MICROCHIP

LCD PIC® MCU Tips 'n Tricks

Table of Contents

NOTES:

TIPS ‘N TRICKS INTRODUCTION

Using an LCD PIC® MCU for any embedded application can provide the benefits of system control and human interface via an LCD. Design practices for LCD applications can be further enhanced through the implementation of these suggested “Tips ‘n Tricks”.

This booklet describes many basic circuits and software building blocks commonly used for driving LCD displays. The booklet also provides references to Microchip application notes that describe many LCD concepts in more detail.

TIP #1 Typical Ordering Considerations and Procedures for Custom Liquid Displays

1. Consider what useful information needs to be displayed on the custom LCD and the combination of alphanumeric and custom icons that will be necessary.
2. Understand the environment in which the LCD will be required to operate. Operating voltage and temperature can heavily influence the contrast of the LCD and potentially limit the type of LCD that can be used.
3. Determine the number of segments necessary to achieve the desired display on the LCD and reference the PIC® Microcontroller LCD matrix for the appropriate LCD PIC Microcontroller.
4. Create a sketch/mechanical print and written description of the custom LCD and understand the pinout of the LCD. (Pinout definition is best left to the glass manufacturer due to the constraints of routing the common and segment electrodes in two dimensions.)

5. Send the proposed LCD sketch and description for a written quotation to at least 3 vendors to determine pricing, scheduling and quality concerns.
   5.1 Take into account total NRE cost, price per unit, as well as any setup fees.
   5.2 Allow a minimum of two weeks for formal mechanical drawings and pin assignments and revised counter drawings.
6. Request a minimal initial prototype LCD build to ensure proper LCD development and ensure proper functionality within the target application.
   6.1 Allow typically 4-6 weeks for initial LCD prototype delivery upon final approval of mechanical drawings and pin assignments.
7. Upon receipt of prototype LCD, confirm functionality before giving final approval and beginning production of LCD.

> **Note:** Be sure to maintain good records by keeping copies of all materials transferred between both parties, such as initial sketches, drawings, pinouts, etc.

TIP #2 LCD PIC® MCU Segment/Pixel Table

TABLE 2-1: SEGMENT MATRIX TABLE

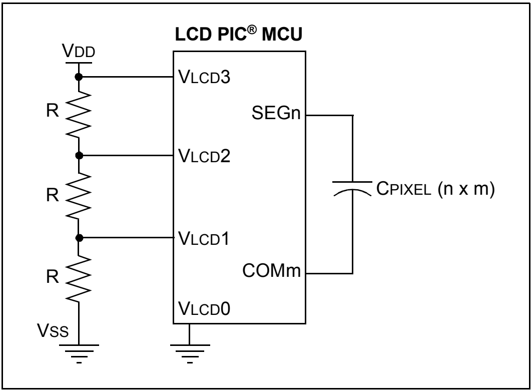
Multiplex Commons	Maximum Number of Segments/Pixels					Bias
	PIC16F913/916	PIC16F914/917	PIC16F946	PIC18F6X90 (PIC18F6XJ90)	PIC18F8X90 (PIC18F8XJ90)	
Static (COM0)	15	24	42	32/ (33)	48	Static
1/2 (COM1: COM0)	30	48	84	64/ (66)	96	1/2 or 1/3
1/3 (COM2: COM0)	45	72	126	96/ (99)	144	1/2 or 1/3
1/4 (COM3: COM0)	60	96	168	128/ (132)	192	1/3

This Segment Matrix table shows that Microchip's 80-pin LCD devices can drive up to 4 commons and 48 segments (192 pixels), 64-pin devices can drive up to 33 segments (132 pixels), 40/44 pin devices can drive up to 24 segments (96 pixels) and 28-pin devices can drive 15 segments (60 segments).

TIP #3 Resistor Ladder for Low Current

Bias voltages are generated by using an external resistor ladder. Since the resistor ladder is connected between V_{DD} and V_{SS}, there will be current flow through the resistor ladder in inverse proportion to the resistance. In other words, the higher the resistance, the less current will flow through the resistor ladder. If we use 10K resistors and V_{DD} = 5V, the resistor ladder will continuously draw 166 µA. That is a lot of current for some battery-powered applications.

