AN778

Implementing the External Memory Interface on PIC18C601/801 MCUs

Author: *Gaurang Kavaiya*
Microchip Technology Inc.

INTRODUCTION

The PIC18C601 and PIC18C801 are the very first members of Microchip's PIC18 family that are ROM-less microcontrollers — that is, they have no on-chip program memory. Both offer the enhanced PIC18 architecture, along with the ability to use different types and sizes of external program memory to exactly fit any application. In addition to standard 1.5 Kbytes of general purpose RAM, the PIC18C601 can address up to 256 Kbytes of external program memory, while the PIC18C801 can address up to 2 Mbytes of external program memory. With this amount of available addressable space, the PIC18C601/801 devices become ideal candidates for more complex applications (e.g. TCP/IP stacks), developed with high level programming languages, such as 'C'.

In addition, PIC18C601/801 devices also make in-system programming possible with its configurable general purpose RAM ("Boot RAM"), which can be configured as a program memory. When program execution takes place from Boot RAM, the external memory bus can be mapped to port I/O. This feature enables the device to perform virtually any programming algorithm in software which does not conform to standard timing requirements. Also, the PIC18C801 offers a completely "glueless" external memory interface solution with its 8-bit De-Multiplexed Interface mode.

The PIC18C601/801 devices provide up to two programmable chip select signals, to partition address space into two different memories. It also provides one programmable I/O chip select signal to locate an 8 Kbyte memory mapped I/O region anywhere in the address space, except the lower 8 Kbyte space.

Given the number and types of memories available today, finding and interfacing memory to the PIC18C601/801 devices potentially becomes a challenging task. This application note describes the PIC18C601/801 external memory interface modes, as well as the methods for interfacing different types of memories with PIC18C601/801. It is expected that the reader will already be familiar with the general PIC18 architecture and instruction set.

This application note is divided into the following sections.

- ***External Program Memory Interface Modes*** provide information on the various memory interface modes available with the PIC18C601/801 microcontrollers. It also discusses the requirements for configuring the controllers and using Table Read and Table Write operations.
- ***Memory Mapping*** explains the memory maps and mapping techniques for the PIC18C601/801 devices, using the on-chip programmable chip select signals.
- ***Memory Mapped I/O*** explains how to use the external memory interface as a mapped I/O port for peripheral devices.
- ***Memory Devices and Interface*** provides information on selecting and implementing interfaces for various types of memory devices.
- ***Memory Timing Analysis*** explains the memory timing requirements for PIC18C601/801 devices, and how to assess memory devices for compatability. The goal of this section is to answer one of the most frequently asked questions: "What memory speed should I use with my *x* MHz CPU?"

1.0 EXTERNAL PROGRAM MEMORY INTERFACE MODES

PIC18C601/801 controllers can be configured to run in either 8-bit or 16-bit Data mode. The appropriate mode is selected by setting the Bus Width configuration bit (BW) in the Configuration register CONFIG2L. The default configuration for the controllers is 16-bit, but this can be changed to 8-bit with the appropriate device programmer.

The 16-bit Data mode is available only in Multiplexed mode, regardless of part selection. Depending on the part chosen, the 8-bit Data mode may be either multiplexed or de-multiplexed; the PIC18C601 supports only the Multiplexed mode, while the PIC18C801 provides only the De-Multiplexed mode.

If the external address bus is configured as an 8-bit external interface, some of the external control signals used in the 16-bit external interface will be mapped to port I/O functions. However, when the external address bus is configured as 16-bit external interface, all of the external control signals used for the 8-bit external interface will also be used for the 16-bit interface. External components are needed to de-multiplex the address for all interface modes. The exception is the PIC18C801 configured in 8-bit Interface mode (Section 1.3.2).

REGISTER 1-1: CONFIGURATION REGISTER 2 LOW (CONFIG2L: BYTE ADDRESS 300002h)

U-0	R/P-1	U-0	U-0	U-0	U-0	U-0	R/P-1
—	BW	—	—	—	—	—	$\overline{\text{PWRTEN}}$
bit 7							bit 0

bit 7 **Unimplemented:** Read as ’0’

bit 6 **BW:** External Bus Data Width bit
1 = 16-bit external bus mode
0 = 8-bit external bus mode

bit 5-1 **Unimplemented:** Read as ’0’

bit 0 **$\overline{\text{PWRTEN}}$:** Power-up Timer Enable bit
1 = PWRT disabled
0 = PWRT enabled

Legend:		
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as ‘0’
- n = Value when device is unprogrammed		u = Unchanged from programmed state

1.1 Physical Implementation

The External Memory Interface is implemented with up to 26 pins on the PIC18C601, and up to 38 pins on the PIC18C801. These pins are reserved for external address and data bus functions. and are also multiplexed with port pins. The port functions are only enabled when:

- Program execution takes place in internal Boot RAM, and
- The EBDIS bit in the MEMCON register is set (MEMCON<7> = '1').

Tables 1-1 and 1-2 list the typical mappings of external bus functions on I/O pins for the PIC18C601 and PIC18C801, respectively.

TABLE 1-1: TYPICAL PORT FUNCTIONS OF PIC18C601

Name	16-bit mode	8-bit mode	Function
RD0/AD0	AD0	AD0	Input/Output or System Bus Address bit 0 or Data bit 0
RD1/AD1	AD1	AD1	Input/Output or System Bus Address bit 1 or Data bit 1
RD2/AD2	AD2	AD2	Input/Output or System Bus Address bit 2 or Data bit 2
RD3/AD3	AD3	AD3	Input/Output or System Bus Address bit 3 or Data bit 3
RD4/AD4	AD4	AD4	Input/Output or System Bus Address bit 4 or Data bit 4
RD5/AD5	AD5	AD5	Input/Output or System Bus Address bit 5 or Data bit 5
RD7/AD6	AD6	AD6	Input/Output or System Bus Address bit 6 or Data bit 6
RD6/AD7	AD7	AD7	Input/Output or System Bus Address bit 7 or Data bit 7
RE0/AD8	AD8	AD8	Input/Output or System Bus Address bit 8 or Data bit 8
RE1/AD9	AD9	AD9	Input/Output or System Bus Address bit 9 or Data bit 9
RE2/AD10	AD10	AD10	Input/Output or System Bus Address bit 10 or Data bit 10
RE3/AD11	AD11	AD11	Input/Output or System Bus Address bit 11 or Data bit 11
RE4/AD12	AD12	AD12	Input/Output or System Bus Address bit 12 or Data bit 12
RE5/AD13	AD13	AD13	Input/Output or System Bus Address bit 13 or Data bit 13
RE6/AD14	AD14	AD14	Input/Output or System Bus Address bit 14 or Data bit 14
RE7/AD15	AD15	AD15	Input/Output or System Bus Address bit 15 or Data bit 15
RG0/ALE	ALE	ALE	Address Latch Enable (ALE) Control pin
RG1/$\overline{OE}$	$\overline{OE}$	$\overline{OE}$	Output Enable ($\overline{OE}$) Control pin
RG2/$\overline{WRL}$	$\overline{WRL}$	$\overline{WRL}$	Write Low ($\overline{WRL}$) Control pin
RG3/$\overline{WRH}$	$\overline{WRH}$	RG3	Input/Output or System Bus Write High ($\overline{WRH}$) Control pin
RG4/BA0	BA0	BA0	Input/Output or System Bus Byte Address bit 0
RF7/$\overline{LB}$	$\overline{LB}$	RF7	Input/Output or System Bus Lower Byte Enable ($\overline{LB}$) Control pin
RF6/$\overline{UB}$	$\overline{UB}$	RF6	Input/Output or System Bus Upper Byte Enable ($\overline{UB}$) Control pin
RF3/$\overline{CSIO}$	$\overline{CSIO}$	$\overline{CSIO}$	Input/Output or System Bus Chip Select I/O
RF4/AD16	AD16	AD16	Input/Output or System Bus Address bit 16 or Data bit 16
RF5/$\overline{CS1}$	$\overline{CS1}$	$\overline{CS1}$	Input/Output or System Bus Chip Select 1

TABLE 1-2: TYPICAL PORT FUNCTIONS OF PIC18C801

Name	16-bit mode	8-bit mode	Function
RD0/AD0	AD0	A0	Input/Output or System Bus Address bit 0 or Data bit 0
RD1/AD1	AD1	A1	Input/Output or System Bus Address bit 1 or Data bit 1
RD2/AD2	AD2	A2	Input/Output or System Bus Address bit 2 or Data bit 2
RD3/AD3	AD3	A3	Input/Output or System Bus Address bit 3 or Data bit 3
RD4/AD4	AD4	A4	Input/Output or System Bus Address bit 4 or Data bit 4
RD5/AD5	AD5	A5	Input/Output or System Bus Address bit 5 or Data bit 5
RD7/AD6	AD6	A6	Input/Output or System Bus Address bit 6 or Data bit 6
RD6/AD7	AD7	A7	Input/Output or System Bus Address bit 7 or Data bit 7
RE0/AD8	AD8	A8	Input/Output or System Bus Address bit 8 or Data bit 8
RE1/AD9	AD9	A9	Input/Output or System Bus Address bit 9 or Data bit 9
RE2/AD10	AD10	A10	Input/Output or System Bus Address bit 10 or Data bit 10
RE3/AD11	AD11	A11	Input/Output or System Bus Address bit 11 or Data bit 11
RE4/AD12	AD12	A12	Input/Output or System Bus Address bit 12 or Data bit 12
RE5/AD13	AD13	A13	Input/Output or System Bus Address bit 13 or Data bit 13
RE6/AD14	AD14	A14	Input/Output or System Bus Address bit 14 or Data bit 14
RE7/AD15	AD15	A15	Input/Output or System Bus Address bit 15 or Data bit 15
RH0/A16	A16	A16	Input/Output or System Bus Address bit 16
RH1/A17	A17	A17	Input/Output or System Bus Address bit 17
RH2/A18	A18	A18	Input/Output or System Bus Address bit 18
RH3/A19	A19	A19	Input/Output or System Bus Address bit 19
RG0/ALE	ALE	ALE	Address Latch Enable (ALE) Control pin
RG1/$\overline{OE}$	$\overline{OE}$	$\overline{OE}$	Output Enable ($\overline{OE}$) Control pin
RG2/$\overline{WRL}$	$\overline{WRL}$	$\overline{WRL}$	Write Low ($\overline{WRL}$) Control pin
RG3/$\overline{WRH}$	$\overline{WRH}$	RG3	Input/Output or System Bus Write High ($\overline{WRH}$) Control pin
RG4/BA0	BA0	BA0	Input/Output or System Bus Byte Address bit 0
RF7/$\overline{LB}$	$\overline{LB}$	RF7	Input/Output or System bus Lower Byte Enable ($\overline{LB}$) Control pin
RF6/$\overline{UB}$	$\overline{UB}$	RF6	Input/Output or System Bus Upper Byte Enable ($\overline{UB}$) Control pin
RF3/$\overline{CSIO}$	$\overline{CSIO}$	$\overline{CSIO}$	Input/Output or System Bus Chip Select I/O
RF4/$\overline{CS2}$	$\overline{CS2}$	$\overline{CS2}$	Input/Output or System Bus Chip Select 2
RF5/$\overline{CS1}$	$\overline{CS1}$	$\overline{CS1}$	Input/Output or system bus Chip Select 1
RJ0/D0	RJ0	D0	Input/Output or System Bus Data bit 0
RJ1/D1	RJ1	D1	Input/Output or System Bus Data bit 1
RJ2/D2	RJ2	D2	Input/Output or System Bus Data bit 2
RJ3/D3	RJ3	D3	Input/Output or System Bus Data bit 3
RJ4/D4	RJ4	D4	Input/Output or System Bus Data bit 4
RJ5/D5	RJ5	D5	Input/Output or System Bus Data bit 5
RJ6/D6	RJ6	D6	Input/Output or System Bus Data bit 6
RJ7/D7	RJ7	D7	Input/Output or System Bus Data bit 7

1.2 16-bit External Interfaces

The 16-bit External mode interface can be configured by setting BW bit in Configuration register, CONFIG2L. Pins AD15:AD0 carry multiplexed address and data information, while pins A19:A16 carry address information only.

The BA0 signal indicates an even or odd address. Since all memory accesses by the controller in 16-bit mode are word-aligned, BA0 is not required and should be left unconnected, even though it is still active. For 16-bit instruction fetch mode, the $\overline{OE}$ output enable signal will enable both bytes of program memory at once to get a 16-bit word.

PIC18C601/801 controllers divide their instruction cycle into four quarters, Q1 through Q4. During Q1, ALE is enabled while address information (A15:A0) is placed on pins AD15:AD0. At the same time, the upper address information (Ax:A16) is available on the upper address bus. On the negative edge of ALE, the address is latched in the external latch. At the beginning of Q3, the $\overline{OE}$ output enable (active low) signal is generated. At the end of Q4, $\overline{OE}$ goes high and data (16-bit word) is read from memory at the low-to-high transition edge of $\overline{OE}$.

The 16-bit mode is divided into three sub-categories, depending on how external memory is organized:

- 16-bit memory with two individual 8-bit memory chips (Byte Write mode)
- 16-bit memory with Byte Select mode
- True 16-bit memory (16-bit Word Write mode)

The control signals used for the 16-bit modes are listed in Table 1-3.

TABLE 1-3: 18C601/801 16-BIT MODE CONTROL SIGNALS

Name	18C601 16-bit mode	18C801 16-bit mode	Function
RG0/ALE	ALE	ALE	Address Latch Enable (ALE) Control pin
RG1/$\overline{OE}$	$\overline{OE}$	$\overline{OE}$	Output Enable ($\overline{OE}$) Control pin
RG2/$\overline{WRL}$	$\overline{WRL}$	$\overline{WRL}$	Write Low ($\overline{WRL}$) Control pin
RG3/$\overline{WRH}$	$\overline{WRH}$	$\overline{WRH}$	Write High ($\overline{WRH}$) Control pin
RG4/BA0	BA0	BA0	Byte Address bit 0
RF3/$\overline{CSIO}$	$\overline{CSIO}$	$\overline{CSIO}$	Chip Select I/O (see Section 3.3)
RF4/$\overline{CS2}$	N/A	$\overline{CS2}$	Chip Select 2 (see Section 3.2)
RF5/$\overline{CS1}$	$\overline{CS1}$	$\overline{CS1}$	Chip Select 1 (see Section 3.1)
RF6/$\overline{UB}$	$\overline{UB}$	$\overline{UB}$	Upper Byte Enable ($\overline{UB}$) Control pin
RF7/$\overline{LB}$	$\overline{LB}$	$\overline{LB}$	Lower Byte Enable ($\overline{LB}$) Control pin
I/O	I/O	I/O	I/O as BYTE/$\overline{WORD}$ Control pin for JEDEC FLASH (with Byte Select mode)

1.2.1 TABLE READ AND WRITE OPERATIONS IN 16-BIT MODE

In addition to the program memory space already covered, PIC18C601/801 devices also have a data memory space. These memory spaces differ in their organization: program memory is 16-bits wide, while data memory is 8-bits wide. To move information between these differently configured spaces, the Table Read (TBLRD) and Table Write (TBLWT) instructions have been provided.

Table Read operations retrieve data from program memory and place it into the data memory space. Table Write operations, on the other hand, store data from the data memory space into program memory. Table operations work with byte entities, moving data through an 8-bit register, TABLAT. A table block containing data is not required to be word aligned, so a table block can start or end at any byte address.

All of the 16-bit modes require special handling of Table Write operations. The appropriate bits in the MEMCON register (WM, or MEMCON<1:0>) must be set prior to any Table Write operation.

At power-on, the default content of MEMCON sets the following system parameters:

- System bus is enabled
- Program RAM is configured as GPR memory from 400h to 5FFh
- A 3-wait state cycle count for Table Reads and Writes is selected
- Table Write operations are set for Byte Write mode

Register 1-2 gives the details of the MEMCON configuration bits.

Note:	The WM<1:0> bits have no effect when the device is configured for 8-bit execution.

