



Discrete Logic Replacement

Dual Port RAM Applications

Author: Anton Tachev
Sofia, Bulgaria

INTRODUCTION

This application note is dedicated to implementation of the PIC12C671 as Dual Port RAM (DP_RAM). This term in this article will mean a RAM structure, which is made accessible from two independent microcontrollers or microprocessors. Below are discussed some basic and specific realizations of DP_RAM with a PIC12C671. I hope such circuit solutions will be useful in designing multiprocessing systems. PIC12C671, in chip RAM, is used as a RAM structure, until access control is realized through the processor's facilities of a PICmicro.

BASIC STRUCTURES

A Simple Nibble's Access (Figure 1)

Description

The DP_RAM is connected to the microcontrollers through the I/O pins. The control sequence for data transfer is described by the items below:

1. The mC1 makes request to obtain data bus by generating + DP_RAM REQ1(PB3) to mC2.
2. This microcontroller accepts the request and puts your pins (Figure 1) into a high-impedance state. At the same time, it acknowledges the request to mC1 by the signal, DP_RAM ACKN.
3. By one of the control words on PB0, PB1 (00 - read; 01 - write), mC1 points to the PIC12C671 direction of data transfer (read or write).
4. The PIC12C671 performs a cycle program, which places the nibbles to data bus D0-D3/ D4-D7. At the same time, the PIC12C671 switches your GP4 as an output, and generates Strobe impulses to the INT input of mC1 synchronizing, which is in fact, data transfer. According to the control words, the PIC12C671 reads or writes data bus via port pins GP0-GP4. The numbers of transferred nibbles and the beginning address is controlled from the PIC12C671.

5. The transfer may be completed in several ways:
 - Waiting for a given fixed time interval of the last Strobe
 - Unique code in the last nibble
 - Initial set of the numbers of nibbles (e.g., 128 nibbles) in the program code of mC1 and PIC12C671
6. The procedure of transfer is completed when DP_RAM REQ1 goes low.
7. The data transfer between mC2 and the PIC12C671 is similar.

A Byte's (Parallel) Connection (Figure 2)

Description

The next step in the evolution of the BASIC STRUCTURE from Figure 1 is a byte's parallel connection. The control sequences are similar to nibble's connection with some peculiarities:

After initialization with the control word, GP4 of PIC_2 remains input. In this way, the Strobe from GP4 (PIC_1) synchronizes transfer of the LSB nibbles into PIC_1 and MSB nibbles into PIC_2.

Up to now, we have viewed DP_RAM applications, which have a separate and consequence access to microcontrollers. The next section offers one application with a given type of parallel access, called quasi-parallel.

Microchip Technology Incorporated, has been granted a nonexclusive, worldwide license to reproduce, publish and distribute all submitted materials, in either original or edited form. The author has affirmed that this work is an original, unpublished work and that he/she owns all rights to such work. All property rights, such as patents, copyrights and trademarks remain with author.

Discrete Logic Replacement

A Cross Connection – Quasi-Parallel (Figure 3)

Description

The specific feature of this structure is a cross connection between the high and low data buses of microcontrollers (mC1 and mC2). In this way, each microcontroller is capable of transferring data to the same PICmicro™. For example: mC1 reads from PIC_1, until mC2 reads from PIC_2. The logic state of port pins PB0, PB1 changes the data transfer. For example: mC1 writes to PIC_2 while mC2 writes to PIC_1.

So far, we saw that the dual port RAM was connected to the microcontroller's ports. In many cases, it is important and convenient to connect DP_RAM to the microcontroller's open bus. Such structures are subject in the following applications.

SPECIFIC STRUCTURES

Open-microcontroller Bus Configuration (Figure 4)

Description

The connection of PIC12C671 as a DP_RAM to open microcontroller buses is necessary in cases where older microcontroller family designs such as 8051 and MC68HC11Ex, Ax, where the usage of an open bus is most inherent. Figure 4 illustrates this type of connection. The basic principle of operation is not different from the one already described in the other paragraphs. The microcontroller's bus transfers data with DP_RAM as an I/O device. INT inputs are not used in the synchronization process. This is accomplished by some I/O pins accepting DP_RAM READY signal.

The last application concerns the implementation of DP_RAM between the microcontrollers of the PIC16C64/65 families, which have an additional microprocessor parallel slave port. The connection is illustrated on Figure 5.

The essential feature in this connection is in the initial phase, when the host PIC16C64 (PIC_1 or PIC_2) sets the mode of operation (read or write). The PORTE works as a common I/O port. In the process of transferring the PORTE sets as a slave microprocessor port, the pins CS0 are shorted to GND, but the RE and WE signals are program in conjunction with CS appearance, generating from Pica's program code.