

Using SRAM With A PIC16CXXX

Author: Rick Evans
Microchip Technology Inc.

INTRODUCTION

There are applications where a significant amount of data memory is required beyond what is in the microcontroller. For example, buffering communications data, creating large volatile tables and arrays. One interesting application is voice storage and playback. Some applications require relatively low frequencies, so a fast address/data bus and expensive FSRAM are not necessary.

This application note uses the PIC16C74. Since the PIC16C74 does not have an external address/data bus, one was created using the I/O ports. A software implementation of a multiplexed address/data bus is more than adequate for some applications. This application note discusses the read and write speeds

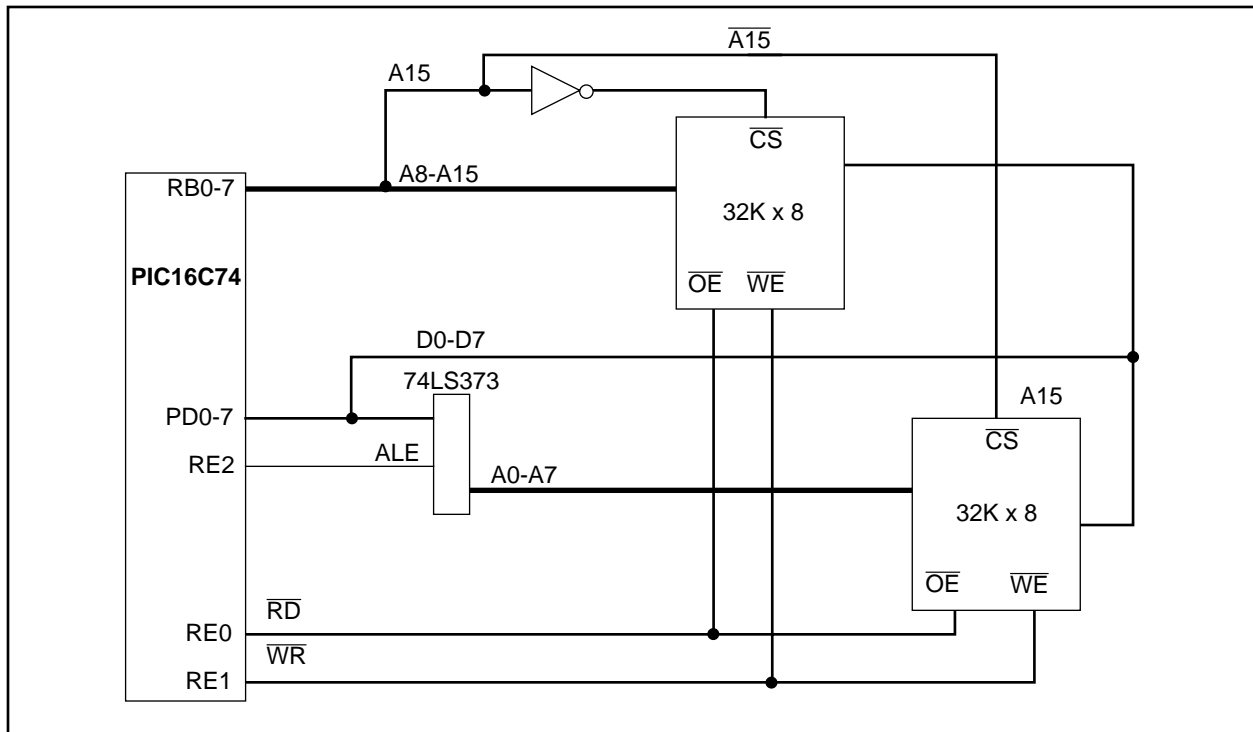
achievable with the PIC16C74 running at 20 MHz, hardware connections necessary, and software routines for reading and writing to external memory.