REGISTER 1-2: MEMCON REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
EBDIS	PGRM	WAIT1	WAIT0	—	—	WM1	WM0
bit7							bit0

bit 7 **EBDIS**: External Bus Disable bit
1 = External system bus disabled, all external bus drivers are mapped as I/O ports
0 = External system bus enabled and I/O ports are disabled

bit 6 **PGRM**: Program RAM Enable bit
1 = 512 bytes of internal RAM enabled as internal program memory from location 1FFE00h to 1FFFFFh, external program memory at these locations is unused. Internal GPR memory from 400h to 5FFh is disabled and returns 00h.
0 = Internal RAM enabled as internal GPR memory from 400h to 5FFh. Program memory from location 1FFE00h to 1FFFFFh is configured as external program memory.

bit 5-4 **WAIT<1:0>**: Table Reads and Writes Bus Cycle Wait Count bits
11 = Table reads and writes will wait 0 TCY
10 = Table reads and writes will wait 1 TCY
01 = Table reads and writes will wait 2 TCY
00 = Table reads and writes will wait 3 TCY

bit 3-2 **Unimplemented**: Read as '0'

bit 1-0 **WM<1:0>**: TBLWT Operation with 16-bit Bus bits
1X = Word Write mode: TABLAT<0> and TABLAT<1> word output, $\overline{WRH}$ active when TABLAT<1> written
01 = Byte Select mode: TABLAT data copied on both Most Significant Byte and Least Significant Byte, $\overline{WRH}$ and ($\overline{UB}$ or $\overline{LB}$) will activate
00 = Byte Write mode: TABLAT data copied on both Most Significant Byte and Least Significant Byte, $\overline{WRH}$ or $\overline{WRL}$ will activate

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
- n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

1.2.1.1 TABLAT and TBLPTR Registers

Two control registers are used in conjunction with the TBLRD and TBLWT instructions. They are:

- TABLAT register
- TBLPTR registers

The Table Latch (TABLAT) is an 8-bit register mapped into the SFR space. The Table Latch is used to hold 8-bit data during data transfers between program memory and data memory.

The Table Pointer (TBLPTR) addresses a byte within the program memory. The TBLPTR is comprised of three special function registers:

- Table Pointer Upper byte (TBLPTRU)
- Table Pointer High byte (TBLPTRH)
- Table Pointer Low byte (TBLPTRL)

These three registers join to form a 21-bit wide pointer, which allows the device to address up to 2 Mbytes of program memory space. TBLPTR is used by the TBLRD and TBLWT instructions. During Table Read and Table Write operations, the Least Significant bit of TBLPTR is copied to BA0. The remainder of TBLPTR is copied to pins AX:A0 of the external address bus, with the upper limit being determined by the microcontroller and mode being used. As an example, when the PIC18C801 is being used, the value of TBLPTR<0> appears on BA0, while the values of TBLPTR<20:1> appear on pins A19:A0.

1.2.1.2 Table Read

The TBLRD instruction is used to retrieve data from external program memory and place it into data memory. TBLPTR points to a byte address in external program memory space. Executing TBLRD, places the byte into TABLAT. In addition, TBLPTR can be modified automatically for the next Table Read operation. Table Reads from external program memory are logically performed one byte at a time.

If the external interface is 8-bit, the bus interface circuitry in TABLAT will load the external value into TABLAT.

If the external interface is 16-bit, interface circuitry in TABLAT will select either the high, or the low byte of the data from the 16-bit bus, based on the Least Significant bit of the address. That is, when LSb is 0, the lower byte (D<7:0>) is selected; when LSb is 1, the upper byte (D<15:8>) is selected.

1.2.1.3 Table Write

The TBLWT instruction stores data from the data memory space into external program memory. PIC18C601/801 devices perform Table Writes, one byte at a time. Table Writes to external memory are two-cycle instructions, unless wait states are enabled.

If the external interface is 8-bit, the bus interface circuitry in TABLAT will copy its value to the external data bus. If the external interface is 16-bit, interface Table Writes depend on the type of external device that is connected and the WM<1:0> bits in the MEMCON register. The code in Example 1-1 describes the use of the Table Write operation for the 16-bit external interface.

EXAMPLE 1-1: USING THE TBLWT INSTRUCTION WITH THE 16-BIT INTERFACE

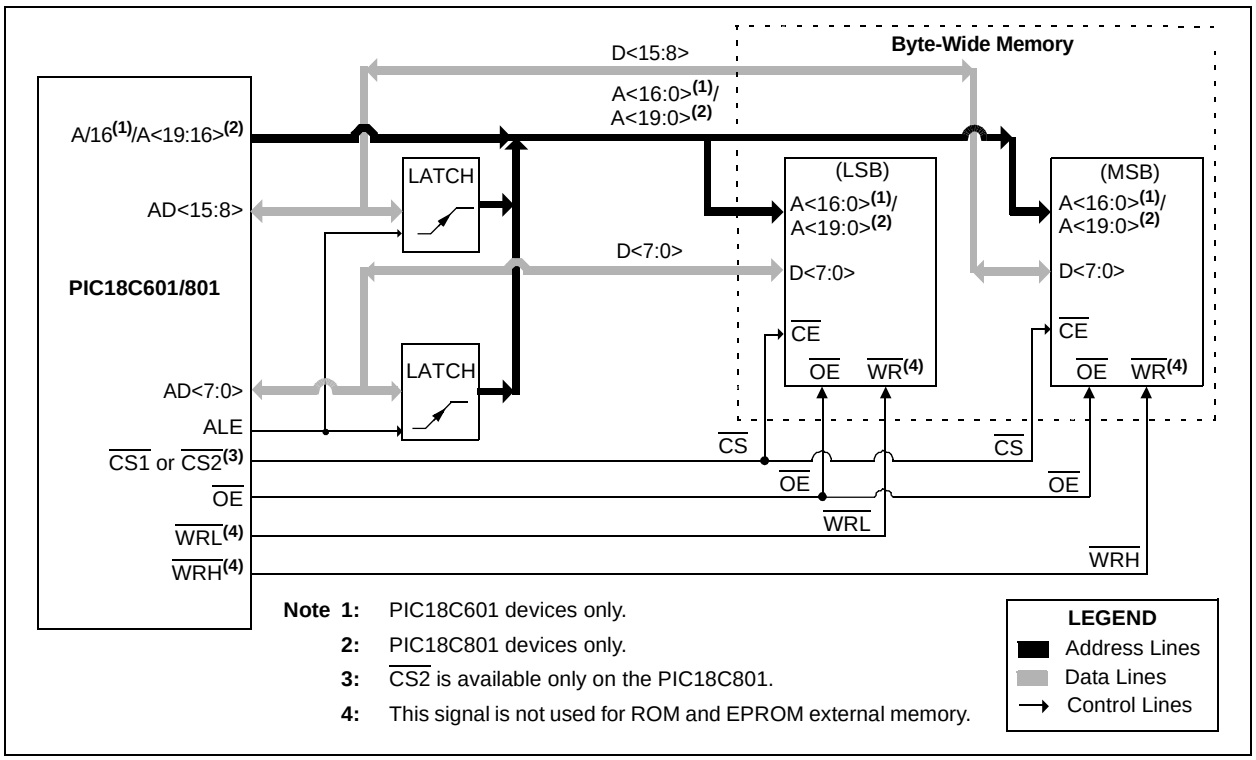
```
movlw       UPPER   (SampleTable)          ;Initialize Table Pointer
movwf       TBLPTRU                        ;with the starting address
movlw       HIGH    (SampleTable)          ;of the Table
movwf       TBLPTRH                        ;
movlw       LOW     (SampleTable)          ;
movwf       TBLPTRL                        ;
movlw       LOW     (DataWord)             ;Load table latch with low byte
movwf       TABLAT                         ;of value to write
tblwt*+                                    ;Write to Program memory and increment Table Pointer
movlw       HIGH    (DataWord)             ;Load W register with high byte of value to write
movwf       TABLAT                         ;Transfer high byte of value to table latch
tblwt*                                     ;Write to next location/Word
movf        Count,W                        ;Next Instruction for logic
```

1.2.2 EXTERNAL TABLE WRITE IN 16-BIT BYTE WRITE MODE

This mode is used for two separate 8-bit memories connected for 16-bit operation. This generally includes basic EPROM and FLASH devices. It allows Table Writes to byte-wide external memories. During a `TBLWT` instruction cycle, the TABLAT data is presented on the upper and lower bytes of the AD15:AD0 bus. The appropriate $\overline{WRH}$ or $\overline{WRL}$ control line is strobed on the LSb of the TBLPTR.

Figure 1-1 shows a typical implementation of the Byte Write mode.

FIGURE 1-1: 16-BIT BYTE WRITE MODE



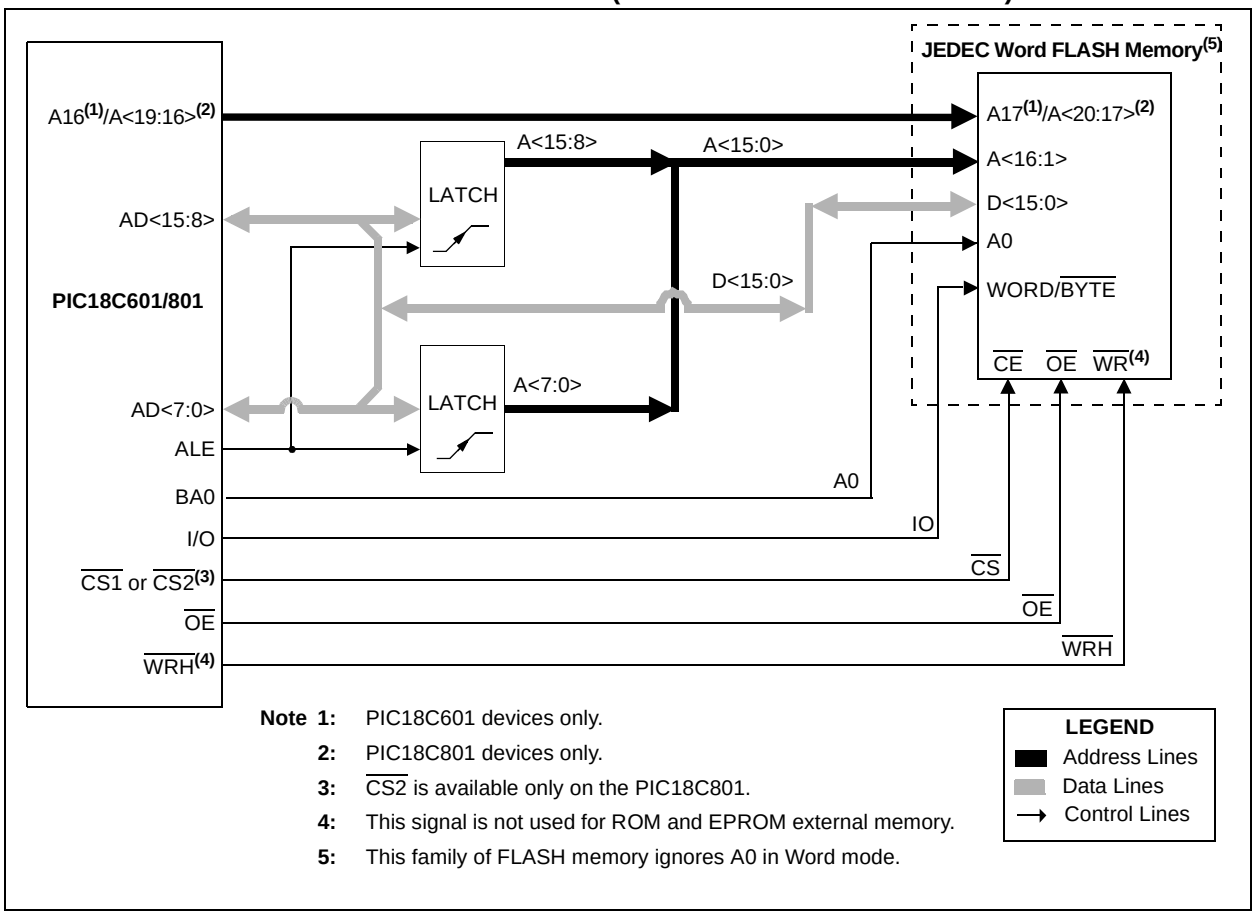
1.2.3 EXTERNAL TABLE WRITE IN 16-BIT BYTE SELECT MODE

This mode allows Table Write operations to word-wide external memories with byte selection capability. This generally includes both word-wide FLASH and SRAM devices. During a `TBLWT` cycle, the TABLAT data is presented on the upper and lower byte of the AD15:AD0 bus. The $\overline{WRH}$ signal is strobed for each write cycle; the $\overline{WRL}$ pin is not used. The BA0 or $\overline{UB}$/$\overline{LB}$ signals are used to select the byte to be written, based on the LSb of the TBLPTR register.

FLASH and SRAM devices use different control signal combinations to implement Byte Select mode. JEDEC standard FLASH memories require that a controller I/O port pin be connected to the memory's BYTE/$\overline{WORD}$ pin to provide the select signal. They also use the BA0 signal from the controller as a byte address (Figure 1-2). JEDEC standard static RAM memories, on the other hand, use the $\overline{UB}$ or $\overline{LB}$ signals to select the byte (Figure 1-3).

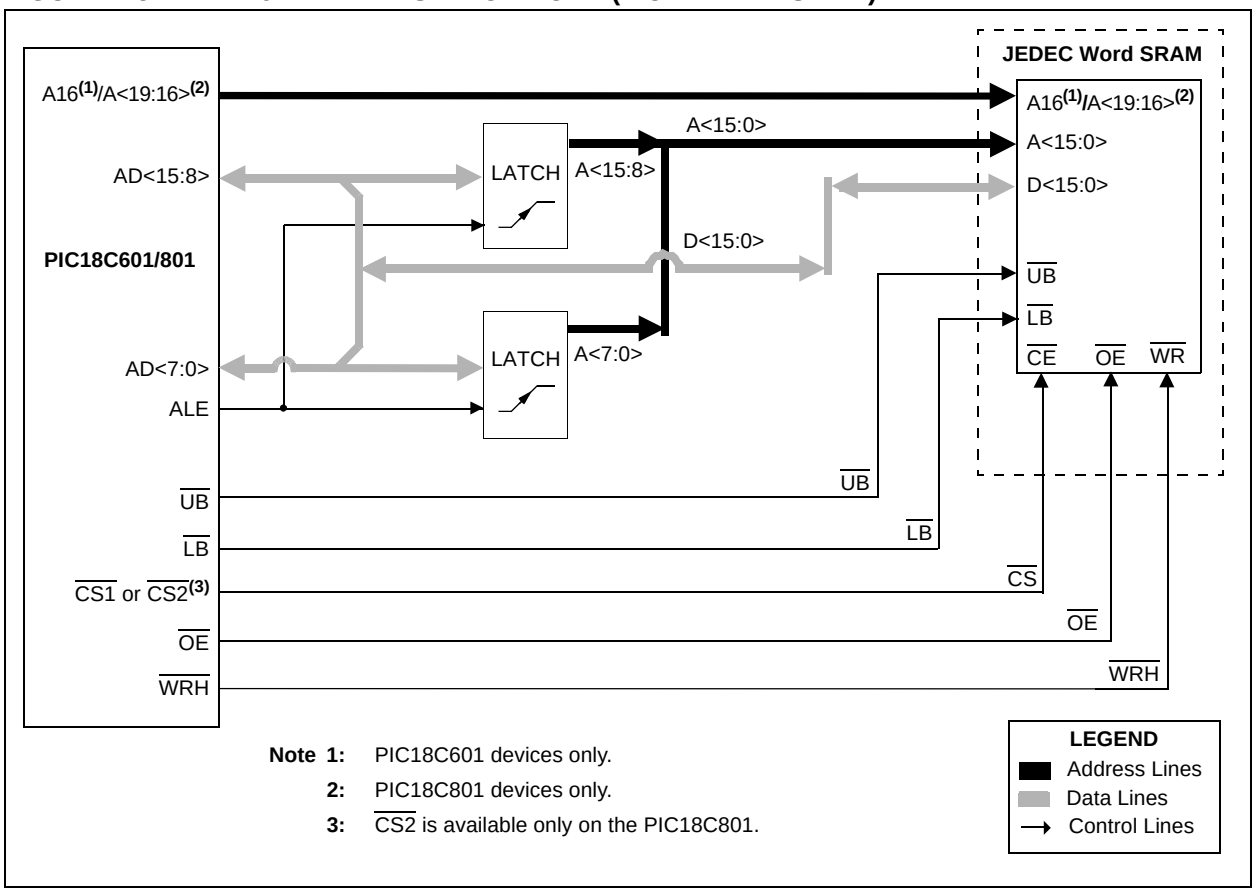
Note: To program a 16-bit FLASH memory with byte select capability, user firmware must dynamically change FLASH memory access mode from Word to Byte mode. This can be achieved by connecting one I/O line to a FLASH Memory mode pin and making sure that the FLASH device is setup in 16-bit mode on power-up. Since instruction fetches are done in 16-bit mode only, care must be taken that FLASH mode is changed only when execution is taking place from Boot RAM.
For additional information, refer to the PIC18C601/801 Device Data Sheet (DS39541).

