Embedded applications increasingly want more integration and power, in less space for less cost. Using low power Serial EEPROMs (SEE) for application firmware, lookup tables, and microcode coupled with small footprints makes for permanent storage at respectable savings. One additional method of saving on the power budget is selectively powering off components when not needed, a basic for embedded power management. The low-power SEEs offered by Microchip Technology Inc., offer an additional benefit, powering the SEE from a microcontroller port. This allows the host controller to not only manipulate the Serial EEPROM Reads and Write, but also the periods when it is powered off or on. Satellite communications use this technique to save power and total dose accumulation. We call this technique POWER PORT™.

The microcontroller port must have sufficient Ioh (source current) to sustain the voltage and current for all memory functions, READ, ERASE, and WRITE. Obviously, not all memory or peripheral devices could be powered thusly, but Microchip’s SEE devices will function in this environment.

The microcontroller, using its internal software and hardware decision functions, determines when it needs to communicate with the memory device, then acts accordingly. Any standard wake-up sequence will accomplish this task. The wake-up code needs only power up the memory and wait for the power to become stable before doing a read or write by driving the POWER PORT high. Then all serial communication executes normally. The SEE devices are powered off for additional power savings and the data or code is utilized from RAM. Obviously, the port output must be allowed to settle, but normal operation of the output structures would guarantee that this would be met. The I/O port Tpd for the Microchip PIC16C5X, is specified at 40ns maximum.

The 24LCXX and 93LCXX CMOS SEE series parts from Microchip were designed to achieve low current consumption across all ranges of operation. The four primary ICC parameters for these products are:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICC STANDBY</td>
<td>Not in an active operation while VCC is supplied.</td>
</tr>
<tr>
<td>ICC READ</td>
<td>The part is in a READ operation.</td>
</tr>
<tr>
<td>ICC PEAK WRITE</td>
<td>The BYTE / PAGE WRITE and ERASE operations have self timed cycles of 10 ms. A typical of 4 ms is the actual time of the operation. This is the amount of time when the ICC requires the most current (PEAK WRITE). The part is drawing STANDBY ICC during the remaining 6ms of the cycle.</td>
</tr>
<tr>
<td>ICC AVG WRITE</td>
<td>The avg of the PEAK WRITE ICC and STANDBY ICC during the self-timed 10ms write cycle.</td>
</tr>
</tbody>
</table>

The attached characteristic curves (Figure 1 and Figure 2) indicate that ICC PEAK WRITE current consumes the most current. The worst case condition is at 6.0V and –40°C. The 24LCXX series parts draw a typical 3.2 mA and the 93LCXX series parts draw a typical of 2.0 mA. These low ICC characteristics offer a unique current saving benefit for battery applications. Figure 3 and Figure 4 illustrate the sink and source current capabilities of the PIC16C5X family of microcontrollers. It is clear from these characterization curves that the microcontroller can deliver sufficient current across all temperature ranges to power a SEE using the POWER PORT technique.

Figure 5 shows the connection scheme for the Microchip PIC16C54. It should be noted that not all versions of competitive microcontrollers are capable of powering a device in this manner and the specific data sheets for the microcontroller being considered must be consulted for maximum source current. The microcontroller port must be capable of sourcing sufficient current for the duration of the write cycle or 10ms, worse case. The peak write requirement for the 24LCXX product family is 3.2 mA at 5.5 Vdc (–40°C).

Listing A demonstrates the appropriate code sequences when using the PIC16C54 microcontroller. The sequences included are power control, start bit, stop bit, send and receive bit, Tx and Rx, and a general addressing routine.
FIGURE 1: TYPICAL Icc FOR 24LCXX

24LCXX Typical Icc Peak Write (mA)

24LCXX Typical Icc Avg Write (μA)

24LCXX Typical Icc Read (μA)

24LCXX Typical Icc Standby (μA)

FIGURE 2: TYPICAL Icc FOR 93CXX

93LCXX Typical Icc Peak Write (mA)

93LCXX Typical Icc Avg Write (μA)

93LCXX Typical Icc Read (μA)

93LCXX Typical Icc Standby (μA)
The primary benefits of this application are:

- The SEE is completely powered down to save power when the SEE is not executing an operation. This will directly effect the total system power consumption. This means that the SEE is in a total quiescent state and even the standby current savings are realized, greatly increasing usable battery life, and consequently allowing for a more sophisticated design on the same power budget.
- The very fast 5 μs power-up time minimizes power-up delay.
- Since the serial operation is gated by a stable microcontroller V\text{OH}, risk of data being corrupted by a glitch is minimized. This, in effect, is a regulated V\text{CC} supply and provides a reliable power source to ensure data integrity.