FIGURE 3-1: RESISTOR LADDER

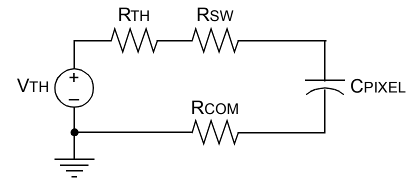


How do we maximize the resistance without adversely effecting the quality of the display? Some basic circuit analysis helps us determine how much we can increase the size of the resistors in the ladder.

The LCD module is basically an analog multiplexer that alternately connects the LCD voltages to the various segment and common pins that connect across the LCD pixels. The LCD pixels can be modeled as a capacitor. Each tap point on the resistor ladder can be modeled as a Thevenin equivalent circuit. The Thevenin resistance is 0 for VLCD3 and VLCD0, so we look at the two cases where it is non-zero, VLCD2 and VLCD1.

The circuit can be simplified as shown in Figure 3-2. RSW is the resistance of the segment multiplex switch; RCOM is the resistance of the common multiplex switch.

FIGURE 3-2: SIMPLIFIED LCD CIRCUIT

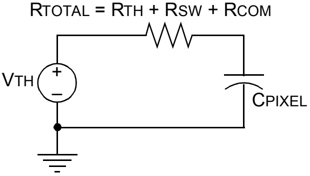


The Thevenin voltage is equal to either 2/3 VDD, or 1/3 VDD, for the cases where the Thevenin resistance is non-zero. The Thevenin resistance is equal to the parallel resistance of the upper and lower parts of the resistor ladder.

FIGURE 3-3: LCD CIRCUIT RESISTANCE ESTIMATE

$R_{TH} = (2R * R)/(2R + R)$
$R_{TH} = 2R^2/3R$
$R_{TH} = 2R/3$

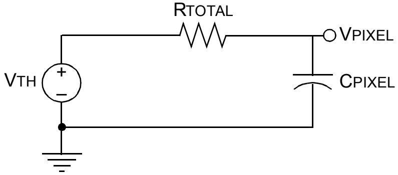
$R_{SW} = 4.7K$
$R_{COM} = 0.4K$



As you can see, we can model the drive of a single pixel as an RC circuit, where the voltage switches from 0V to VLCD2, for example. For LCD PIC® microcontrollers, we can estimate the resistance of the segment and common switching circuits as about 4.7K and 0.4K, respectively.

We can see that the time for the voltage across the pixel to change from 0 to VTH will depend on the capacitance of the pixel and the total resistance, of which the resistor ladder Thevenin resistance forms the most significant part.

FIGURE 3-4: VOLTAGE CHANGE ACROSS PIXEL



The step response of the voltage across a pixel is subject to the following equation:

EQUATION 3-1:

$$V_{PIXEL} = V_{TH}\,(1 - e^{-t/RC})$$

By manipulating the equation, we can see that it will take a time equal to 4 time constants for the pixel voltage to reach 98% of the bias voltage.

FIGURE 3-5: STEP RESPONSE DIAGRAM

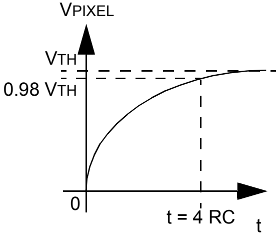
$V_{PIXEL}/V_{TH} = 1 - e^{-t/RC}$

$98\% = 1 - e^{-t/RC}$

$2\% = e^{-t/RC}$

$\ln(.02) = -t/RC$

$t = \sim 4\ RC$



Now we need to estimate the capacitance. Capacitance is proportional to the area of a pixel. We can measure the area of a pixel and estimate the capacitance as shown. Obviously, a bigger display, such as a digital wall clock, will have bigger pixels and higher capacitance.