FIGURE 1-2: 16-BIT BYTE SELECT MODE (WORD-WIDE FLASH MEMORY)



Note 1: PIC18C601 devices only.
2: PIC18C801 devices only.
3: $\overline{CS2}$ is available only on the PIC18C801.
4: This signal is not used for ROM and EPROM external memory.
5: This family of FLASH memory ignores A0 in Word mode.

FIGURE 1-3: 16-BIT BYTE SELECT MODE (WORD-WIDE SRAM)



Note 1: PIC18C601 devices only.

2: PIC18C801 devices only.

3: $\overline{CS2}$ is available only on the PIC18C801.

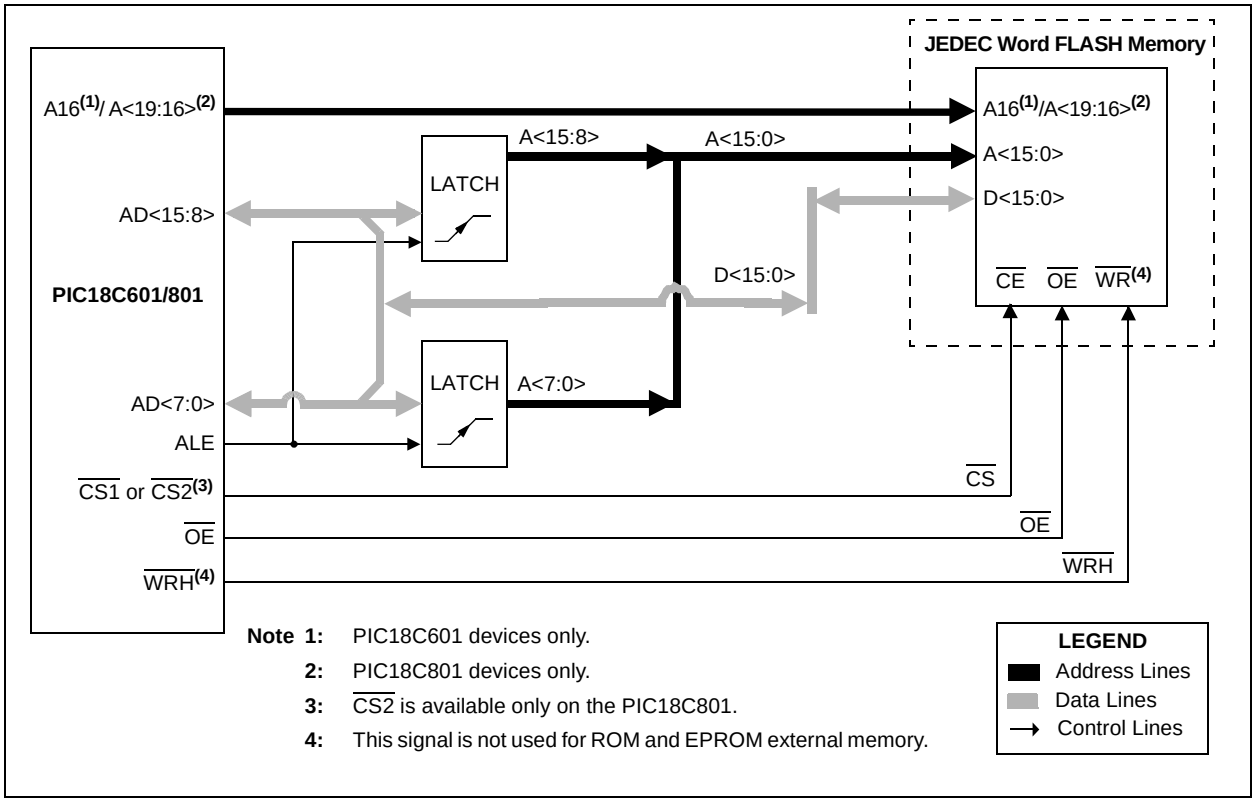
1.2.4 EXTERNAL TABLE WRITE IN 16-BIT WORD WRITE MODE

This mode is used for word-wide memories, which includes some of the EPROM and FLASH type memories. This mode allows opcode fetches and Table Reads from all forms of 16-bit memory, and Table Writes to any type of word-wide external memories. This method makes a distinction between TBLWT cycles to even or odd addresses. During a TBLWT cycle to an even address (TBLPTR<0> = '0'), the TABLAT data is transferred to a holding latch and the external address data bus is tri-stated for the data portion of the bus cycle. No write signals are activated.

During a TBLWT cycle to an odd address (TBLPTR<0> = '1'), the TABLAT data is presented on the upper byte of the AD15:AD0 bus. The contents of the holding latch are presented on the lower byte of the AD15:AD0 bus. The $\overline{WRH}$ signal is strobed for each write cycle; the $\overline{WRL}$ pin is unused. The signal on the BA0 pin indicates the LSb of TBLPTR, but it is left unconnected. Instead, the $\overline{UB}$ and $\overline{LB}$ signals are active to select both bytes. The obvious limitation to this method is that the Table Write must be done in pairs on a specific word boundary to correctly write a word location.

Figure 1-4 shows a typical implementation of this mode.

FIGURE 1-4: 16-BIT WORD WRITE MODE



1.3 8-bit External Interfaces

1.3.1 8-BIT MULTIPLEXED EXTERNAL INTERFACE

This interface is only available on the PIC18C601. It requires the use of either a lower processor operating frequency as compared to 16-bit modes, or the use of a faster memory device.

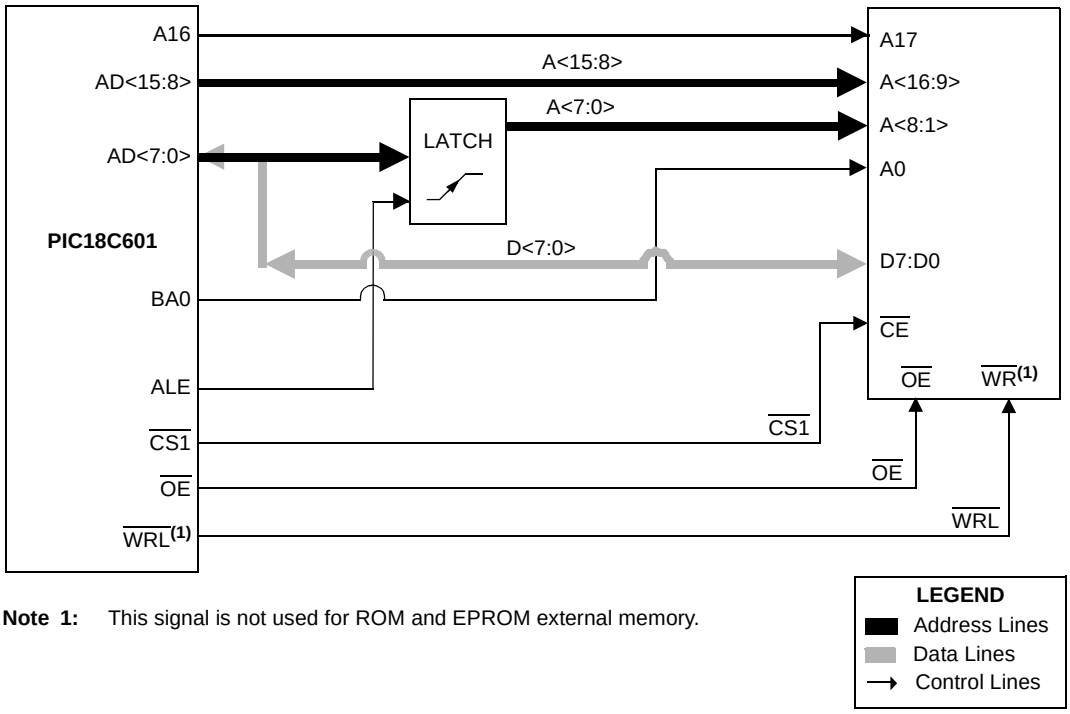
In this mode, the low order address and data bytes are multiplexed, and require a single latch to de-multiplex the address and data busses. Instructions are fetched as two 8-bit bytes within one instruction cycle. Pin BA0 from the controller must be connected to address pin A0 of the memory device(s); because of this, controller address pins A16:A0 are connected to memory address pins A17:A1. The output enable ($\overline{OE}$) signal will enable the first byte of program memory for a portion of the cycle, the second byte will be read to from the 16-bit instruction word when BA0 changes.

When the 8-bit interface is selected, the $\overline{WRH}$, $\overline{UB}$ and $\overline{LB}$ pins are not used; they revert to I/O port functions. The $\overline{WRL}$ signal is active on every external write.

The external address is 18-bits wide, which allows for addressing of up to 256 Kbytes. External Table Reads and Table Write are performed one byte at a time.

Figure 1-5 shows a typical implementation of this mode. The control signals are described in Table 1-4.

FIGURE 1-5: 8-BIT MULTIPLEXED MODE FOR PIC18C601



Note 1: This signal is not used for ROM and EPROM external memory.

TABLE 1-4: 8-BIT MULTIPLEXED MODE CONTROL SIGNALS

Name	8-bit Mux mode	Function
RG0/ALE	ALE	Address Latch Enable (ALE) Control pin
RG1/$\overline{OE}$	$\overline{OE}$	Output Enable ($\overline{OE}$) Control pin
RG2/$\overline{WRL}$	$\overline{WRL}$	Write Low ($\overline{WRL}$) Control pin
RG4/BA0	BA0	Byte Address bit 0
RF3/$\overline{CSIO}$	$\overline{CSIO}$	Chip Select I/O (see Section 3.3)
RF5/$\overline{CS1}$	$\overline{CS1}$	Chip Select 1 (see Section 3.1)

1.3.2 8-BIT WITH DE-MULTIPLEXED EXTERNAL INTERFACE

This interface is only available on the PIC18C801. It requires the use of either a lower processor operating frequency as compared to 16-bit modes, or the use of a faster memory device.

The address and data busses are separate and do not require any external latches for de-multiplexing. The instructions are fetched as two 8-bit bytes on a dedicated data bus (PORTJ); the address is presented for the entire duration of the fetch cycle on a separate address bus. The two bytes are fetched during one instruction cycle.

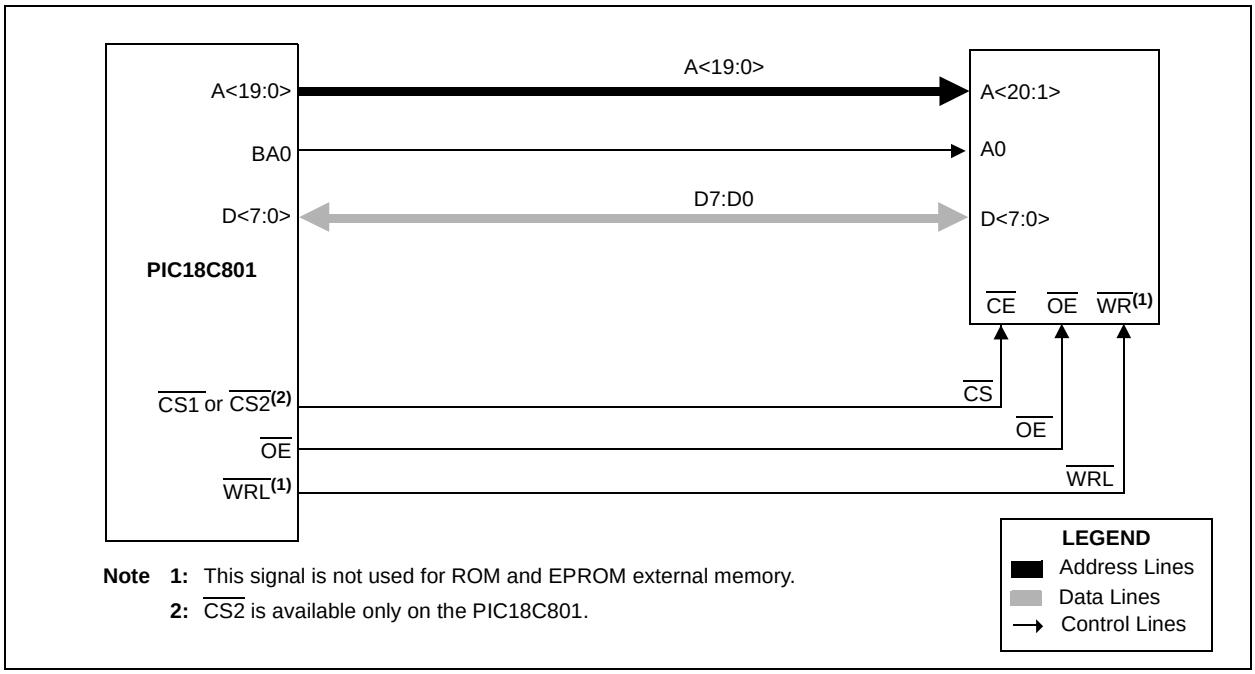
Pin BA0 from the controller must be connected to address pin A0 of the memory device(s); because of this, controller address pins A19:A0 are connected to memory address pins A20:A1. The output enable ($\overline{OE}$) signal will enable the first byte of program memory for a portion of the cycle; the second byte will be read to from the 16-bit instruction word when BA0 changes.

The external address is 21-bits wide, which allows for addressing of up to 2 Mbytes. External Table Reads and Table Writes are performed one byte at a time.

When the 8-bit de-multiplexed interface is selected, the $\overline{WRH}$, $\overline{UB}$ and $\overline{LB}$ pins are not used; they revert to I/O port functions. The $\overline{WRL}$ signal is active on every external write.

The control signals for this interface are described in Table 1-5.

FIGURE 1-6: 8-BIT DE-MULTIPLEXED MODE FOR PIC18C801



Note 1: This signal is not used for ROM and EPROM external memory.

2: $\overline{CS2}$ is available only on the PIC18C801.

TABLE 1-5: 8-BIT DE-MULTIPLEXED MODE CONTROL SIGNALS

Name	8-bit De-Mux Mode	Function
RG0/ALE	ALE	Address Latch Enable (ALE) Control pin
RG1/$\overline{OE}$	$\overline{OE}$	Output Enable ($\overline{OE}$) Control pin
RG2/$\overline{WRL}$	$\overline{WRL}$	Write Low ($\overline{WRL}$) Control pin
RG4/BA0	BA0	Byte Address bit 0
RF3/$\overline{CSIO}$	$\overline{CSIO}$	Chip Select I/O (see Section 3.3)
RF4/$\overline{CS2}$	$\overline{CS2}$	Chip Select 2 (see Section 3.2)
RF5/$\overline{CS1}$	$\overline{CS1}$	Chip Select 1 (see Section 3.1)

2.0 MEMORY MAPPED I/O

In general, ROMless microcontrollers have less dedicated I/O ports available than their ROM equipped counterparts. To get around this limitation, additional I/O channels are made available through memory mapped communications with peripheral devices. Normally, this is achieved in one of two ways:

- Using discrete digital logic
- Using programmable peripherals

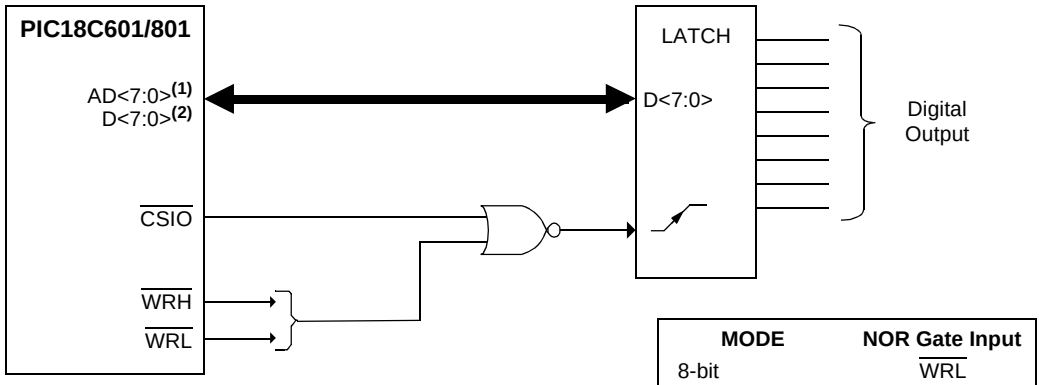
2.1 Discrete Digital Logic

In general, latches are required for output ports, while tri-state buffers are used for input ports.

