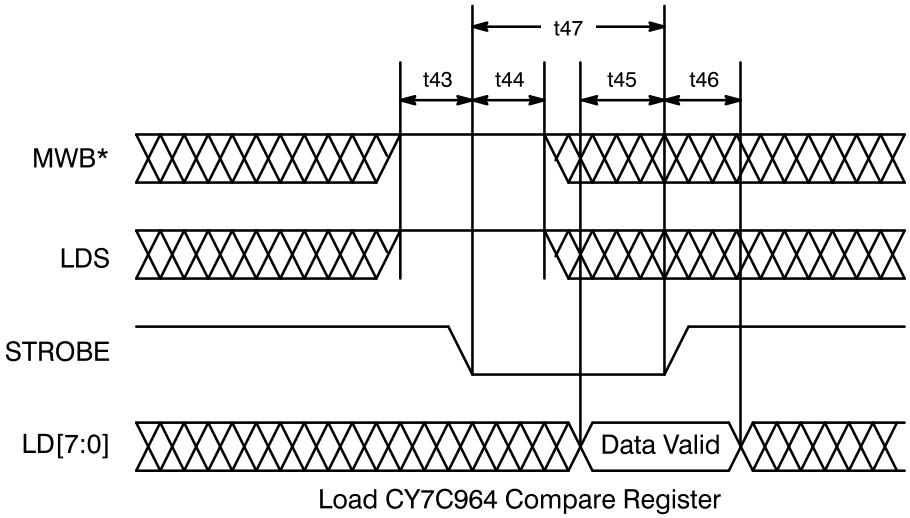


**Figure 4−2. Compare Register Load Timing**

This load cycle can be generated by decoding a separate address region or chip select signal for the CY7C964 comparator registers. In applications with a 32-bit local bus, it is desirable to load all three CY7C964s in parallel by having the host processor perform a 32-bit write cycle to the address region that will activate STROBE. The select signal for the address region is connected to the STROBE input on all CY7C964s. Boards that implement VIC068A or VIC64 interface should connect LDS to LA2, thereby decoding the mask register at the *base address* of the address region and the compare register at the *base address + 4*. LDS also controls the operation of the D64 block transfer data multiplexer/demultiplexer. If it

is not connected to LA2, this logic will not operate properly. For CY7C960 and CY7C961 designs, LDS is connected directly to the controller.

The mask and compare registers can be set to select any contiguous address region on the VMEbus. These registers do not preload and therefore power up in an unknown state. It is advisable to initialize these registers as soon as possible in the system boot sequence. Note that the act of writing the compare register clears the mask register.

The CY7C964 comparator output signal VCOMP* supplies the result from the equality compare logic. VCOMP* drives Low when the input matches the loaded conditions. The CY7C964 VCOMP* signals are not directly compatible with the VIC SLSEL0* and SLSEL1* slave select signals. The short (10 ns) address set-up time to AS* active for VMEbus slave boards does not meet the worst case compare out delay of the CY7C964 VCOMP* signal. combining this with the potential output glitching that can occur with an asynchronous comparator can cause problems for the VIC. It is recommended that the VCOMP* signal be externally filtered prior to being used with the VIC SLSEL0* or SLSEL1* signal. Most applications will require some external comparison logic to combine VCOMP* signals from the NMSB and MSB device, furnishing finer grained VMEbus decoding: this logic can also be used to filter the CY7C964 VCOMP* signals. Note that CY7C960 and CY7C961 have a built-in decode delay feature, which eliminates the need for external filtering.

### 4.3.5 Local Data Swap Buffer Logic

Local Data Swap Buffer logic is a requirement for all 32-bit local bus designs that perform 8- or 16-bit transfers. The swap buffer moves data to and from the lower section of the VMEbus, D[15:0], to the upper segments of the local bus, D[31:16]. VMEbus requires that all 8- and 16-bit data transfers be performed on the D[15:0] section of the bus. The CY7C964s work properly with the VIC-controlled or CY7C960/961-controlled swap buffer. For the VIC case, if an isolation buffer is implemented, care should be taken to ensure that the local data bus is driven to the least-significant CY7C964 during address/mask register programming cycles. One way to ensure this is to assert the CS* and PAS* signals to the VIC068/VIC64, thus causing VIC to assert the ISOBE* signal to the isolation buffer.