EQUATION 3-2:

$$C_{PIXEL} = 1500\ pF/cm^2$$
$$AREA_{PIXEL} = 1\ mm * 3\ mm = .03\ cm^2$$
$$C_{PIXEL} = 45\ pF$$

We want the time constant to be much smaller than the period of the LCD waveform, so that rounding of the LCD waveform will be minimized. If we want the RC to be equal to 100 μS, then the total resistance can be calculated as shown:

EQUATION 3-3:

$$R_{TOTAL} = 100\ \mu S/45\ pF = 2.22\ mOhms$$
$$R_{TH} = 2.2M - 5.1K = 2.2M$$

The resistance of the switching circuits within the LCD module is very small compared to this resistance, so the Thevenin resistance of the resistor ladder at $V_{LCD}2$ and $V_{LCD}1$ can be treated the same as R_{TOTAL}. We can then calculate the value for R that will give us the correct Thevenin resistance.

EQUATION 3-4:

$$R = 3\ R_{TH}/2 = 3.3M$$

Now we can calculate the current through the resistor ladder if we used 3.3 mOhm resistors.

EQUATION 3-5:

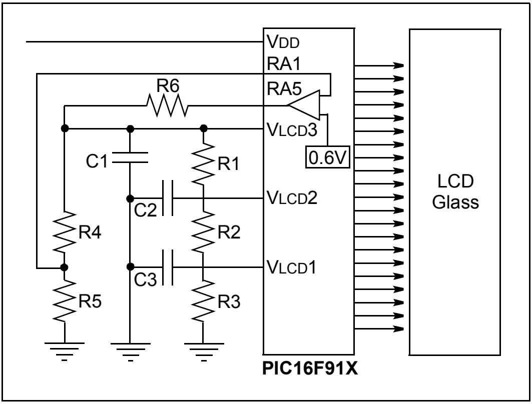
$$R_{LADDER} = 9.9M,$$
$$I_{LADDER} = 5V/9.9M = 0.5\ \mu A$$

Use this process to estimate maximum resistor sizes for your resistor ladder and you will drastically reduce power consumption for your LCD application. Don’t forget to observe the display over the operating conditions of your product (such as temperature, voltage and even, humidity) to ensure that contrast and display quality is good.

TIP #4 Contrast Control with a Buck Regulator

Contrast control in any of the LCD PIC MCUs is accomplished by controlling the voltages applied to the VLCD voltage inputs. The simplest contrast voltage generator is to place a resistor divider across the three pins. This circuit is shown in the data sheet. The resistor ladder method is good for many applications, but the resistor ladder does not work in an application where the contrast must remain constant over a range of VDDs. The solution is to use a voltage regulator. The voltage regulator can be external to the device, or it can be built using a comparator internal to the LCD PIC microcontroller.

FIGURE 4-1: VOLTAGE GENERATOR WITH RESISTOR DIVIDER



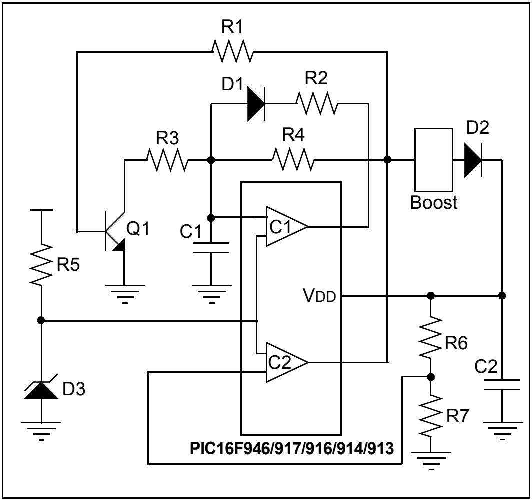
The PIC16F946/917/916/914/913 devices have a special Comparator mode that provides a fixed 0.6V reference. The circuit shown in Figure 4-1 makes use of this reference to provide a regulated contrast voltage. In this circuit, R1, R2 and R3 provide the contrast control voltages. The voltage on VLCD3 is compared to the internal voltage reference by dividing the voltage at VLCD3 at R4 and R5 and applying the reduced voltage to the internal comparator. When the voltage at VLCD3 is close to the desired voltage, the output of the comparator will begin to oscillate. The oscillations are filtered into a DC voltage by R6 and C1. C2 and C3 are simply small bypass capacitors to ensure that the voltages at VLCD1 and VLCD2 are steady.