Figure 2-1 demonstrates the requirements for a typical output port. Normally, latches have one active high control signal, with data being latched at the signal's high-to-low transition. The controller data bus (D<7:0>) is connected to the data input bus of the latch. The $\overline{CSIO}$ and appropriate $\overline{WR}$ control lines are NORed to produce the latch control signal.

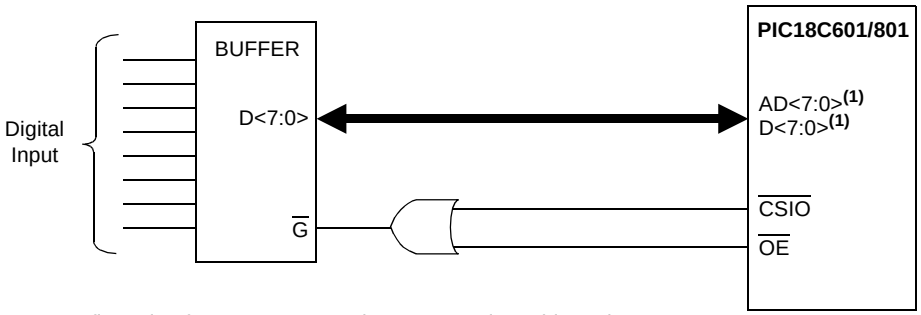
Figure 2-2 demonstrates the requirements for a digital input interface. Tri-state buffers usually have one active low control signal; when it is active, input data is transferred to the buffer output. The controller data bus (D<7:0>) is connected to the output bus of the buffer. The $\overline{CSIO}$ and $\overline{OE}$ lines are ORed to produce the buffer control signal.

FIGURE 2-1: OUTPUT INTERFACE USING DISCRETE DIGITAL LOGIC



Note 1: Configuration for PIC18C601 and PIC18C801 in 16-bit mode.
2: Configuration for PIC18C801 in 8-bit mode..

FIGURE 2-2: INPUT INTERFACE USING DISCRETE DIGITAL LOGIC



Note 1: Configuration for PIC18C601 and PIC18C801 in 16-bit mode.
2: Configuration for PIC18C801 in 8-bit mode..

2.2 Programmable Peripherals

Some commonly used peripherals, such as the Intel® compatible 8255 (programmable peripheral interface) and the 8279 (programmable keyboard/display interface), have 8-bit interfaces. These devices can be connected to ROMless microcontrollers for memory mapped I/O operation, using $\overline{CSIO}$ as a control line.

Figures 2-3 through 2-5 demonstrate methods for interfacing programmable peripheral devices with ROMless microcontrollers. Some peripherals (such as the 8255 and 8279) have one or two address lines to select internal registers. As these are 8-bit devices, special care is required for the 16-bit modes. The addressing scheme is selected in such a way that it is common for all modes. The base address will be an even address specified by the CSELIO register. (See Section 3.3 for details on specifying the location of the 8 Kbyte region for I/O).

The peripheral device's data bus (D7:D0) is connected to the controller's data bus. Peripheral address pins A0 and A1 (in some cases, only A0) are connected to A0 and A1 of the controller. For 8-bit mode, address pins A0 and A1 of the peripheral device (in some case only A0) can be connected with BA0 and A0 of the controller.

The $\overline{RD}$ control pin of the peripheral is connected to the $\overline{OE}$ pin of the controller. The $\overline{WR}$ pin of the peripheral is connected to either the $\overline{WRL}$ or $\overline{WRH}$ pin of the controller, depending on the memory interface mode.

The internal registers of the peripheral device can be accessed by Table Read and Table Write instructions. Addresses for the registers start with the base address specified by the CSELIO register, incrementing with an offset of 02h in 16-bit mode or 01h in 8-bit mode. (See the Address/Register tables in Figures 2-3 through 2-5 for details.) For 16-bit Table Write operations, the upper byte will be dummy data.

FIGURE 2-3: 16-BIT MEMORY MAPPED I/O FOR THE PIC18C601/801 – PROGRAMMABLE PERIPHERAL DEVICES

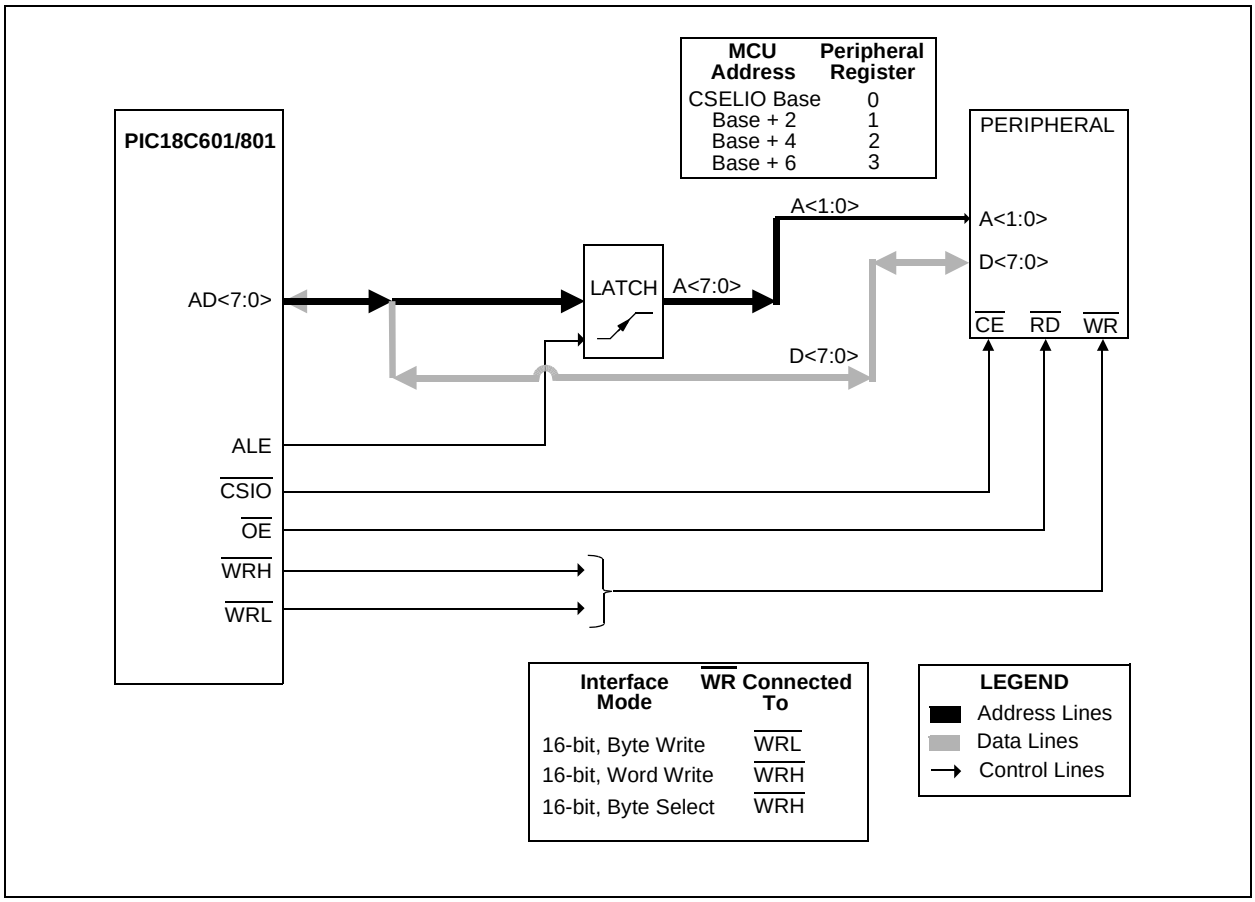


FIGURE 2-4: 8-BIT MEMORY MAPPED I/O FOR THE PIC18C801 – PROGRAMMABLE PERIPHERAL DEVICES

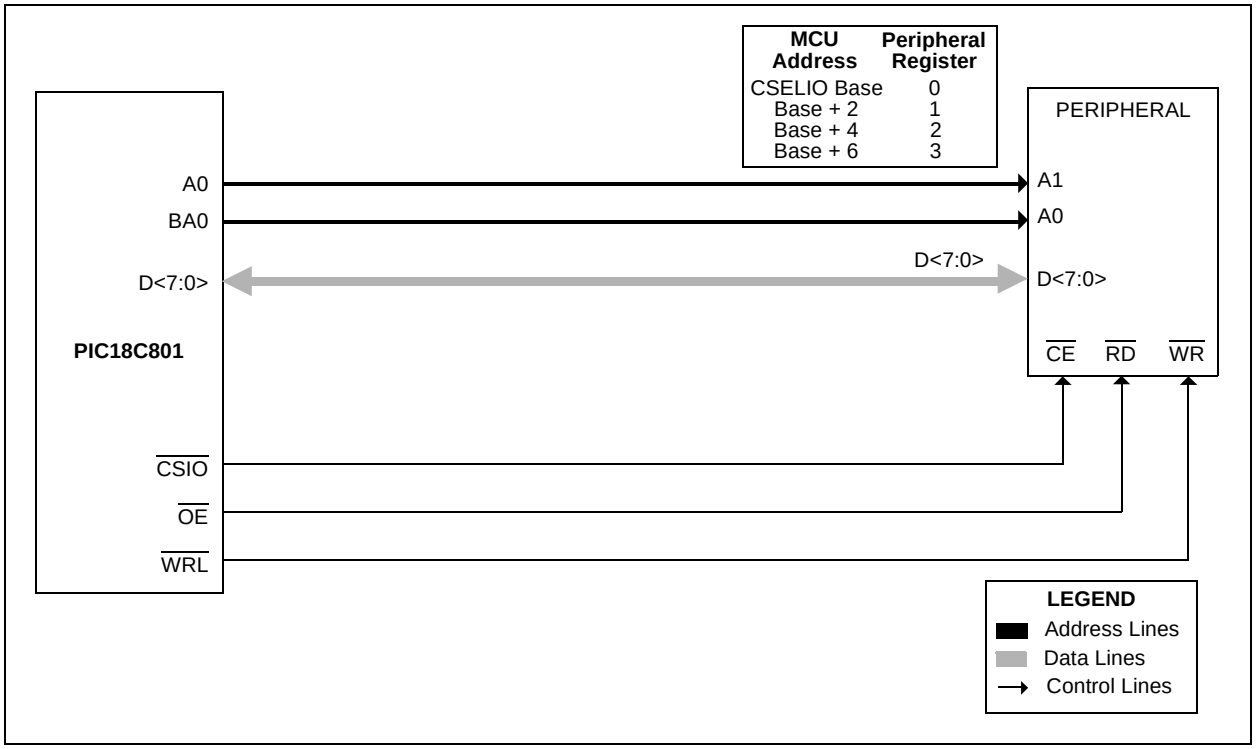
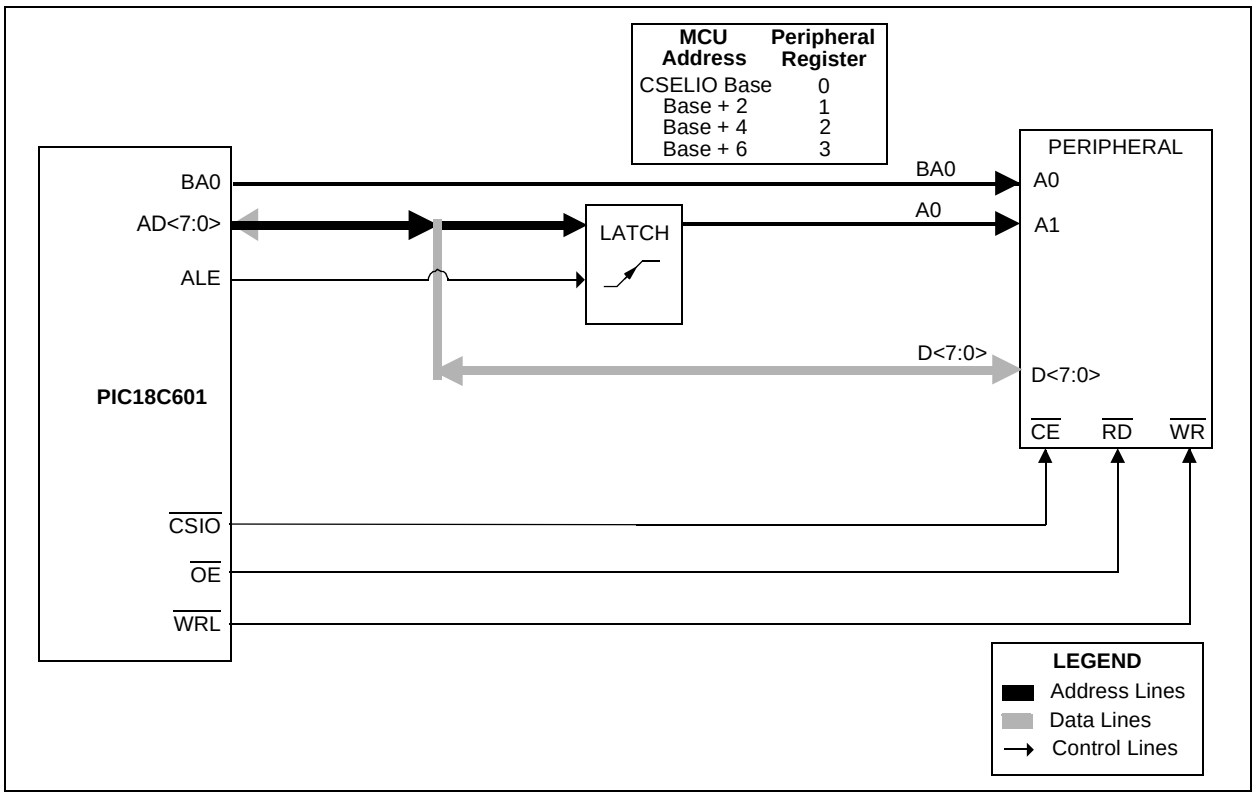


FIGURE 2-5: 8-BIT MEMORY MAPPED I/O FOR THE PIC18C601 – PROGRAMMABLE PERIPHERAL DEVICES



3.0 MEMORY MAPPING

PIC18C601/801 microcontrollers are capable of supporting a wide variety of memories and memory configurations. Users can connect up to two different types of memories in a single system. Different memories can be enabled or disabled using combinations of the chip select signals.

While chip select signals are normally generated by externally decoding the address lines, PIC18C601/801 devices provide up to two programmable on-board chip select signals and one I/O chip select signal. This minimizes the amount of additional external circuitry required to interface the external memory and memory mapped I/O devices. The PIC18C601 provides $\overline{CS1}$ and $\overline{CSIO}$, while the PIC18C801 provides $\overline{CS1}$, $\overline{CS2}$ and $\overline{CSIO}$.

When enabled, a chip select signal is asserted whenever the CPU accesses the dedicated range of addresses, specified in the chip select registers, CSEL2 and CSELIO. If both the CSEL2 and CSELIO registers are 00h, all of the chip selects are disabled, and their corresponding pins are configured as I/O. In addition, when program execution takes place from internal Boot RAM, all chip selects are configured in their inactive states. For the PIC18C601, $\overline{CS1}$ is always enabled unless the $\overline{CSIO}$ signal is active. The CSEL2 register in the PIC18C601 is not implemented, and can be used as a general purpose register.