IMPLEMENTATION

The multiplexed address/data bus was created using nineteen port pins (Figure 1). PORTD is address lines 0 through 7 multiplexed with data lines 0 through 7. PORTB is the upper address lines 8-15. PORTE<0> is the \overline{RD} , PORTE<1> is \overline{WR} and PORTE<2> is ALE (address latch enable). One SRAM chip is address range 0000h-7FFFh, and the other SRAM chip is address range 8000h-FFFFh. The chip selecting is done using A15 and A15. A 74LS373 latch is needed to demultiplex the low order address/data bus.

The software needed to initialize the ports is in the subroutine `init_muxbus` (Appendix A). The initial states of the address lines and bus control signals are shown in the comments of the subroutine.

FIGURE 1: BLOCK DIAGRAM OF MULTIPLEXED ADDRESS/DATA BUS ON A PIC16C74



READ CYCLE

The software to do the read cycle is very straight forward. The read cycle drives the address onto the 16-bit bus, then latches the low order address lines 0-7 on to the memory device (Figure 2). When ALE goes low, address 0-7 is latched. Then the low order bus lines 0-7 are changed to input pins. Next, the read pin (RD) goes low which turns on the output buffers of the memory device. Some time later, the data from the memory is driven on to the data lines 0-7. Then \overline{RD} goes high and the output buffers of the memory device are disabled. Next, the low order data lines are changed back to outputs. The subroutine `read_extmem` shows the code used to emulate a read cycle on PORTB and PORTD.

You can ascertain from Table 1, that a slow SRAM can be used. There are three critical SRAM read cycle specifications:

- TACC: Address access time
- TOE: Output enable time
- TDF: Data float time

The address access time TACC corresponds to the TAVDV (1.6 μ s) of the emulated muxed bus. A FUJITSU MB84256C-70 has an access time of 70 ns. The output enable time TOE from the Fujitsu data sheet is 35 ns. The emulated muxed bus specification TRLDV (200 ns) corresponds to the TOE of the Fujitsu SRAM. The data float time TDF is only important if you are doing back to back bus cycles.

WRITE CYCLE

The write cycle moves data to the external SRAM. The address is driven on the sixteen address lines, then the low order address is latched by making ALE go high and then low. The data is moved to the low order address lines (data lines). Finally, the \overline{WR} pin is driven low and then high. When write goes high, the data is written to the SRAM. Figure 3 shows the bus timing diagram for the write cycle.

The write cycle is even simpler to implement in software than the read cycle. The subroutine `write_extmem` shows code used to write out data to an external SRAM.

The write pulse width is an important specification in memory design as well. For the PIC16C74 emulated address/data bus the write pulse width is TWLWH. The value for the write pulse width is very large compared to actual address/data bus implementations. The TAVWH specification corresponds to the TAW specification of most memory devices. The emulated address/data bus has relatively slow bus timings compared with most memory access times such as EPROMs and SRAMs.

SUMMARY

A 40-pin PIC16CXXX device such as the PIC16C74 can interface to external memory. This application note used sixteen address lines multiplexed with eight data lines to read/write to two external SRAMs. The read/write cycle times can be calculated easily given the single cycle instruction architectures of the PIC16CXXX. Flash, EPROM and other parallel bus memory devices can be similarly utilized.

TABLE 1: READ AND WRITE CYCLE BUS TIMINGS

Timing	Description	Minimum	Maximum
T _{CY}	Instruction cycle time @ 20 MHz	200 ns	DC
T _{LLH}	ALE pulse width	1 T _{CY}	—
T _{AVDV}	Address valid to data valid	—	7 T _{CY}
T _{RLDV}	Read low to data valid	1 T _{CY}	—
T _{RHDZ}	Read high to data float	0	1 T _{CY}
T _{WLWH}	WRITE pulse width	1 T _{CY}	—
T _{WHDX}	WRITE high to data no longer valid (data hold time)	—	2 T _{CY}
T _{AVWH}	Address valid to write high	—	5 T _{CY}