TIP #5 Contrast Control Using a Boost Regulator

In LCD Tip #4, a buck converter was created using a comparator. This circuit works great when V_{DD} is greater than the LCD voltage. The PIC microcontroller can operate all the way down to 2.0V, whereas most low-voltage LCD glass only operates down to 3V. In a battery application, it is important to stay operational as long as possible. Therefore, a boost converter is required to boost 2.0V up to 3.0V for the LCD.

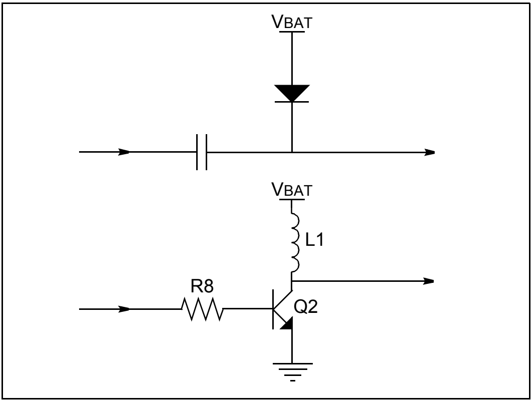
The figure below shows one circuit for doing this.

FIGURE 5-1: BOOST CONVERTER



In this circuit, both comparators are used. The voltage setpoint is determined by the value of Zenier diode D3 and the voltage at R6:R7. The rest of the circuit creates a simple multivibrator to stimulate a boost circuit. The boost circuit can be inductor or capacitor-based. When the output voltage is too low, the multivibrator oscillates and causes charge to build up in C2. As the voltage at C2 increases, the multivibrator will begin to operate sporadically to maintain the desired voltage at C2.

FIGURE 5-2: TWO TYPES OF BOOST CONVERTER

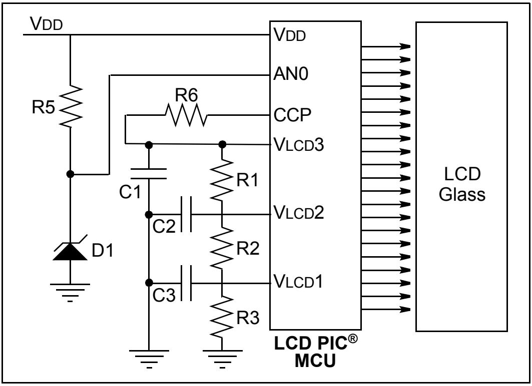


The two methods of producing a boost converter are shown above. The first circuit is simply a switched capacitor type circuit. The second circuit is a standard inductor boost circuit. These circuits work by raising VDD. This allows the voltage at VLCD to exceed VDD.

TIP #6 Software Controlled Contrast with PWM for LCD Contrast Control

In the previous contrast control circuits, the voltage output was set by a fixed reference. In some cases, the contrast must be variable to account for different operating conditions. The CCP module, available in the LCD controller devices, allows a PWM signal to be used for contrast control. In Figure 6-1, you see the buck contrast circuit modified by connecting the input to RA6 to a CCP pin. The resistor divider created by R4 and R5 in the previous design are no longer required. An input to the ADC is used to provide feedback but this can be considered optional. If the ADC feedback is used, notice that it is used to monitor the VDD supply. The PWM will then be used to compensate for variations in the supply voltage.