The chip selects are unaffected by any device RESETS, *except* the Power-on Reset. The RESET value of these registers enable all three chip selects with $\overline{CS1}$ active (= '0'), $\overline{CS2}$ inactive (= '1'), and $\overline{CSIO}$ inactive (= '1') (with the RESET address = 000000h).

Details of the operation of the chip selects are provided in the following sections. Figure 3-1 further illustrates the relationships between the chip selects and the program memory map.

3.1 $\overline{CS1}$

The $\overline{CS1}$ signal is active low and inactive high. $\overline{CS1}$ is enabled by writing a value other than 00h into either the CSEL2 register, or the CSELIO register. If both registers are programmed to 00h, the $\overline{CS1}$ signal is not enabled and the RF5/$\overline{CS1}$ pin is configured as I/O.

The $\overline{CS1}$ signal is active for a range of addresses that are specified in the CSEL2 and CSELIO registers. By default, it is active for the entire program memory range from 000000h to 1FDFFh. It will always be active for the lower 8 Kbytes of program memory. $\overline{CS1}$ is active for all addresses for which $\overline{CS2}$ and $\overline{CSIO}$ are inactive. Therefore, if CSEL2 is equal to 20h and CSELIO is equal to 80h, the $\overline{CS1}$ signal will be active for the addresses that fall between 000000h and 03FFFFh.

When the device cycles through a Power-on Reset, the $\overline{CS1}$ chip select is the only active chip select.

Note:	Because it is the only active chip select signal on Power-on Reset, $\overline{CS1}$ must always be connected to one of the external memory devices .

3.2 $\overline{CS2}$

The $\overline{CS2}$ signal is also active low and inactive high. A value of 00h in the Chip Select 2 Register (CSEL2) disables the $\overline{CS2}$ signal and configures the RF4/$\overline{CS2}$ pin as I/O.

The program memory range for $\overline{CS2}$ is determined by the value contained in CSEL2. An 8-bit value in this register determines the starting address for the activation of $\overline{CS2}$. The 8-bit value is decoded as one of 256 boundaries in the program memory space, each being 8 Kbyte in size. For example, if the value contained in the CSEL2 register is 128 (80h), then the $\overline{CS2}$ signal will be asserted whenever the address is greater than, or equal to 8192 x 128, or 1,048,576 (100000h).

When the device cycles through a Power-on Reset, $\overline{CS2}$ becomes inactive.

3.3 $\overline{\text{CSIO}}$

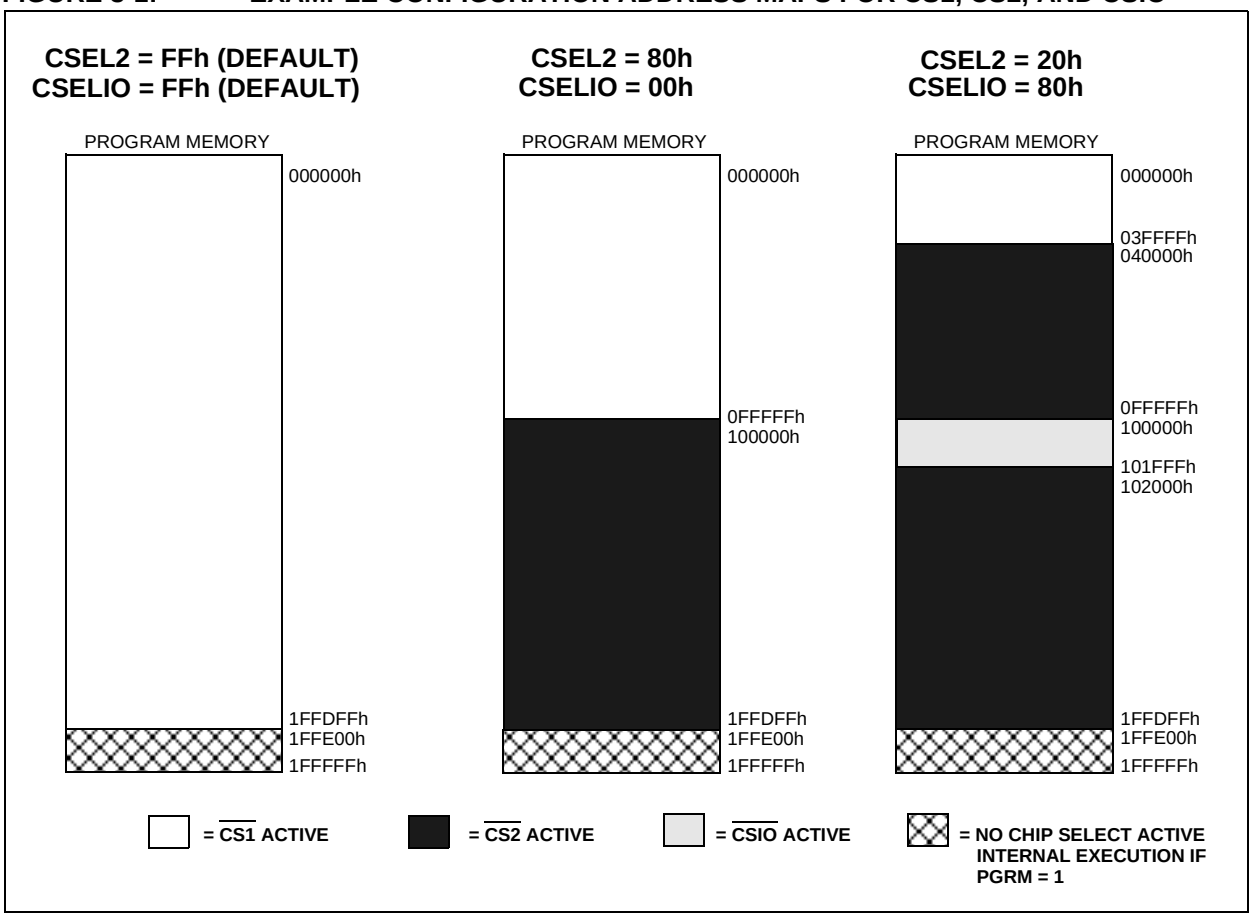
The $\overline{\text{CSIO}}$ signal is also active low and inactive high. A value of 00h in the Chip Select 2 Register (CSEL2) disables the $\overline{\text{CSIO}}$ output, and configures the RF3/$\overline{\text{CSIO}}$ pin as I/O.

The I/O chip select is active for a fixed 8 Kbyte address range. The location of the I/O chip select is determined by the value contained in the I/O Chip Select register (CSELIO). The eight-bit value is decoded as one of 256 8 Kbyte banks of program memory that, when addressed, will assert the $\overline{\text{CSIO}}$ output. If, for instance, the value contained in the $\overline{\text{CSIO}}$ register is 128 (80h), then the $\overline{\text{CSIO}}$ pin will be asserted for the address range between and including 8192 x 128 to ((8192 x 128) + 8192), or 100000h to 101FFFh. If the 8 Kbyte address block overlaps the address range of the $\overline{\text{CS2}}$ signal, the $\overline{\text{CSIO}}$ signal will be active and the $\overline{\text{CS2}}$ signal will be inactive for that 8 Kbyte block of addresses decoded by the CSELIO register.

When the device cycles through a Power-on Reset, $\overline{\text{CSIO}}$ becomes inactive.

Note:	The RESET state of both CSEL2 and CSELIO registers on Power-on Reset is FFh. This allows the $\overline{\text{CS1}}$, $\overline{\text{CS2}}$ and $\overline{\text{CSIO}}$ signals to be enabled. The $\overline{\text{CS1}}$ signal will be active for all addresses less than 1FE000h, the $\overline{\text{CSIO}}$ signal is active for all addresses greater than, or equal to 1FE000h, and $\overline{\text{CS2}}$ is inactive, since it shares the same address value as $\overline{\text{CSIO}}$. This ensures that the chip select signals are not floating if external memory is present, and its chip enable inputs are tied to the chip selects.

FIGURE 3-1: EXAMPLE CONFIGURATION ADDRESS MAPS FOR $\overline{\text{CS1}}$, $\overline{\text{CS2}}$, AND $\overline{\text{CSIO}}$



REGISTER 3-1: CSEL2 REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
CSL7	CSL6	CSL5	CSL4	CSL3	CSL2	CSL1	CSL0
bit 7							bit 0

bit 7-0 **CSL<7:0>**: Chip Select 2 Address Decode bits

XXh = All eight bits are compared to the Most Significant bits PC<20:13> of the program counter. If PC<20:13> ≥ CSL<7:0> register, then the $\overline{CS2}$ signal is low.
If PC<20:13> < CSL<7:0>, $\overline{CS2}$ is high.

00h = $\overline{CS2}$ is inactive

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
- n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

REGISTER 3-2: CSELIO REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
CSIO7	CSIO6	CSIO5	CSIO4	CSIO3	CSIO2	CSIO1	CSIO0
bit7							bit0

bit 7 **CSIO<7:0>**: Chip Select I/O Address Decode bits

XXh = All eight bits are compared to the Most Significant bits PC<20:13> of the program counter. If PC<20:13> = CSIO<7:0>, then the $\overline{CSIO}$ signal is low. If not, $\overline{CSIO}$ is high.

00h = $\overline{CSIO}$ is inactive

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
- n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

4.0 MEMORY DEVICES AND INTERFACES

The PIC18C601/801 ROMless devices are designed to support a variety of memory devices. A variety of control signals are provided to facilitate the interface with many memory chips.

4.1 Memory Devices with x8 Organization

4.1.1 x8 ARRANGEMENT

In this arrangement, address pins AX:A0 of the controller are connected to address pins AX+1:A1 of the memory device. Pin BA0 is connected to the A0 pin of memory. Controller data pins D7:D0 are connected to memory data lines D7:D0. The controller's $\overline{OE}$ pin is connected to the $\overline{OE}$ pin of the memory device, and the controller's $\overline{WRL}$ pin is connected to the memory's $\overline{WE}$ pin.

For the PIC18C801, all address and data signals are directly available at pin. For the PIC18C601, pins AD7:AD0 are multiplexed; one external latch is required for de-multiplexing the address and data busses.

Instructions are fetched as two 8-bit bytes. The output enable ($\overline{OE}$) signal will enable one byte of program memory for a portion of cycle, then the LSb of address BA0 will change and the second byte will be read to from the 16-bit instruction word.

External Table Reads and Table Writes are performed one byte at a time.

Examples of the 8-bit interfaces are provided in Figure 1-5 (Multiplexed mode) and Figure 1-6 (De-Multiplexed mode).

Note:	The 8-bit memory interfaces require the use of either faster memory devices, or a lower controller operating frequency (as compared to similar 16-bit interfaces).

4.1.2 x16 ARRANGEMENT

For this arrangement, two byte-wide memory chips are used, one each for the LSB and MSB. The controller address lines AX:A0 are connected to the AX:A0 address lines of both memories. BA0 is left unconnected. The controller data lines are connected to D7:D0 of one memory device (LSB), and to D15:D8 of the second device (MSB). The controller $\overline{OE}$ pin is connected with the $\overline{OE}$ pin of both memories. The controller $\overline{WRL}$ pin is connected to $\overline{WE}$ of the LSB memory, while $\overline{WRH}$ is connected to $\overline{WE}$ of the MSB memory.

Instructions are fetched as a 16-bit word. The output enable ($\overline{OE}$) signal will enable one byte of both memories for a portion of cycle to form the 16-bit instruction word.

External Table Reads are logically performed one byte at a time, although the memory will read a 16-bit word externally. The Least Significant bit of the address internally selects between high and low bytes.

For Table Writes, configure the MEMCON register for Byte Write mode (MEMCON<1:0>='00'). During a TBLWT cycle, the TABLAT data is presented on the upper and lower byte of the AD bus. The appropriate $\overline{WRH}$ or $\overline{WRL}$ signal is strobed based on the LSb of the TBLPTR.

An example of this interface is provided in Figure 1-1.

4.2 Memory Devices with x16 Organization

For this arrangement, the controller address pins AX:A0 are connected to the memory address pins AX:A0. BA0 is not used, and left unconnected. The controller data pins D15:D0 are connected to D15:D0 of the memory device. The controller's $\overline{OE}$ control pin is connected with the $\overline{OE}$ pin of the memory, and the controller's $\overline{WRH}$ pin is connected to the memory's $\overline{WE}$ pin.

Instructions are fetched as a single 16-bit word. The output enable ($\overline{OE}$) signal will enable word from memory for a portion of the cycle to get the 16-bit instruction word.

External Table Reads are logically performed one byte at a time, although the memory will read a 16-bit word externally. The Least Significant bit of the address will internally select between high and low bytes.

For Table Write, configure the MEMCON register for Word Write mode (MEMCON<1:0>='1x'). During a TBLWT cycle to an even address (TBLPTR<0> = '0'), the TABLAT data is transferred to a holding latch and the external address data bus is tri-stated for the data portion of the bus cycle. No write signals are activated. During a TBLWT cycle to an odd address (TBLPTR<0> = '1'), the TABLAT data is presented on the upper byte of the AD bus (AD15:AD8). The contents of the holding latch are presented on the lower byte of the AD bus (AD7:AD0). The $\overline{WRH}$ signal is strobed for each write cycle and the $\overline{WRL}$ signal is unused. The BA0 signal indicates the LSb of TBLPTR, but it is not used. The $\overline{UB}$ and $\overline{LB}$ control signals are active to select both bytes.

An example of the 16-bit Word Write interface is shown in Figure 1-4.

4.3 Memory Devices with x8/x16 Selectable Organization

The memory devices covered in this section are capable of supporting both x8 and x16 organization. The method used for interfacing a device depends on the organization used. Additionally, the addressing modes discussed here differ on how the controller address lines are used to fetch byte or word entities. To distinguish these, the schemes in this section will be referred to as “Basic Byte” and “Basic Word”.

4.3.1 “BASIC BYTE” ADDRESSING SCHEMES

In this scheme, all available memory address lines are used to directly address byte-size data entities.

In general, TSOP56 package JEDEC FLASH devices fall into the “Basic Byte” category.

4.3.1.1 Byte (8-bit) Mode

The arrangement is similar to the 8-bit modes illustrated in Figures 1-5 and 1-6. Address pins AX:A0 for the controller are connected to address pins AX+1:A1 of memory. The controller’s BA0 pin is connected to A0 of the memory device. For the PIC18C801, data pins D7:D0 are connected directly to the data pins of the memory, providing a “glueless” interface (Figure 4-2). For the PIC18C601, the low order address and data signals are multiplexed; controller pins AD7:AD0 are directly connected to the memory to provide data, while an external latch de-multiplexes address pins A7:A0 (Figure 4-1). The controller and memory $\overline{OE}$ pins are connected, while the controller $\overline{WRL}$ pin is connected to the memory $\overline{WR}$ pin.

Instructions are fetched as two 8-bit bytes. The output enable ($\overline{OE}$) signal will enable one byte of program memory for the first portion of cycle; then the BA0 signal changes, and the second byte is read to form the 16-bit instruction word. External Table Reads and Table Writes are performed one byte at a time.

4.3.1.2 Word (16-bit) Mode

For this arrangement, the controller’s AX:A0 address pins are connected to address pins AX+1:A1 of memory. Controller pin BA0 and memory pin A0 are left unconnected. The controller’s D15:D0 data pins are connected to the memory’s D15:D0 data pins. The controller and memory $\overline{OE}$ pins are connected, while the controller’s $\overline{WRH}$ pin is connected to the $\overline{WE}$ pin of the memory. The connections for this mode are shown in Figure 4-3.

For Byte Select mode, the controller’s BA0 pin must be connected to A0 of the memory. The WORD/$\overline{BYTE}$ pin of the memory is connected to an I/O pin on the controller to enable dynamic Word/Byte mode switching. Dynamic switching of FLASH devices is not possible if a program is executing from the same FLASH at the same time. See Section 1.2.3 for more information.