Several cautions need to be noted:

1. Gang powering multiple devices must not exceed the I/O port IO\text{H} or capacitive load specifications.
2. The total power requirements vs. power budget must be considered, including the extra drain on the microcontroller.
3. The microcontroller I\text{CC} max must not be exceeded.
4. Normal decoupling methods must be employed.
5. The microcontroller IO\text{H} for the port in use must not be exceeded.

Figure 6 shows a typical power on to start bit sequence. Notice that the device is available to receive a clock at 5 μs after V\text{CC} has become stable.
APPENDIX A:
LIST P=16C54
;
; Sample test program to power up serial EEPROM
; using PIC16/17 port A, then write one byte and read same byte, then repeat forever.
;
;*******************************************************************************************
port_a equ 5h ; port 5 used for device
; address select
port_b equ 6h ; port 6 used for data and clock lines
eeprom equ 0ah ; bit buffer
addr equ 0ch ; address register
datai equ 0dh ; stored data input reg.
datao equ 0eh ; stored data output reg.
slave equ 0fh ; device address
; (1010xxx0)
txbuf equ 10h ; tx buffer
count equ 11h ; bit counter
bcount equ 12h ; byte counter
rxbuf equ 13h ; receive buffer
loops equ 15h ; delay loop counter
loops2 equ 16h ; delay loop counter 2
;
; Bit Assignments
;
; di equ 7 ; eeprom input
do equ 6 ; eeprom output
sdata equ 7 ; data line (port_b)
sclk equ 6 ; clock line (port_b)
vcc equ 3 ; vcc for dut (port_a)
;
org 01ffh
begin goto PWRUP
org 000h
goto PWRUP
;
;*******************************************************************************************
; DELAY ROUTINE
; this routine takes the value in loops and loops that many times. Every increase in ‘loops’ yields approx 1 more millisecond.
; i.e., if ‘loops’ is 10 then the wait period is approx 10 milliseconds.
;
;———————————————————————————————————————————————————————————————————————————————————————————
WAIT
;
; top2 movlw .110
; top movwf loops2 ; sit and wait
nop
nop
nop
nop
nop
decfsz loops2; inner loop done?
goto top; no, go again
decfsz loops ; outer loop done?
goto top2; no, go again
retlw 0; yes, return from sub
; Start Bit Subroutine
; this routine generates a start bit

BSTART
    movlw b'00111111'
    tris port_b ; port b for output
    bcf port_b,sdata ; set clock high
    bcf port_b,sclk ; set clock high
    ; data line goes low during high clock for start bit
    bcf port_b,sdata
    bcf port_b,sclk ; start clock train
    retlw 0
; End of Subroutine

; Stop Bit Subroutine
; this routine generates a stop bit

BSTOP
    movlw b'00111111'
    tris port_b ; set data/clock lines as outputs
    bcf port_b,sdata ; make sure data line is low
    bcf port_b,sclk ; set clock high
    bcf port_b,sdata ; data goes high while clock high
    ; for stop bit
    bcf port_b,sclk ; set clock low again

nop
nop
retlw 0
;
;
End of Subroutine
;
;*******************************************************************************************
;
Serial data send 1 bit from PIC16/17 to dut
;
;
BITOUT
movlw b'00111111'    ; set data,clock as outputs
tris port_b
btfss eeprom,do
goto BIT0
bsf port_b,sdata ; output bit 0
btfss eeprom,di
goto TX
bsf port_b,sclk ; data line clocked low by device
;
BIT0
bsf port_b,sdata ; output bit 0

CLK1
nop
nop
nop
nop
bcf port_b,sclk ; return clock line low

BIT2
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
bcf port_b,sclk ; set clock line high

BITIN
movlw b'10111111'    ; make sdata an input line
tris port_b
bcf eeprom,di ; assume input bit low
bsf port_b,sclk ; set clock line high
nop ; just sit here a sec
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
nop
btfsc port_b,sdata ; read data line
bsf eeprom,di ; set input bit if needed
bcf port_b,sclk ; set clock line low
retlw 0 ; hit the road
;
;
TX
movlw .8
movwf count ; set the #bits to 8
;
TXLP
bcf eeprom,do
btfsc txbuf,7
bsf eeprom,do  ; otherwise data bit =1
call BITOUT    ; serial data out
rlf txbuf      ; rotate txbuf left
decfsz count   ; # bits done?
goto TXLP      ; no - go again
call BITIN     ; read ack bit
rlf txbuf      ; rotate txbuf left
retlw 0

; End of Subroutine

;**************************************************

; Receive data Routine
; this routine gets a byte of data from the part into 'rxbuf'
;
RX
movlw .8        ; set # bits to 8
movwf count
clrf rxbuf      ; clear receive buffer