FIGURE 6-1: SOFTWARE CONTROLLED VOLTAGE GENERATOR



TIP #7 Driving Common Backlights

Any application that operates in a low light condition requires a backlight. Most low-cost applications use one of the following backlights:

1) Electroluminescent (EL),

2) LEDs in series, or

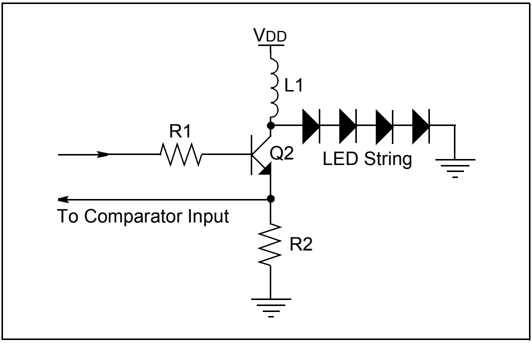
3) LEDs in parallel.

Other backlight technologies, such as CCFL, are more commonly used in high brightness graphical panels, such as those found in laptop computers. The use of white LEDs is also more common in color LCDs, where a white light source is required to generate the colors.

Driving an EL panel simply requires an AC signal. You may be able to generate this signal simply by using an unused segment on the LCD controller. The signal can also be generated by a CCP module or through software. The AC signal will need to pass through a transformer for voltage gain to generate the required voltage across the panel.

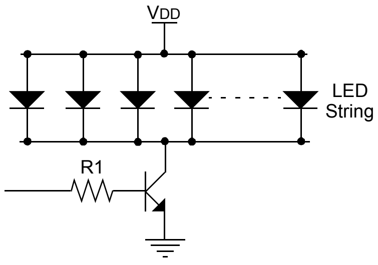
LEDs in series can be easily driven with a boost power supply. In the following diagram, a simple boost supply is shown. In this circuit, a pulse is applied to the transistor. The pulse duration is controlled by current through R2. When the pulse is turned off, the current stored in the inductor will be transferred to the LEDs. The voltage will rise to the level required to drive the current through the LEDs. The breakdown voltage of the transistor must be equal to the forward voltage of the LEDs multiplied by the number of LEDs. The comparator voltage reference can be adjusted in software to change the output level of the LEDs.

FIGURE 7-1: SIMPLE BOOST SUPPLY



If the LEDs are in parallel, the drive is much simpler. In this case, a single transistor can be used to sink the current of many LEDs in parallel. The transistor can be modulated by PWM to achieve the desired output level. If VDD is higher than the maximum forward voltage, a resistor can be added to control the current, or the transistor PWM duty cycle can be adjusted to assure the LEDs are operating within their specification.

FIGURE 7-2: LEDS IN PARALLEL



TIP #8 In-Circuit Debug (ICD)

There are two potential issues with using the ICD to debug LCD applications. First, the LCD controller can Freeze while the device is Halted. Second, the ICD pins are shared with segments on the PIC16F946/917/916/914/913 MCUs.

When debugging, the device is Halted at breakpoints and by the user pressing the pause button. If the ICD is configured to Halt the peripherals with the device, the LCD controller will Halt and apply DC voltages to the LCD glass. Over time, these DC levels can cause damage to the glass; however, for most debugging situations, this will not be a consideration. The PIC18F LCD MCUs have a feature that allows the LCD module to continue operating while the device has been Halted during debugging. This is useful for checking the image of the display while the device is Halted and for preventing glass damage if the device will be Halted for a long period of time.

The PIC16F946/917/916/914/913 multiplex the ICSP™ and ICD pins onto pins shared with LCD segments 6 and 7. If an LCD is attached to these pins, the device can be debugged with ICD; however, all the segments driven by those two pins will flicker and be uncontrolled. As soon as debugging is finished and the device is programmed with Debug mode disabled, these segments will be controlled correctly.

TIP #9 LCD in Sleep Mode

If you have a power-sensitive application that must display data continuously, the LCD PIC microcontroller can be put to Sleep while the LCD driver module continues to drive the display.

To operate the LCD in Sleep, only two steps are required. First, a time source other than the main oscillator must be selected as the LCD clock source, because during Sleep, the main oscillator is Halted. Options are shown for the various LCD PIC MCUs.