Instructions are fetched as 16-bit words. The output enable ($\overline{OE}$) signal will enable word from memory for a portion of cycle to fetch the 16-bit instruction word. External Table Reads and Writes are logically performed one byte at a time, although the memory will read a 16-bit word externally. The Least Significant bit of the address will internally select between high and low bytes. Table Writes can be performed in either Word Write or Byte Select mode. (Refer to Sections 1.2.3 and 1.2.4 for more details.)

FIGURE 4-1: 8-BIT “BASIC BYTE” ADDRESSING SCHEME FOR THE PIC18C601

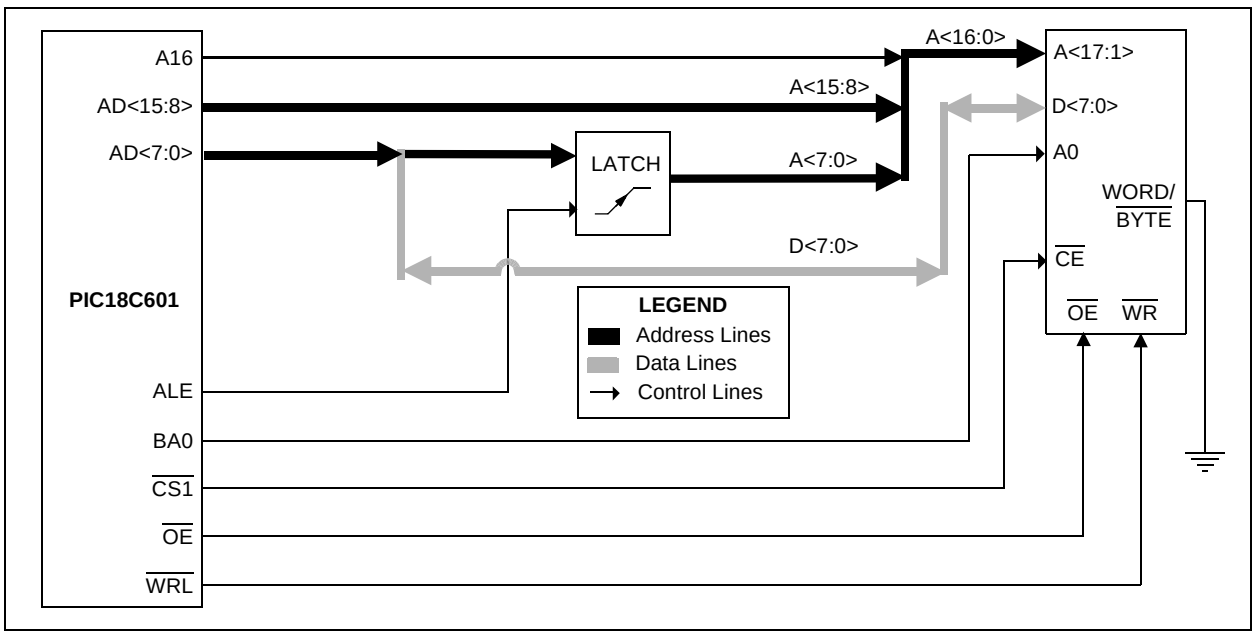
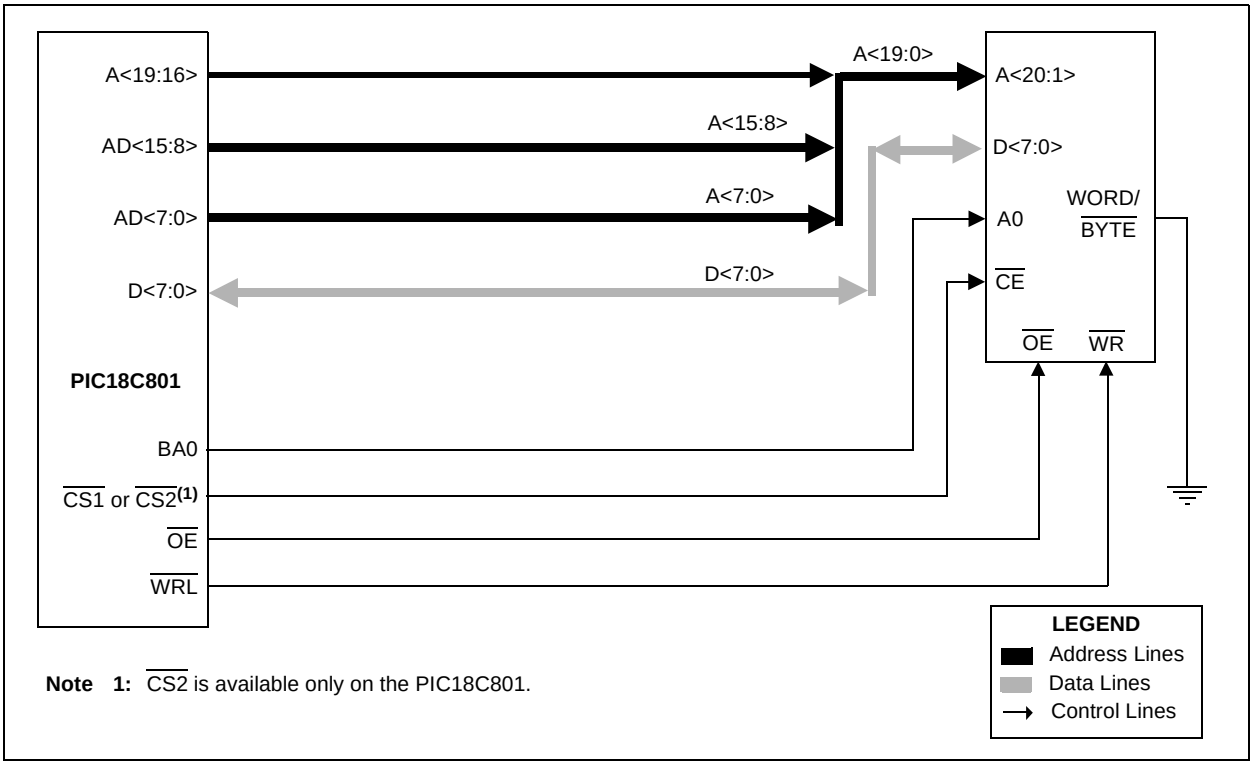
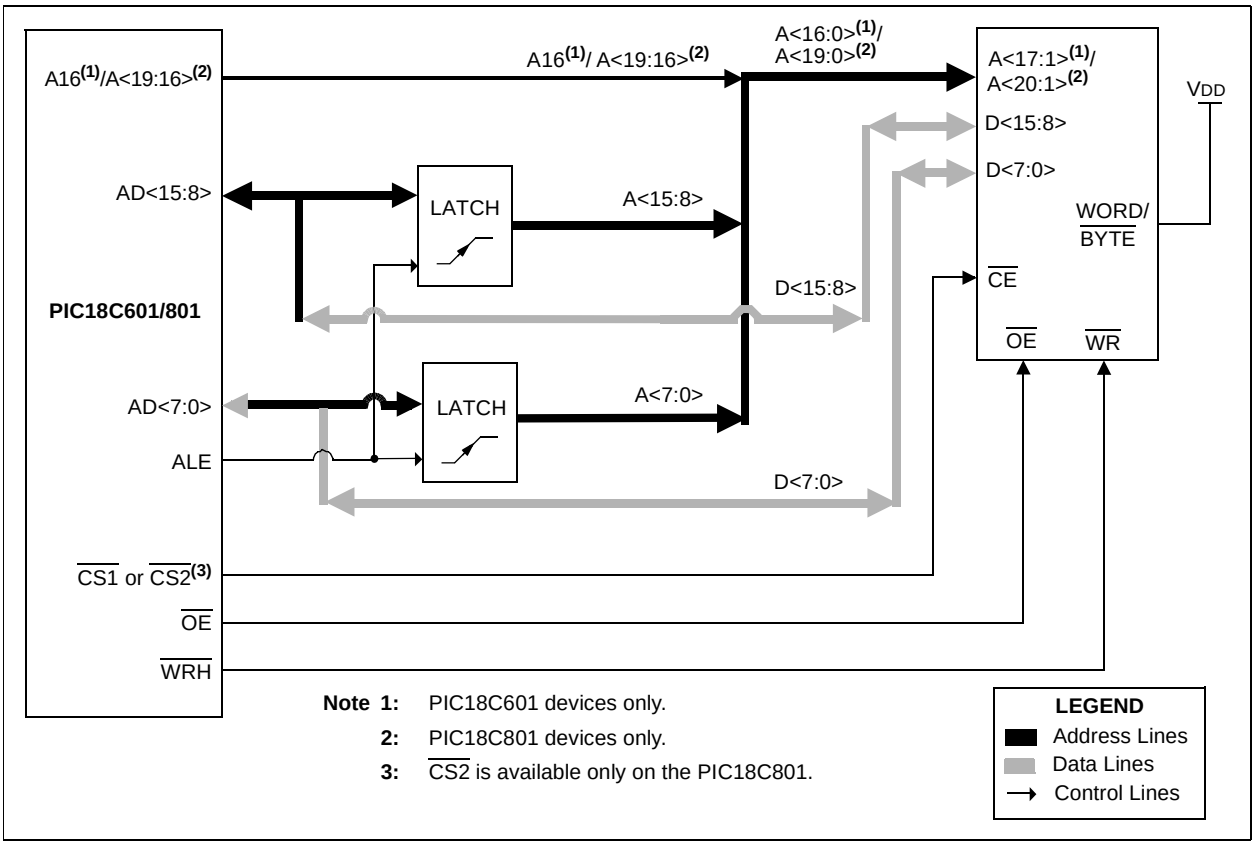


FIGURE 4-2: 8-BIT "BASIC BYTE" ADDRESSING SCHEME FOR THE PIC18C801



Note 1: CS2 is available only on the PIC18C801.

FIGURE 4-3: 16-BIT "BASIC BYTE" ADDRESSING SCHEME FOR THE PIC18C601/801



Note 1: PIC18C601 devices only.
2: PIC18C801 devices only.
3: CS2 is available only on the PIC18C801.

4.3.2 "BASIC WORD" ADDRESSING SCHEMES

In this scheme, all available memory address lines are used to address word (16-bit) data entities. Functionally, one pin (e.g. DQ15) is used as the LSb address input, and acts as a multi-function pin, depending on the operation mode (Byte or Word).

In general, TSOP48 package FLASH devices fall into the "Basic Word" category.

4.3.2.1 Byte (8-bit) Mode

In this arrangement, controller address pins AX:A0 are connected to memory address lines AX:A0. Controller data pins D<7:0> are connected to memory data pins D7:D0. BA0 is connected to a memory multi-function pin (e.g., DQ15). The controller and memory $\overline{OE}$ pins are connected, while the controller $\overline{WRL}$ pin is connected to the memory's $\overline{WE}$ pin.

For the PIC18C601, pins AD7:AD0 are multiplexed, so one external latch is required for de-multiplexing the address and data busses (Figure 4-4). For the PIC18C801, data is directly available at pin (Figure 4-5).

Instructions are fetched as two 8-bit bytes. The output enable ($\overline{OE}$) signal will enable one byte of program memory for a portion of cycle; then the BA0 signal will change state for the LSb of address, and the second byte will be read to from the 16-bit instruction word. External Table Reads and Table Writes are performed one byte at a time.

4.3.2.2 Word (16-bit) Mode

For this arrangement, controller address pins AX:A0 are connected to memory address pins AX:0. BA0 is left unconnected. The controller data pins D15:D0 pins are connected to memory data pins D15:D0. The controller and memory $\overline{OE}$ pins are connected, while controller pin $\overline{WRH}$ is connected to the memory's $\overline{WE}$ pin. The connections for this mode are shown in Figure 4-6.

Instructions are fetched as one 16-bit word. The output enable ($\overline{OE}$) signal will enable word from memory for a portion of the cycle to get the 16-bit instruction word. External Table Reads and Writes are logically performed one byte at a time, although the memory will read a 16-bit word externally. The Least Significant bit of the address will internally select between high and low bytes.

For Byte Select mode, the controller's BA0 pin must be connected to pin D15/A1 of the memory. The WORD/$\overline{BYTE}$ pin of the memory is connected to an I/O pin on the controller to enable dynamic Word/Byte mode switching. Dynamic switching of FLASH devices is not possible if a program is executing from the same FLASH at the same time. See Section 1.2.3 for more information.

Table Writes can be performed in either Word Write or Byte Select mode. (Refer to Sections 1.2.3 and 1.2.4 for more details.)