RXLP
rlf rxbuf      ; rotate buffer left 1 bit
bcf rxbuf,0    ; assume bit is zero
call BITIN     ; read a bit
btfsc eeprom,di ; input bit high?
bsf rxbuf,0    ; yes, set buffer bit high
decfsz count   ; 8 bits done?
goto RXLP      ; no, do another
bcf eeprom,do  ; set ack bit = 0
call BITOUT    ; to finish transmission
retlw 0

;**************************************************

; Power up routine
; this routine blinks the lights
;
PWRUP
movlw b'00000001'
tris port_a  ; set RA0 as input, rest output
bsf port_a,vcc ; turn on power to dut
call PWRUP    ; wait for dut to power up

;**************************************************

; Byte Write Routine
; this writes the data in "55h" to the first byte
; in the serial EEPROM.
;
WRBYTE

movlw b'10100000' ; set slave address and write mode
movwf slave
movlw b'01010101' ; set data to 55h
movwf datao

clrf addr      ; set address to 00h

call BSTART    ; generate start bit
movf slave,w  ; get slave address
movwf txbuf    ; into transmit buffer
call TX        ; and send it
movf addr,w   ; get word address
movwf txbuf    ; into transmit buffer
call TX        ; and send it
movf datao,w  ; move data
movwf txbuf    ; to transmit buffer
call TX ; and transmit it
call BSTOP ; generate stop bit
movlw .10
movwf loops ; set delay time to give
; now drop through and do the read

*******************************************************************************************
;         READ (read routine)
;         this routine reads the first address
;         of the dut

READ ;
movlw b'10100000' ; set slave address and write mode
movwf slave ;
clrf addr ; set address to 00h
; call BSTART ; generate start bit
nop
nop
movf slave,w ; get slave address
movwf txbuf ; into transmit buffer
call TX ; and send it
call TX ; and send it
nop
nop
call BSTART ; generate start bit
nop
nop
movlw b'10100001' ; get slave address and read mode
movwf txbuf ; into transmit buffer
call TX ; and transmit it
nop
call RX ; get 8 bits of data
bsf eeprom,do
call BITOUT ; send high ack bit and then a
call BSTOP ; stop bit to end transmission from dut
nop ;
NOP
bCF port_a,vcc ; turn power to dut off
movlw .100
movwf loops
call WAIT ; wait awhile
goto PWRUP ; go do the whole thing over again

END
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.

Information contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. 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.

Trademarks

The Microchip name and logo, the Microchip logo, FilterLab, KEELOQ, microID, MPLAB, PIC, PICmicro, PICMASTER, PICSTART, PRO MATE, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

dsPIC, ECONOMONITOR, FanSense, FlexROM, fuzzyLAB, In-Circuit Serial Programming, ICSP, ICEPIC, microPort, Migratable Memory, MPASM, MPLIB, MPLINK, MPSIM, MXDEV, PICC, PICDEM, PICDEM.net, rPIC, Select Mode and Total Endurance are trademarks of Microchip Technology Incorporated in the U.S.A.

Serialized Quick Turn Programming (SQTP) is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

© 2002, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

Printed on recycled paper.
**AMERICAS**

**Corporate Office**
2335 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**
2335 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: 770-640-0075

**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: 972-818-7924 Fax: 972-818-2924

**Detroit**
Tri-Aria Office Building
32265 Northwestern Highway, Suite 190
Farmington Hills, MI 48334
Tel: 248-538-2250 Fax: 248-538-2260

**Kokomo**
9548 North Vine Street
Kokomo, IN 46902
Tel: 765-864-8360 Fax: 765-864-8387

**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: 631-233-5305 Fax: 631-233-5305

**San Jose**
Microchip Technology Inc.
2107 North First Street, Suite 590
San Jose, CA 95131
Tel: 408-436-7955 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 Lawson 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 951
Bei Hai Wan Tai 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. 3401, 24th Floor,
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
Unit 71 WuLi Road
Fuzhou 350001, China
Tel: 86-591-7503506 Fax: 86-591-7503521

**Europe**

**Denmark**
Microchip Technology Nordic ApS
Regus Business Centre
Lautrup høj 1-3
Ballpark DK-2750 Denmark
Tel: 45 4420 9895 Fax: 45 4420 9910

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

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

**Italy**
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

**Japan**
Microchip Technology Japan K.K.
Benex S-1 6F
3-18-20, Shin'yokohama
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, 189980
Tel: 65-6334-8870 Fax: 65-6334-8850

**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

© 2002 Microchip Technology Inc.