TABLE 9-1: OPTIONS FOR LCD IN SLEEP MODE

Part	LCD Clock Source	Use in Sleep?
PIC16C925/926	FOSC/256	No
	T1OSC	Yes
	Internal RC Oscillator	Yes
PIC16F946/917/ 916/914/913	FOSC/8192	No
	T1OSC/32	Yes
	LFINTOSC/32	Yes
PIC18F6X90 PIC18F8X90 PIC18F6XJ90 PIC18F8XJ90	(FOSC/4)/8192	No
	T1OSC	Yes
	INTRC/32	Yes

Second, the Sleep Enable bit (SLPEN) must be cleared. The LCD will then continue to display data while the part is in Sleep. It's that easy!

When should you select the internal RC oscillator (or LFINTOSC) over the Timer1 oscillator? It depends on whether your application is time-sensitive enough to require the accuracy of a crystal on the Timer1 oscillator or not. If you have a timekeeping application, then you will probably have a 32 kHz crystal oscillator connected to Timer1.

Since Timer1 continues to operate during Sleep, there is no penalty in using Timer1 as the LCD clock source. If you don’t need to use an external oscillator on Timer1, then the internal RC oscillator (INTRC or LFINTOSC) is more than sufficient to use as the clock source for the LCD and it requires no external components.

TIP #10 How to Update LCD Data Through Firmware

To update the LCD, the content of the LCDDATA registers is modified to turn on, or off, each pixel on the LCD display. The application firmware will usually modify buffer variables that are created to correspond with elements on the display, such as character positions, bar graph, battery display, etc.

When the application calls for a display update, the values stored in the buffer variables must be converted to the correct setting of the pixel bits, located in the LCDDATA registers.

For Type-A waveforms, the LCD Data registers may be written any time without ill effect. However, for Type-B waveforms, the LCD Data registers can only be written every other LCD frame in order to ensure that the two frames of the Type-B waveform are compliments of one another. Otherwise, a DC bias can be presented to the LCD.

The LCD Data registers should only be written when a write is allowed, which is indicated by the WA bit in the LCDCON register being set.

On the PIC16C926 parts, there is no WA bit. The writing of the pixel data can be coordinated on an LCD interrupt. The LCD interrupt is only generated when a multiplexed (not static) Type-B waveform is selected.

TIP #11 Blinking LCD

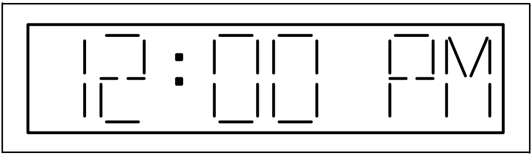
Information can be displayed in more than one way with an LCD panel. For example, how can the user's attention be drawn to a particular portion of the LCD panel? One way that does not require any additional segments is to create a blinking effect.

Look at a common clock application. The ":" between the hours and minutes is commonly made to blink once a second (on for half a second, off for half a second). This shows that the clock is counting in absence of the ticking sound or second hand that accompanies the usual analog face clock. It serves an important purpose of letting the user know that the clock is operating.

If there is a power outage, then it is common for the entire clock display to blink. This gives the user of the clock an immediate indication that the clock is no longer showing the correct time.

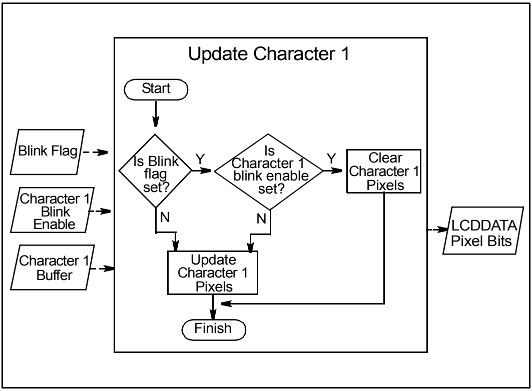
When the user sets the time, then blinking is commonly used to show that a new mode has been entered, such as blinking the hours to identify that the hours are being set, or blinking the minutes to show that the minutes are being set. In a simple clock, blinking is used for several different purposes. Without blinking effects, the common digital clock would not be nearly as user friendly.