FIGURE 4-4: 8-BIT "BASIC WORD" ADDRESSING SCHEME FOR THE PIC18C601

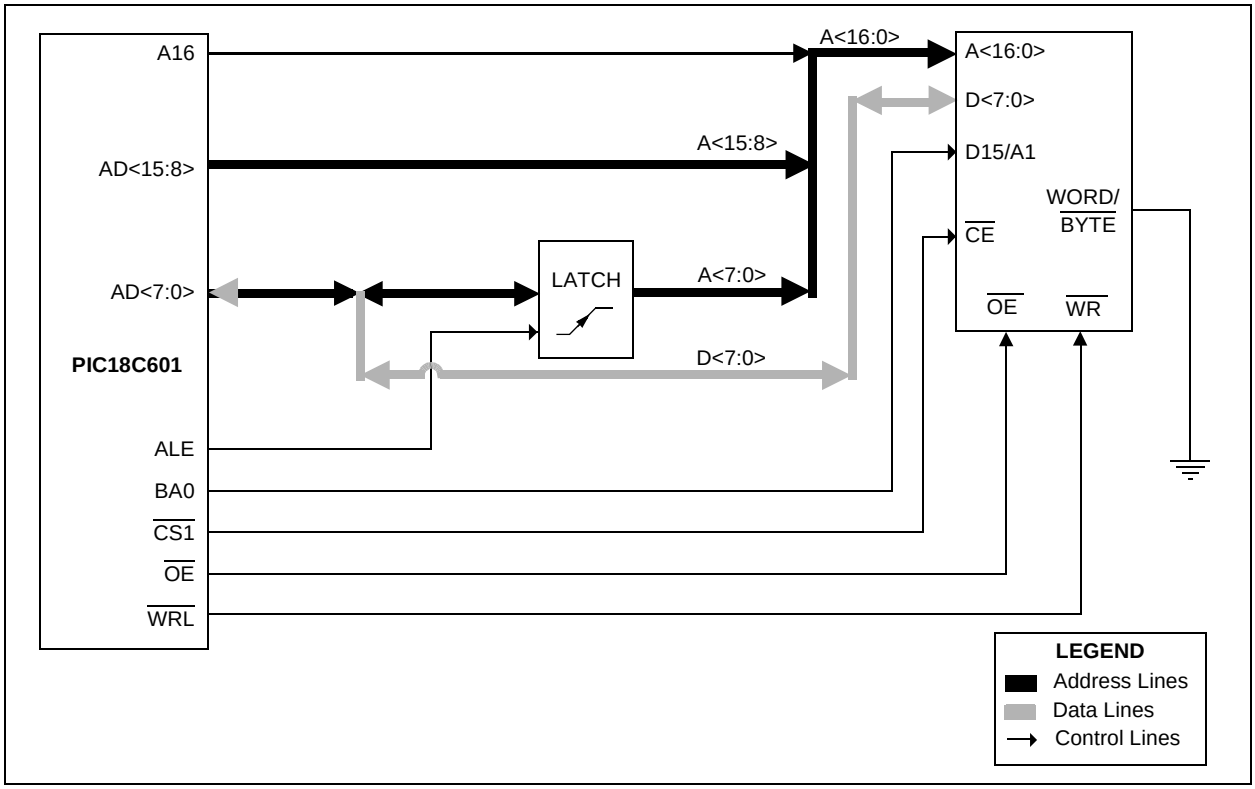


FIGURE 4-5: 8-BIT "BASIC WORD" ADDRESSING SCHEME FOR THE PIC18C801

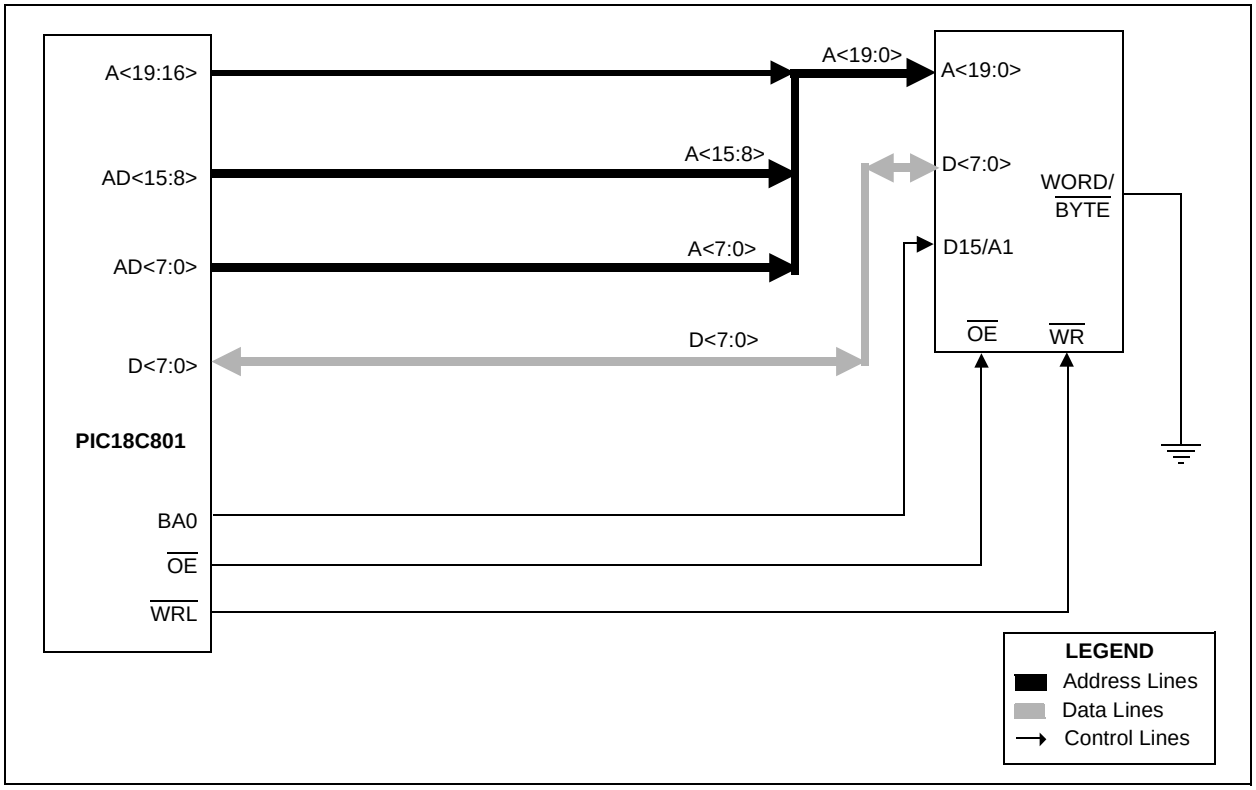
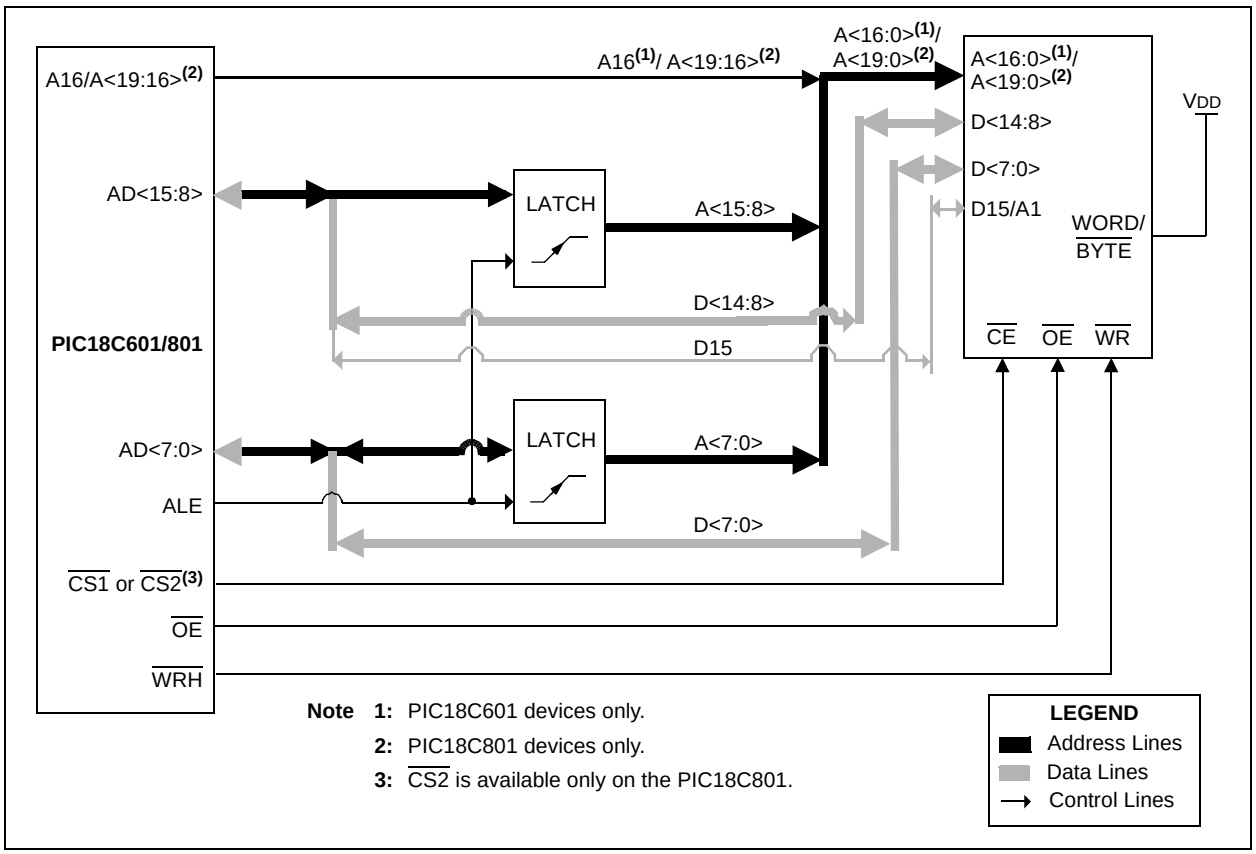


FIGURE 4-6: 16-BIT "BASIC WORD" ADDRESSING SCHEME FOR THE PIC18C601/801



Note 1: PIC18C601 devices only.
2: PIC18C801 devices only.
3: CS2 is available only on the PIC18C801.

5.0 MEMORY TIMING CONSIDERATIONS

Instruction fetching depends on the processor speed, so memory used must provide data output at that speed. This requires consideration on selection of memory for required access time. The memory access time is dependent on main oscillator frequency and External Interface mode. The 8-bit mode requires either a faster memory or a lower processor operating frequency than is available in 16-bit mode.

Most of the program memory timing characteristics are defined relative to the instruction cycle period (TCY). This period equals four times the input oscillator time-base period (TOSC), or one-fourth of the oscillator frequency (FOSC):

$$T_{CY} = 4/F_{OSC} = 4 * T_{OSC}$$

For a data read operation, it is apparent that address access time for the memory (address to output delay, or t_{ACC}) must be less than the controller's T_{ACC} (address valid to data valid delay). However, it is also necessary to consider the propagation delay of the external latch, or tPROP. Depending on the choice of TTL technology, tPROP varies from 5 ns to 40 ns. For 16-bit modes, the memory access time may be expressed by the relationship:

$$t_{ACC} < T_{ACC} - t_{PROP}$$

For 8-bit mode, the BA0 signal toggles within the $\overline{OE}$ period to fetch word instruction. Therefore, memory devices for 8-bit mode systems should supply data in half the time of 16-bit mode systems. The memory access time for 8-bit mode is expressed by the relationship:

$$t_{ACC} < (T_{ACC} - t_{PROP}) / 2$$

Another consideration is the memory device's t_{OE}, or $\overline{OE}$ to output delay. This must be less than the microcontroller's T_{OE} (time from $\overline{OE}$ falling edge to data valid) of 0.5TCY-25 ns. The $\overline{OE}$ pulse width is fixed, so if t_{OE} is longer than T_{OE}, it will not meet the requirement for the minimum data setup time of 20 ns.

In mathematical terms, this means:

$$t_{OE} < T_{OE}, \textit{ or } t_{OE} < 0.5T_{CY}\text{-}25 \text{ ns}$$

After establishing that the controller can successfully access the memory, it's still necessary to compare the memory device's data hold (t_{OH}) and float (t_{DF}) specifications with the controller's ToeH2adD specification (time from $\overline{OE}$ low-to-high transition to AD driven). Although access time is the main criteria for determining device compatibility, the data float time (also occasionally referred to as the *un-access time*) can't be ignored. If the memory device output is unable to go to float in this interval, a bus contention situation will result, in which the next cycle address driven by the CPU will collide with the remnants of the previous cycle memory output. So mathematically,

$$t_{DF} < \text{ToeH2adD}$$

Finally, the external latch used for de-multiplexing the address/data busses must satisfy some of its own timing requirements:

- It must be able to operate within the ALE pulse width interval of 0.25TCY;
- The address must set in the latch within the latch address setup time. This is defined as the interval from address out to the ALE trailing edge, or 0.25TCY-10 ns;
- The address should latch when the ALE signal goes low; and
- Any additional address hold time following the ALE trailing edge should be less than the time from the trailing edge to the address out invalid time of 5 ns.

Table 5-1 lists approximate memory timing requirements for some typical oscillator frequencies. Normally, if access time requirements are met, all other requirements will match.

Detailed timing diagrams and requirements follow on the next two pages. Figure 5-1 and Table 5-2 provide information on 16-bit timing operations, while Figure 5-2 and Table 5-3 provide the same information for 8-bit operations.

TABLE 5-1: SELECTED MEMORY TIMING REQUIREMENTS AT VARIOUS OSCILLATOR FREQUENCIES

Oscillator Frequency	t_{ACC}* (16-bit mode)	t_{ACC}* (8-bit mode)	t_{OE}	t_{DF}	TadV2alL Address Setup Time
4 MHz	<715 ns	<360 ns	<475 ns	<120 ns	<240 ns
10 MHz	<265 ns	<130 ns	<175 ns	<45 ns	<90 ns
16 MHz	<150 ns	<75 ns	<100 ns	<25 ns	<50 ns
20 MHz	<115 ns	<60 ns	<75 ns	<20 ns	<40 ns
25 MHz	<85 ns	<40 ns	<55 ns	<15 ns	<30 ns

* Propagation delay t_{PROP} is assumed to be 10 ns.

FIGURE 5-1: PROGRAM MEMORY FETCH TIMING DIAGRAM (16-BIT)

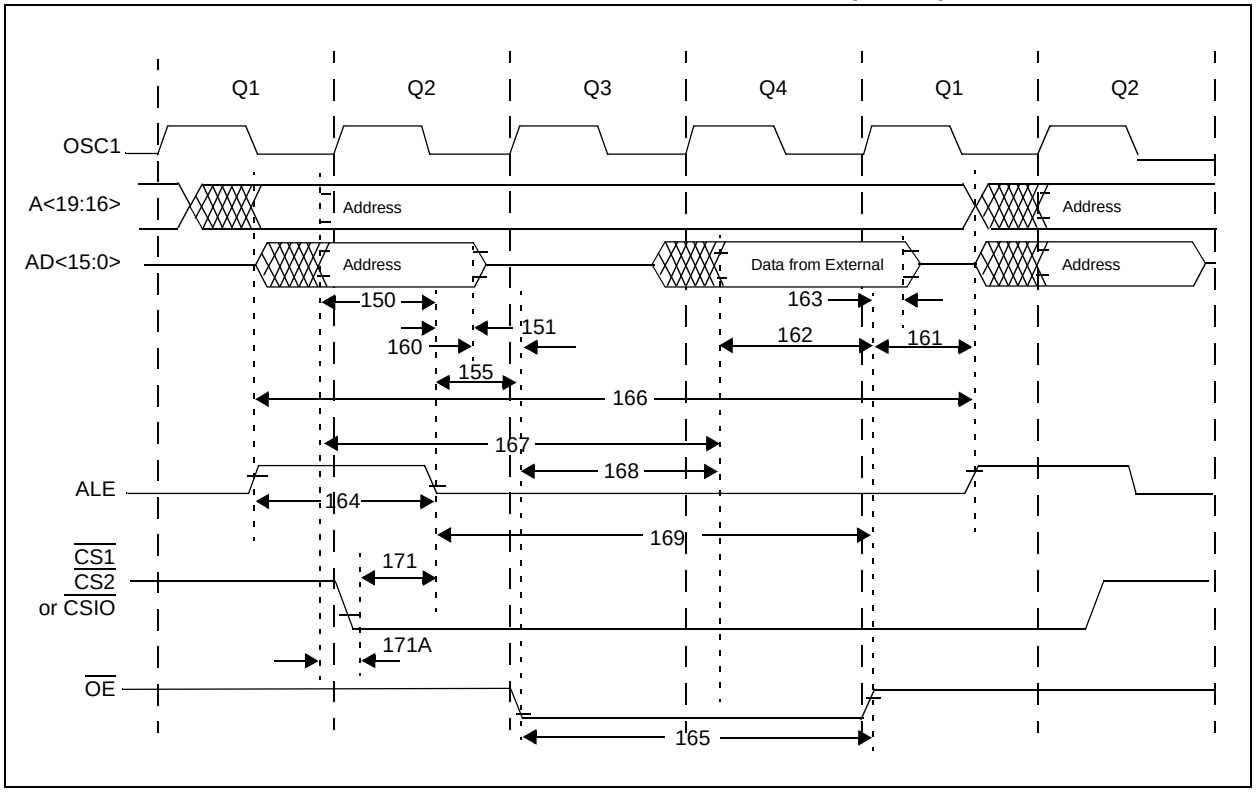


TABLE 5-2: CLKOUT AND I/O TIMING REQUIREMENTS (16-BIT)

Param No	Symbol	Characteristics	Min	Typ	Max	Units
150	TadV2alL	Address out valid to ALE↓ (address setup time)	0.25TCY-10	—	—	ns
151	TalL2adl	ALE↓ to address out invalid (address hold time)	5	—	—	ns
155	TalL2oeL	ALE ↓ to $\overline{OE}$ ↓	10	0.125TCY	—	ns
160	TadZ2oeL	AD high-Z to $\overline{OE}$ ↓ (bus release to $\overline{OE}$)	0	—	—	ns
161	ToeH2adD	$\overline{OE}$ ↑ to AD driven	0.125TCY-5	—	—	ns
162	TadV2oeH	LS Data valid before $\overline{OE}$ ↑ (data setup time)	20	—	—	ns
163	ToeH2adl	$\overline{OE}$ ↑ to data in invalid (data hold time)	0	—	—	ns
164	TalH2alL	ALE pulse width	—	0.25TCY	—	ns
165	ToeL2oeH	$\overline{OE}$ pulse width	0.5TCY-5	0.5TCY	—	ns
166	TalH2alH	ALE↑ to ALE↑ (cycle time)	—	TCY	—	ns
167	T_{ACC}	Address valid to data valid	0.75TCY-25	—	—	ns
168	T_{OE}	$\overline{OE}$ ↓ to data valid	—	—	0.5TCY-25	ns
169	TalL2oeH	ALE ↓ to $\overline{OE}$ ↑	0.625TCY-10	—	0.625TCY+10	ns
171	TalH2csL	Chip select active to ALE ↓	0.25TCY-20	—	—	ns
171A	TubL2oeH	AD valid to chip select active	—	—	10	ns

FIGURE 5-2: PROGRAM MEMORY FETCH TIMING DIAGRAM (8-BIT)