FIGURE 2: READ CYCLE ON MULTIPLEXED ADDRESS/DATA BUS

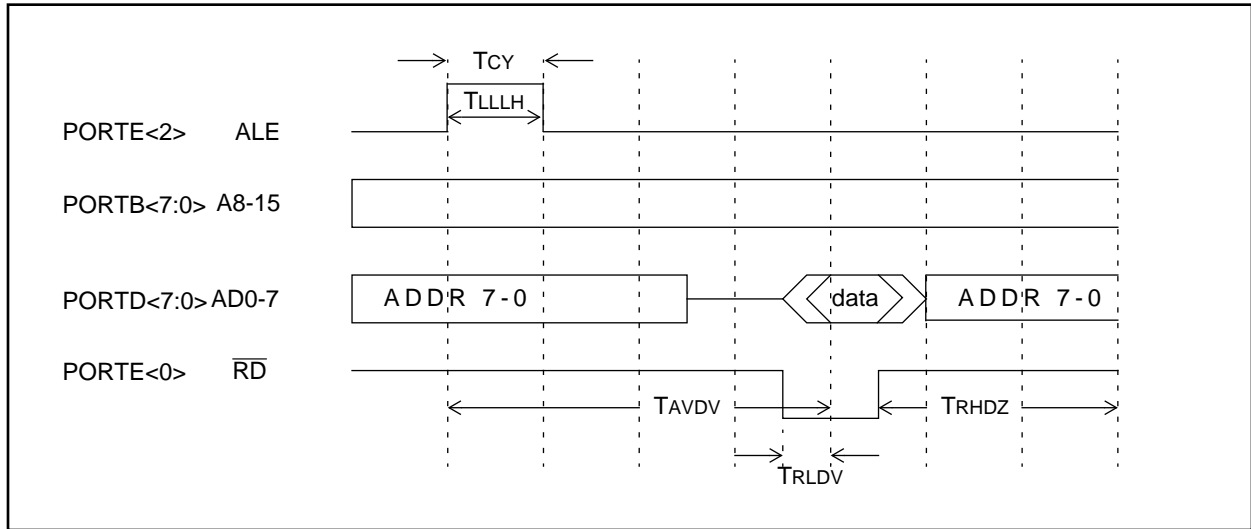
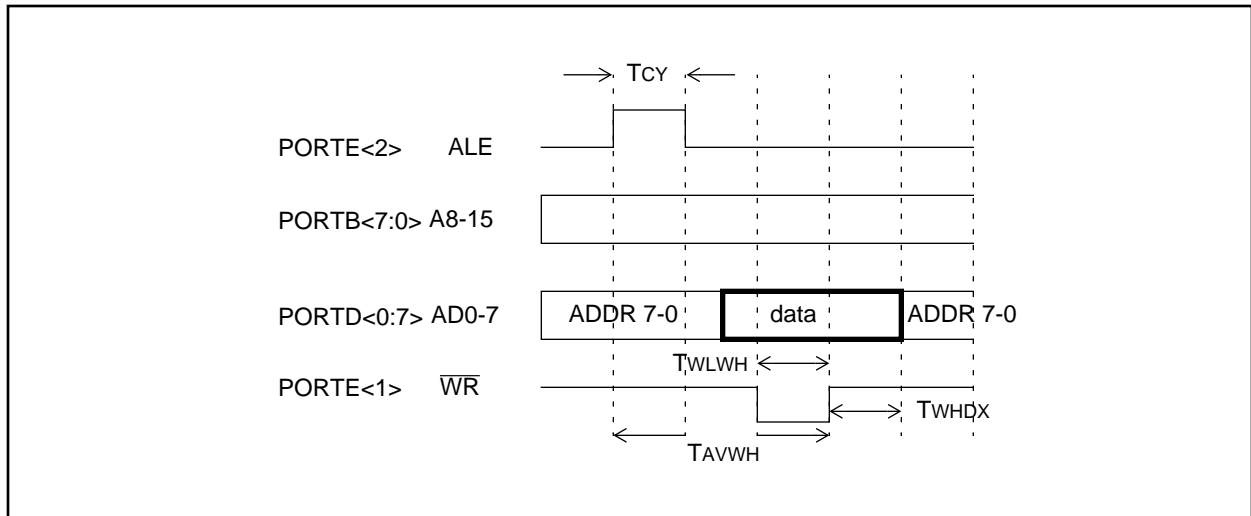