FIGURE 11-1: COMMON CLOCK APPLICATION



Fortunately, blinking is quite easy to implement. There are many ways to implement a blinking effect in software. Any regular event can be used to update a blink period counter. A blink flag can be toggled each time the blink period elapses. Each character or display element that you want to blink can be assigned a corresponding blink enable flag. The flowchart for updating the display would look like:

FIGURE 11-2: UPDATING DISPLAY FLOWCHART



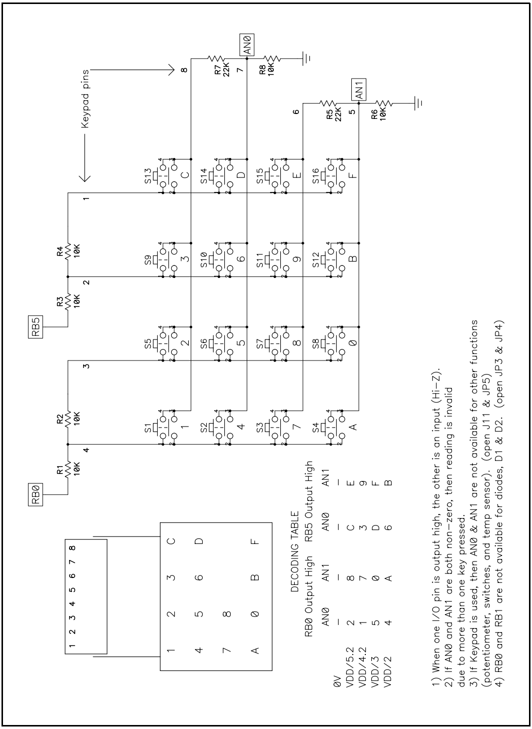
TIP #12 4 x 4 Keypad Interface that Conserves Pins for LCD Segment Drivers

A typical digital interface to a 4 x 4 keypad uses 8 digital I/O pins. But using eight pins as digital I/Os can take away from the number of segment driver pins available to interface to an LCD.

By using 2 digital I/O pins and 2 analog input pins, it is possible to add a 4 x 4 keypad to the PIC microcontroller without sacrificing any of its LCD segment driver pins.

The schematic for keypad hook-up is shown in Figure 12-1. This example uses the PIC18F8490, but the technique could be used on any of the LCD PIC MCUs.

FIGURE 12-1: KEYPAD HOOK-UP SCHEMATIC



The two digital I/O pins that are used are RB0 and RB5, but any two digital I/O pins could work. The two analog pins used are AN0 and AN1.

To read the keypad, follow the steps below:

1. First, make RB0 an output high and RB5 an input (to present a high impedance).
2. Perform two successive A/D conversions, first on AN0, then on AN1.
3. Save the conversion results to their respective variables; for example, `RB0_AN0_Result` and `RB0_AN1_Result`.
4. Next, make RB5 an output high and RB0 an input (to present a high impedance).
5. Perform two successive A/D conversions, first on AN0, then on AN1.
6. Save the conversion results to their respective variables; for example, `RB5_AN0_Result` and `RB5_AN1_Result`.
7. There are now 4 variables that represent a key press in each quadrant of the 4 x 4 keypad:
   - `RB0_AN0_Result` – denotes key press of 1, 2, 4 or 5.
   - `RB0_AN1_Result` – denotes key press of 7, 8, A or 0.
   - `RB5_AN0_Result` – denotes key press of 3, C, 6 or D.
   - `RB5_AN1_Result` – denotes key press of 9, E, B or F.
8. Finally, check each value against the matching column of Table 12-1. If it is within +/-10% of a value, then it can be taken to indicate that the corresponding key has been pressed.