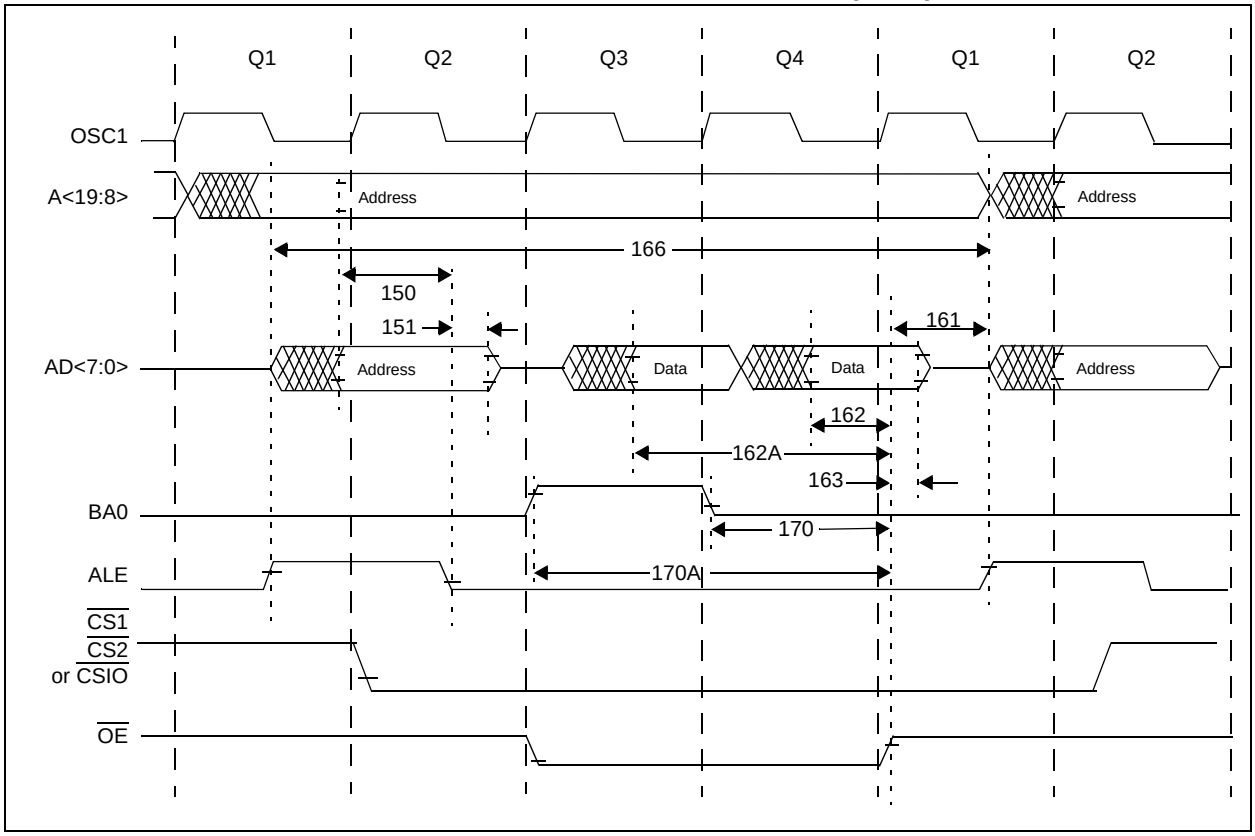


TABLE 5-3: PROGRAM MEMORY FETCH TIMING REQUIREMENTS (8-BIT)

Param No	Symbol	Characteristics	Min	Typ	Max	Units
150	TadV2alL	Address out valid to ALE ↓ (address setup time)	0.25TCY-10	—	—	ns
151	TalL2adl	ALE ↓ to address out invalid (address hold time)	5	—	—	ns
161	ToeH2adD	$\overline{OE}$ ↑ to AD driven	0.125TCY-5	—	—	ns
162	TadV2oeH	LS Data valid before $\overline{OE}$ ↑ (data setup time)	20	—	—	ns
162A	TadV2oeH	MS Data valid before $\overline{OE}$ ↑ (data setup time)	0.25TCY+20	—	—	ns
163	ToeH2adl	$\overline{OE}$ ↑ to data in invalid (data hold time)	0	—	—	ns
166	TalH2alH	ALE ↑ to ALE ↑ (cycle time)	—	TCY	—	ns
170	TubH2oeH	BA0 = 0 valid before $\overline{OE}$ ↑	0.25TCY-10	—	—	ns
170A	TubL2oeH	BA0 = 1 valid before $\overline{OE}$ ↑	0.5TCY-10	—	—	ns

SUMMARY

The PIC18C601/801 family of devices provide a flexible external memory interface to support a variety of embedded systems requirements for cost effective implementation with few external components. A multitude of control signals and interface modes make it possible to use a variety of memory devices with ease. The same interface also provides the ability to use memory mapped I/O with a number of peripheral devices.

APPENDIX A: SUMMARY OF MEMORY DEVICES(1)

Manufacturer	Part ID	Programming Algorithm Family(2)	Organization	Basic Byte/Word Addressing(4)	Remarks
AMD	29F series	A	x8		
	29F series	A	x16		
	29F series	A	x8/x16	Byte	
ATMEL	29 series	B(3)	x8		Sector Programming
	29 series	B(3)	x16		Sector Programming
	49 series	B	x8		
	49 series	B	x16		
	49 series	B	x8/x16	Byte	
INTEL	Boot Block	C	x8/x16	Byte	
	Strata FLASH/ FLASH File	C	x8		
	Strata FLASH/ FLASH File	C	x8/x16	Word	
SHARP	28F series	C	x8		
	28F series	C	x16		
	28F series	C	x8/x16	Word	
ST	29F series	A	x8		
	29F series	A	x16		
	29F series	A	x8/x16	Byte	
Samsung	FLASH products in this family have multiplexed address/data/command lines and are incompatible with PIC18C601/801 devices.				
Catalyst	Boot Block FLASH	C	x8		
	Bulk Erase FLASH	(5)	x8 x16		
Hyundai	29F series	A	x8	x	
	29F series	A	x8/x16	Byte	
Micron	Boot Block	C	x8		
	Boot Block	C	x8/x16	Byte	
	Even Sectored	C	x8		
	Even Sectored	C	x8/x16	Word	
SST	39F series	B	x8		
	29EE series	B(3)	x8		Sector Programming
NexFlash	29F series	B	x8		

Note 1: This listing is provided only as an example of typical memory devices available. It is not meant to be exhaustive.

2: Details of each programming algorithm family are provided in Appendix B.

3: For these devices, users must provide all data in the sector. The device will first erase the entire sector, then program it. These devices do not support Sector Erase commands.

4: Applicable only to x8/x16 selectable devices.

5: These devices have a unique set of programming algorithms. They are omitted for the sake of brevity.

APPENDIX B: PROGRAMMING ALGORITHMS FOR REPRESENTATIVE MEMORY DEVICES(1)

Command	Program Algorithm	Bus Cycles												
		Cycles Needed	First		Second		Third		Fourth		Fifth		Sixth	
			Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data
Read mode/ RESET	A	1	X	F0	—	—	—	—	—	—	—	—	—	—
	B	1	X	F0	—	—	—	—	—	—	—	—	—	—
	C	1	X	FF	—	—	—	—	—	—	—	—	—	—
Read Mfg. ID	A	4	555	AA	2AA	55	555	90	X00	01	—	—	—	—
	B	4	5555	AA	2AAA	55	5555	90	XX00	01	—	—	—	—
	C	2	X	90	(IA)	(ID)	—	—	—	—	—	—	—	—
Read Device ID	A	4	555	AA	2AA	55	555	90	X01	AD	—	—	—	—
	B	4	5555	AA	2AAA	55	5555	90	XX01	20	—	—	—	—
	C	2	X	90	(IA)	(ID)	—	—	—	—	—	—	—	—
Write	A	4	555	AA	2AA	55	555	A0	(WA)	(WD)	—	—	—	—
	B	4	5555	AA	2AAA	55	555	A0	(WA)	(WD)	—	—	—	—
	C	2	(WA)	40	(WA)	(WD)	—	—	—	—	—	—	—	—
Block Erase	A	6	555	AA	2AA	55	555	80	555	AA	2AA	55	(BA)	30
	B	6	5555	AA	2AAA	55	5555	80	5555	AA	2AAA	55	(BA)	30
	C	2	(BA)	20	(BA)	D0	—	—	—	—	—	—	—	—
Erase Suspend	A	1	X	B0	—	—	—	—	—	—	—	—	—	—
	B	(2)												
	C	1	X	B0	—	—	—	—	—	—	—	—	—	—
Erase Resume	A	1	X	30	—	—	—	—	—	—	—	—	—	—
	B	(2)												
	C	1	X	D0	—	—	—	—	—	—	—	—	—	—
Chip Erase	A	6	555	AA	2AA	55	555	80	555	AA	2AA	55	555	10
	B	6	5555	AA	2AAA	55	5555	80	5555	AA	2AAA	55	5555	10
	C	2	X	30	X	D0	—	—	—	—	—	—	—	—
Sector Protect Verify	A	4	555	AA	2AA	55	555	90	(SGA)	00/01	—	—	—	—
	B	4	5555	AA	2AAA	55	5555	90	(SGA)	00/01	—	—	—	—
	C	(2)												

Legend: WA = Write Address, WD = Write Data, IA = Identifier Address, ID = Identifier Data
BA = Block Address, SGA = Sector Group Address

Note 1: The information provided in this table is for reference only, and is not meant to be a comprehensive description of the device programming algorithms. For complete information, please refer to the manufacturer's data sheet.

2: Instruction unimplemented in this programming algorithm family.

Note the following details of the code protection feature on PICmicro® MCUs.

- The PICmicro family meets the specifications contained in the Microchip Data Sheet.
- Microchip believes that its family of PICmicro microcontrollers is one of the most secure products of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the PICmicro microcontroller in a manner outside the operating specifications contained in the data sheet. The person doing so may be engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as “unbreakable”.
- Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our product.

If you have any further questions about this matter, please contact the local sales office nearest to you.

DNV Certification, Inc. USA — ANSI-RAB QMS ACCREDITED — DNV — DNV MSC The Netherlands Accredited by the RvA — ISO 9001 / QS-9000 REGISTERED FIRM

Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999. The Company’s quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs and microperipheral products. In addition, Microchip’s quality system for the design and manufacture of development systems is ISO 9001 certified.

WORLDWIDE SALES AND SERVICE

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200 Fax: 480-792-7277
Technical Support: 480-792-7627
Web Address: http://www.microchip.com

Rocky Mountain
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7966 Fax: 480-792-7456

Atlanta
500 Sugar Mill Road, Suite 200B
Atlanta, GA 30350
Tel: 770-640-0034 Fax: 770-640-0307

Austin - Analog
8303 MoPac Expressway North
Suite A-201
Austin, TX 78759
Tel: 512-345-2030 Fax: 512-345-6085

Boston
2 Lan Drive, Suite 120
Westford, MA 01886
Tel: 978-692-3848 Fax: 978-692-3821

Boston - Analog
Unit A-8-1 Millbrook Tarry Condominium
97 Lowell Road
Concord, MA 01742
Tel: 978-371-6400 Fax: 978-371-0050

Chicago
333 Pierce Road, Suite 180
Itasca, IL 60143
Tel: 630-285-0071 Fax: 630-285-0075

Dallas
4570 Westgrove Drive, Suite 160
Addison, TX 75001
Tel: 972-818-7423 Fax: 972-818-2924

Dayton
Two Prestige Place, Suite 130
Miamisburg, OH 45342
Tel: 937-291-1654 Fax: 937-291-9175

Detroit
Tri-Atria Office Building
32255 Northwestern Highway, Suite 190
Farmington Hills, MI 48334
Tel: 248-538-2250 Fax: 248-538-2260

Los Angeles
18201 Von Karman, Suite 1090
Irvine, CA 92612
Tel: 949-263-1888 Fax: 949-263-1338

New York
150 Motor Parkway, Suite 202
Hauppauge, NY 11788
Tel: 631-273-5305 Fax: 631-273-5335

San Jose
Microchip Technology Inc.
2107 North First Street, Suite 590
San Jose, CA 95131
Tel: 408-436-7950 Fax: 408-436-7955

Toronto
6285 Northam Drive, Suite 108
Mississauga, Ontario L4V 1X5, Canada
Tel: 905-673-0699 Fax: 905-673-6509

ASIA/PACIFIC

Australia
Microchip Technology Australia Pty Ltd
Suite 22, 41 Rawson Street
Epping 2121, NSW
Australia
Tel: 61-2-9868-6733 Fax: 61-2-9868-6755

China - Beijing
Microchip Technology Consulting (Shanghai) Co., Ltd., Beijing Liaison Office
Unit 915
New China Hong Kong Manhattan Bldg.
No. 6 Chaoyangmen Beidajie
Beijing, 100027, No. China
Tel: 86-10-85282100 Fax: 86-10-85282104

China - Chengdu
Microchip Technology Consulting (Shanghai) Co., Ltd., Chengdu Liaison Office
Rm. 2401, Ming Xing Financial Tower
No. 88 TIDU Street
Chengdu 610016, China
Tel: 86-28-6766200 Fax: 86-28-6766599

China - Fuzhou
Microchip Technology Consulting (Shanghai) Co., Ltd., Fuzhou Liaison Office
Rm. 531, North Building
Fujian Foreign Trade Center Hotel
73 Wusi Road
Fuzhou 350001, China
Tel: 86-591-7557563 Fax: 86-591-7557572

China - Shanghai
Microchip Technology Consulting (Shanghai) Co., Ltd.
Room 701, Bldg. B
Far East International Plaza
No. 317 Xian Xia Road
Shanghai, 200051
Tel: 86-21-6275-5700 Fax: 86-21-6275-5060

China - Shenzhen
Microchip Technology Consulting (Shanghai) Co., Ltd., Shenzhen Liaison Office
Rm. 1315, 13/F, Shenzhen Kerry Centre,
Renminnan Lu
Shenzhen 518001, China
Tel: 86-755-2350361 Fax: 86-755-2366086

Hong Kong
Microchip Technology Hongkong Ltd.
Unit 901, Tower 2, Metroplaza
223 Hing Fong Road
Kwai Fong, N.T., Hong Kong
Tel: 852-2401-1200 Fax: 852-2401-3431

India
Microchip Technology Inc.
India Liaison Office
Divyasree Chambers
1 Floor, Wing A (A3/A4)
No. 11, O'Shaugnessey Road
Bangalore, 560 025, India
Tel: 91-80-2290061 Fax: 91-80-2290062

Japan
Microchip Technology Japan K.K.
Benex S-1 6F
3-18-20, Shinyokohama
Kohoku-Ku, Yokohama-shi
Kanagawa, 222-0033, Japan
Tel: 81-45-471- 6166 Fax: 81-45-471-6122

Korea
Microchip Technology Korea
168-1, Youngbo Bldg. 3 Floor
Samsung-Dong, Kangnam-Ku
Seoul, Korea 135-882
Tel: 82-2-554-7200 Fax: 82-2-558-5934

Singapore
Microchip Technology Singapore Pte Ltd.
200 Middle Road
#07-02 Prime Centre
Singapore, 188980
Tel: 65-334-8870 Fax: 65-334-8850

Taiwan
Microchip Technology Taiwan
11F-3, No. 207
Tung Hua North Road
Taipei, 105, Taiwan
Tel: 886-2-2717-7175 Fax: 886-2-2545-0139

EUROPE

Denmark
Microchip Technology Denmark ApS
Regus Business Centre
Lautrup hoj 1-3
Ballerup DK-2750 Denmark
Tel: 45 4420 9895 Fax: 45 4420 9910

France
Arizona Microchip Technology SARL
Parc d'Activite du Moulin de Massy
43 Rue du Saule Trapu
Batiment A - ler Etage
91300 Massy, France
Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79

Germany
Arizona Microchip Technology GmbH
Gustav-Heinemann Ring 125
D-81739 Munich, Germany
Tel: 49-89-627-144 0 Fax: 49-89-627-144-44

Germany - Analog
Lochhamer Strasse 13
D-82152 Martinsried, Germany
Tel: 49-89-895650-0 Fax: 49-89-895650-22

Italy
Arizona Microchip Technology SRL
Centro Direzionale Colleoni
Palazzo Taurus 1 V. Le Colleoni 1
20041 Agrate Brianza
Milan, Italy
Tel: 39-039-65791-1 Fax: 39-039-6899883

United Kingdom
Arizona Microchip Technology Ltd.
505 Eskdale Road
Winnersh Triangle
Wokingham
Berkshire, England RG41 5TU
Tel: 44 118 921 5869 Fax: 44-118 921-5820

06/01/01