FIGURE 3: WRITE CYCLE BUS TIMINGS



APPENDIX A: PROGRAM LISTING: EXTERNAL MEMORY SUBROUTINES

```
*****
;*          Initialize the Multiplexed Address/Data Bus
;*
;* AD0-AD7 is PORTD 0-7
;* A8-A15 is PORTB 0-7
;* ALE is PORTE.2
;* RD# is PORTE.0
;* WR# is PORTE.1
;*
;* This init routine sets the multiplexed address/data bus up as
;* A0-A15 --> output low
;* ALE --> OUTPUT LOW
;* RD#,WR# --> OUTPUT HIGH
;*
*****

Init_MUXBUS
    bsf    STATUS,RP0          ;switch to bank 1 registers
    clrf   TRISB               ;set A8-A15 as output
    clrf   TRISD               ;set AD0-AD7 as output
    movlw  0xf8
    andwf  TRISE,F            ;ALE,RD#,WR# as output

    bcf    STATUS,RP0          ;switch to bank 0 registers

    clrf   ADHIGH              ;set A8-A15 to 0 (PORTB)
    clrf   ADLOW               ;set AD0-AD7 to 0 (PORTD)
    movlw  3                   ;ale=0,rd#=1,wr#=1
    movwf  PORTE
    return

;*****
;*          Read External Memory of muxed bus
;*
;* INPUT: PORTB =A8-A15, PORTD = AD0-AD7
;* OUTPUT: W reg contains 8-bit data read from ext. mem.
;* CHANGED: W reg, ALE, RD#
;* (This READ routine has been modified to save the low order
;* address before a READ is done. The data read from memory will
;* destroy the address. After the read is done, the low order address
;* is written back out to PORTD.)
;*
;*****

read_extmem
    movf   ADLOW,W             ;save low order address
    movwf  ADLOW_IMAGE

    bsf    PORTE,ALE           ;ALE high for 200ns, RD#, WR# low
    bcf    PORTE,ALE           ;ALE goes low (A0-7 latched)
    bsf    STATUS,RP0
    movlw  0xff
    movwf  TRISD               ;make PORTD input
    bcf    STATUS,RP0

    bcf    PORTE,RD            ;drop READ low

    movf   ADLOW,W             ;move read data from AD bus to w reg
    bsf    PORTE,RD            ;pull READ high (RD pulse is 400ns)
    bsf    STATUS,RP0
    clrf   TRISD               ;make PORTD (ADLOW) output again
    bcf    STATUS,RP0

    movwf  w_image             ;save READ data
    movf   ADLOW_IMAGE,W       ;restore low order address
```

```
    movwf  ADLOW          ;on port
    movf   w_image,W     ;restore READ data to w

    return
;*****
;*          Write to External Memory on muxed bus
;* INPUT:  PORTB= A8-A15, PORTD = AD0-AD7, W= 8-bit data to write
;* OUTPUT: NOTHING
;* CHANGED: PORTE IS TOGGLED FOR ALE,WR# AND PUT BACK TO 011B
;* (This WRITE routine has been modified to save the low order
;* address before a write is done. Then the low order address
;* is put back on PORTD after the write.)
;*****
write_extmem
    movwf  w_image      ;save w (data to write)
    movf   ADLOW,W      ;save the low order address
    movwf  ADLOW_IMAGE  ;restore w (data to write)
    movf   w_image,W

    bsf    PORTE,ALE    ;ALE high for 200ns, RD#,WR# low
    bcf    PORTE,ALE    ;latch lower address
    movwf  ADLOW        ;move write data to AD0-7
    bcf    PORTE,WR     ;WR# low for 200ns
    bsf    PORTE,WR     ;latch data in external memory

    movf   ADLOW_IMAGE,W
    movwf  ADLOW        ;restore low order address

    return
```