TABLE 12-1: KEYPAD VALUES

Value +/-10%	RB0_AN0	RB0_AN1	RB5_AN0	RB5_AN1
<V_{DD}/10	—	—	—	—
V_{DD}/5.2	2	8	C	E
V_{DD}/4.2	1	7	3	9
V_{DD}/3	5	0	D	F
V_{DD}/2	4	A	6	B

9. This loop should be repeated about once every 20 ms or so.

Don’t forget a debounce routine. For example, require the above steps (with 20 ms delay between) to return the same key value twice in a row for that key to be considered pressed. Also, require a no key press to be returned at least twice before looking for the next key press.

When keys within the same quadrant are pressed simultaneously, voltages other than the four valid levels shown in the table may be generated. These levels can either be ignored, or if you want to use simultaneous key presses to enable certain functions, you can add decoding for those levels as well.

REFERENCE DOCUMENTATION

AN220, "Watt-Hour Meter Using PIC16C923 and CS5460" (DS00220)

AN582, "Low-Power Real-Time Clock" (DS00582)

AN587, Interfacing PIC® MCUs to an LCD Module" (DS00587)

AN649, "Yet Another Clock Featuring the PIC16C924" (DS00649)

AN658, "LCD Fundamentals Using PIC16C92X Microcontrollers" (DS00658)

TB084, "Contrast Control Circuits for the PIC16F91X" (DS91084)

NOTES:

NOTES:

NOTES:

Printed on recycled paper.

Worldwide Sales and Service

AMERICAS

Corporate Office
Tel: 480-792-7200
Technical Support:
http://support.microchip.com

Atlanta
Tel: 678-957-9614

Boston
Tel: 774-760-0087

Chicago
Tel: 630-285-0071

Dallas
Tel: 972-818-7423

Detroit
Tel: 248-538-2250

Kokomo
Tel: 765-864-8360

Los Angeles
Tel: 949-462-9523

Santa Clara
Tel: 408-961-6444

Toronto
Tel: 905-673-0699

ASIA/PACIFIC

Australia
Tel: 61-2-9868-6733

China-Beijing
Tel: 86-10-8528-2100

China-Chengdu
Tel: 86-28-8665-5511

China-Fuzhou
Tel: 86-591-8750-3506

China-Hong Kong SAR
Tel: 852-2401-1200

China-Qingdao
Tel: 86-532-8502-7355

China-Shanghai
Tel: 86-21-5407-5533

China-Shenyang
Tel: 86-24-2334-2829

China-Shenzhen
Tel: 86-755-8203-2660

China-Shunde
Tel: 86-757-2839-5507

China-Wuhan
Tel: 86-27-5980-5300

China-Xian
Tel: 86-29-8833-7250

India-Bangalore
Tel: 91-80-4182-8400

India-New Delhi
Tel: 91-11-4160-8631

India-Pune
Tel: 91-20-2566-1512

Japan-Yokohama
Tel: 81-45-471- 6166

Korea-Gumi
Tel: 82-54-473-4301

Korea-Seoul
Tel: 82-2-554-7200

Malaysia-Penang
Tel: 60-4-646-8870

Philippines-Manila
Tel: 63-2-634-9065

Singapore
Tel: 65-6334-8870

Taiwan-Hsin Chu
Tel: 886-3-572-9526

Taiwan-Kaohsiung
Tel: 886-7-536-4818

Taiwan-Taipei
Tel: 886-2-2500-6610

Thailand-Bangkok
Tel: 66-2-694-1351

EUROPE

Austria-Weis
Tel: 43-7242-2244-39

Denmark-Copenhagen
Tel: 45-4450-2828

France-Paris
Tel: 33-1-69-53-63-20

Germany-Munich
Tel: 49-89-627-144-0

Italy-Milan
Tel: 39-0331-742611

Netherlands-Drunen
Tel: 31-416-690399

Spain-Madrid
Tel: 34-91-708-08-90

UK-Wokingham
Tel: 44-118-921-5869

12/08/06

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Microchip received ISO/TS-16949:2002 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

Microchip Technology Inc.
2355 W. Chandler Blvd. • Chandler, AZ 85224 U.S.A.
Phone: 480-792-7200 • Fax: 480-792-9210
www.microchip.com