

UTMC APPLICATION NOTE

Interfacing the Industry 80C86 and the S μ MMITTM and Shared Memory

Introduction

The monolithic CMOS UT69151 S μ MMIT provides the system designer with an intelligent solution to MIL-STD-1553 multiplexed serial data bus design problems. The S μ MMIT is a single-chip device that implements all three of the defined MIL-STD-1553 functions - Remote Terminal, Bus Controller and Monitor. Operating either autonomously or with a tightly coupled host, the S μ MMIT will solve a wide range of MIL-STD-1553 interface problems. A powerful RISC processing unit provides automatic message handling, message status, general status, and interrupt information. The register-based interface architecture provides many programmable functions as well as extensive information pertinent to device maintenance. In either of the three operating modes, the S μ MMIT can access up to 64K x 16 of external memory (65,536 x 16).

DMA Configuration

The S μ MMIT interfaces easily with the 80C86 that is configured to operate in minimum mode. The configuration assigns the 80C86 as the bus master. The interface requires an external arbitration logic to handle the necessary hand shaking. The following section describes the architecture of the S μ MMIT to 80C86 interface with shared memory.

S μ MMIT DMA Arbitration

The process of the S μ MMIT assuming control of the bus while another master relinquishes control is called DMA Arbitration. Figure 2 depicts a DMA sequence. Three S μ MMIT signals are involved in the arbitration process: DMAR (DMA Request), DMAG (DMA Grant) and DMACK (DMA Acknowledge). The S μ MMIT asserts DMAR when it needs to control the bus. This signal is routed to the arbitration circuit where it is synchronized and sent to the 80C86 as the HOLD input signal. Once received the 80C86 generates the HDLA signal. This signal is then sent to the arbitration circuitry where it is processed and sent to the S μ MMIT as the DMAG signal. Once the S μ MMIT receives the DMAG signal, it generates the DMACK signal and sends it to the arbitration circuitry which uses it to maintain an active HOLD signal. The DMACK will be active while the S μ MMIT is accessing memory. Once the S μ MMIT is finished with its DMA operations, it drops the DMACK signal which allows the 80C86 to regain control of the bus.

S μ MMIT Memory Bus Signals

The S μ MMIT bus signals are split up into three groups: Address, Data, and Control. Five bidirectional signals (A0 - A4) and 11 three-state signals (A5 - A15) comprise the address bus. The word size for the S μ MMIT is always sixteen bits. The data buses consist of 16 bidirectional signals numbered D0 through D15. The memory control signals consist of RCS (RAM Chip Select), RRD (RAM Read), and RWR (RAM Write). RCS goes low to select the memory being accessed. RRD pulses low during the read operation. RWR pulses low during the write operation. Figures

3 and 4 represent the DMA read and write operations. Consult the SμMMIT Family Product Handbook for precise actual timing relationships.

80C86 Architecture

The 80C86 has combined Address and Data bus commonly referred to as time multiplexed. Thus, external latches are necessary to store the address before the data can be placed onto the bus.

Each processor bus cycle consists of at least four CLK cycles. These are referred to as T1, T2, T3 and T4. The address is emitted during T1 and data is transferred during T3 and T4. T2 is used primarily for changing the direction of the bus during read operations.

During T1 of any bus cycles, the Address Latch Enable (ALE) signal is emitted by the processor. At the trailing edge of the pulse, a valid address will be latched into external latches.

The 80C86 can address up to a maximum of 64K I/O byte registers or 32K I/O word registers. The I/O address is emitted on A0 through A15. Address lines A16 through A19 are set to zero during I/O operation.

80C86 DMA Arbitration

The 80C86 DMA, operating in the minimum mode, consists of two signals HOLD and HLDA (Hold Acknowledge). Hold indicates that another master is requesting access to the bus. Once this is received, the 80C86 will relinquish control at time T4 or T1 of a clock cycle. At this time, the HLDA signal will be emitted and will stay active as long as the HOLD signal is applied. The hold signal is not an asynchronous input. External synchronization should be provided.

80C86 Memory and I/O Bus Signals

The 80C86 uses four signals to control data flow. These consist of M/IO (Memory/Input output), DT/R (Data Transfer/Receive), RD (Read), and WR (Write). The M/IO signal is used to choose between memory access and I/O register access and becomes active during T4 of the preceding bus cycle. The DT/R signal is used to control direction of data flow through external transceivers. The RD signal is used to access devices connected to the bus and is active during T2 and T3. The WR signal is used to write to devices connected to the bus and is active during T2 and T3. All of the four signals mentioned above are placed into a high impedance mode when the HLDA signal is active.

80C86 Memory Organization

The Processor has 20 address bits, which is addressed between 00000H and FFFFFH. Certain locations in memory are reserved for specific CPU operations. Locations from address FFFF0H through FFFFFH are reserved for jump to initial program loading routine. Following a RESET, the CPU will always begin execution from this location. Therefore, these memory locations must reside in PROM. Locations 00000H through 003FFH are reserved for interrupt handling routines. This application note stores these memory addresses are to be stored in PROM. This application note also allows the 80C86 and the SμMMIT to access 64K x 16 of SRAM. Locations 400000H

through 47FFFH contains one 32K x 16 bank of memory and 800000H through 87FFFH contains the other 32K x 16 bank of memory. Note: Address bits A19 and A18 were chosen as chip selects for the SRAM arbitration. However, it is not mandatory to use these bits. The designer is free to choose other bits if it would better suit their needs. Figure 5 shows how the memory was organized for this application note.

DMA Hardware Configuration

Figure 1 illustrates a simplified block diagram containing the SμMMIT and the 80C86, along with the DMA arbitration circuitry, latches and memory devices. The arbitration circuit is blocked out in Figure 1. This block shows how it could easily be synthesized inside a UTMC RAD-PALTM UT22VP10. The ABEL program below implements the design shown. The diagram also shows the use of three RHMSI UT55ACTS373 latches. These are used to store the address emitted by the 80C86.

DMA Arbitration ABEL File

```

module arbitration
title 'DMA arbitration'
////////////////////////////////////////////////////////////////
//This module implements the necessary arbitration needed to interface//
//the 80C86 with the SμMMIT.                                     //
////////////////////////////////////////////////////////////////

Declarations
    arbitration DEVICE 'UT22VP10'
“Inputs
    RD, WR, DTR, CLK, MIO, HDLA, A18, A19    pin;
    DMAR, DMACK, RCS                          pin
“Outputs
    SCS, RDWR, PCS, CS1, CS2, DMAG          pin istype 'com';
    HOLD                                     pin istype 'reg';

Equations
    SCS      = RD & WR + MIO;
    RDWR     = !DTR;
    DMAG     = !HLDA;
    PCS      = !((( !HDLA & MIO) & (!A19 & !A18)
                # (( !HDLA & MIO) & (A19 & A18)));
    CS1      = !(PCS & A18 & !RSC);
    CS1      = !(PCS & A19 & !RSC);
    HOLD.clk = !CLK;
    HOLD.d   = !(DMAR & DMACK);

end

```