WORLDWIDE SALES & SERVICE

AMERICAS

Corporate Office

Microchip Technology Inc.
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 602-786-7200 Fax: 602-786-7277
Technical Support: 602 786-7627
Web: <http://www.microchip.com>

Atlanta

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

Boston

Microchip Technology Inc.
5 Mount Royal Avenue
Marlborough, MA 01752
Tel: 508-480-9990 Fax: 508-480-8575

Chicago

Microchip Technology Inc.
333 Pierce Road, Suite 180
Itasca, IL 60143
Tel: 630-285-0071 Fax: 630-285-0075

Dallas

Microchip Technology Inc.
14651 Dallas Parkway, Suite 816
Dallas, TX 75240-8809
Tel: 972-991-7177 Fax: 972-991-8588

Dayton

Microchip Technology Inc.
Two Prestige Place, Suite 150
Miamisburg, OH 45342
Tel: 937-291-1654 Fax: 937-291-9175

Los Angeles

Microchip Technology Inc.
18201 Von Karman, Suite 1090
Irvine, CA 92612
Tel: 714-263-1888 Fax: 714-263-1338

New York

Microchip Technology Inc.
150 Motor Parkway, Suite 416
Hauppauge, NY 11788
Tel: 516-273-5305 Fax: 516-273-5335

San Jose

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

Toronto

Microchip Technology Inc.
5925 Airport Road, Suite 200
Mississauga, Ontario L4V 1W1, Canada
Tel: 905-405-6279 Fax: 905-405-6253

ASIA/PACIFIC

Hong Kong

Microchip Asia Pacific
RM 3801B, Tower Two
Metroplaza
223 Hing Fong Road
Kwai Fong, N.T., Hong Kong
Tel: 852-2-401-1200 Fax: 852-2-401-3431

India

Microchip Technology India
No. 6, Legacy, Convent Road
Bangalore 560 025, India
Tel: 91-80-229-0061 Fax: 91-80-229-0062

Korea

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

Shanghai

Microchip Technology
RM 406 Shanghai Golden Bridge Bldg.
2077 Yan'an Road West, Hongjiao District
Shanghai, PRC 200335
Tel: 86-21-6275-5700
Fax: 86 21-6275-5060

Singapore

Microchip Technology Taiwan
Singapore Branch
200 Middle Road
#10-03 Prime Centre
Singapore 188980
Tel: 65-334-8870 Fax: 65-334-8850

Taiwan, R.O.C

Microchip Technology Taiwan
10F-1C 207
Tung Hua North Road
Taipei, Taiwan, ROC
Tel: 886 2-717-7175 Fax: 886-2-545-0139

EUROPE

United Kingdom

Arizona Microchip Technology Ltd.
Unit 6, The Courtyard
Meadow Bank, Furlong Road
Bourne End, Buckinghamshire SL8 5AJ
Tel: 44-1628-851077 Fax: 44-1628-850259

France

Arizona Microchip Technology SARL
Zone Industrielle de la Bonde
2 Rue du Buisson aux Fraises
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 München, Germany
Tel: 49-89-627-144 0 Fax: 49-89-627-144-44

Italy

Arizona Microchip Technology SRL
Centro Direzionale Colleone
Palazzo Taurus 1 V. Le Colleoni 1
20041 Agrate Brianza
Milan, Italy
Tel: 39-39-6899939 Fax: 39-39-6899883

JAPAN

Microchip Technology Intl. Inc.
Benex S-1 6F
3-18-20, Shin Yokohama
Kohoku-Ku, Yokohama
Kanagawa 222 Japan
Tel: 81-4-5471- 6166 Fax: 81-4-5471-6122

5/8/97



MICROCHIP

All rights reserved. © 1997, Microchip Technology Incorporated, USA. 6/97

Information contained in this publication regarding device applications and the like is intended for suggestion only and may be superseded by updates. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Microchip's products as critical components in life support systems is not authorized except with express written approval by Microchip. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights. The Microchip logo and name are registered trademarks of Microchip Technology Inc. in the U.S.A. and other countries. All rights reserved. All other trademarks mentioned herein are the property of their respective companies.