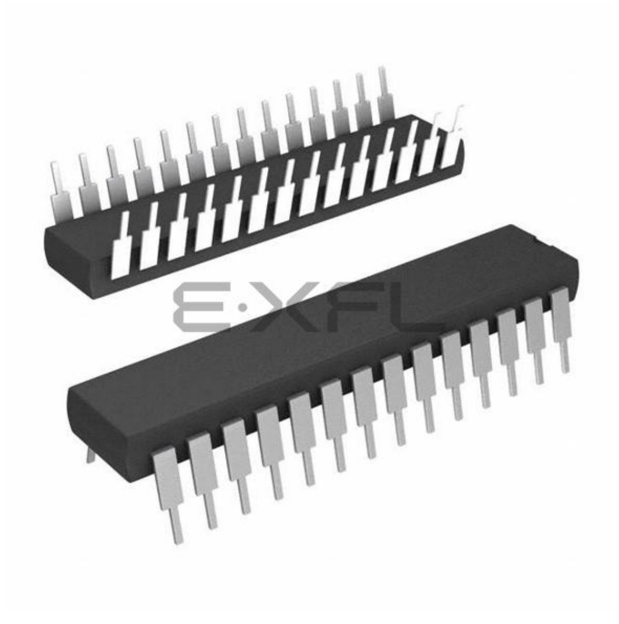


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	4MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	22
Program Memory Size	7KB (4K x 14)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	192 x 8
Voltage - Supply (Vcc/Vdd)	4V ~ 5.5V
Data Converters	-
Oscillator Type	External
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Through Hole
Package / Case	28-DIP (0.300", 7.62mm)
Supplier Device Package	28-SPDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16c63a-04-sp

TIP #3 Configuring Port Pins

All PIC MCUs have bidirectional I/O pins. Some of these pins have analog input capabilities. It is very important to pay attention to the signals applied to these pins so the least amount of power will be consumed.

Unused Port Pins

If a port pin is unused, it may be left unconnected but configured as an output pin driving to either state (high or low), or it may be configured as an input with an external resistor (about 10 kΩ) pulling it to V_{DD} or V_{SS}. If configured as an input, only the pin input leakage current will be drawn through the pin (the same current would flow if the pin was connected directly to V_{DD} or V_{SS}). Both options allow the pin to be used later for either input or output without significant hardware modifications.

Digital Inputs

A digital input pin consumes the least amount of power when the input voltage is near V_{DD} or V_{SS}. If the input voltage is near the midpoint between V_{DD} and V_{SS}, the transistors inside the digital input buffer are biased in a linear region and they will consume a significant amount of current. If such a pin can be configured as an analog input, the digital buffer is turned off, reducing both the pin current as well as the total controller current.

Analog Inputs

Analog inputs have a very high-impedance so they consume very little current. They will consume less current than a digital input if the applied voltage would normally be centered between V_{DD} and V_{SS}. Sometimes it is appropriate and possible to configure digital inputs as analog inputs when the digital input must go to a low power state.

Digital Outputs

There is no additional current consumed by a digital output pin other than the current going through the pin to power the external circuit. Pay close attention to the external circuits to minimize their current consumption.

TIP #4 Use High-Value Pull-Up Resistors

It is more power efficient to use larger pull-up resistors on I/O pins such as MCLR, I^2C™ signals, switches and for resistor dividers. For example, a typical I^2C pull-up is 4.7k. However, when the I^2C is transmitting and pulling a line low, this consumes nearly 700 uA of current for each bus at 3.3V. By increasing the size of the I^2C pull-ups to 10k, this current can be halved. The tradeoff is a lower maximum I^2C bus speed, but this can be a worthwhile trade in for many low power applications. This technique is especially useful in cases where the pull-up can be increased to a very high resistance such as 100k or 1M.

TIP #5 Reduce Operating Voltage

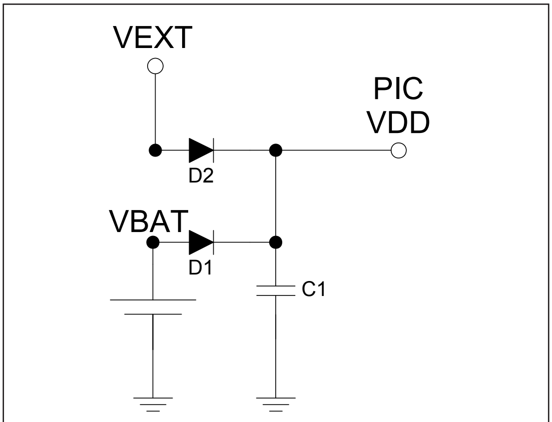
Reducing the operating voltage of the device, V_{DD}, is a useful step to reduce the overall power consumption. When running, power consumption is mainly influenced by the clock speed. When sleeping, the most significant factor is leakage in the transistors. At lower voltages, less charge is required to switch the system clocks and transistors leak less current.

It is important to pay attention to how reducing the operating voltage reduces the maximum allowed operating frequency. Select the optimum voltage that allows the application to run at its maximum speed. Refer to the device data sheet for the maximum operating frequency of the device at the given voltage.

TIP #7 Battery Backup for PIC MCUs

For an application that can operate from either an external supply or a battery backup, it is necessary to be able to switch from one to the other without user intervention. This can be accomplished with battery backup ICs, but it is also possible to implement with a simple diode OR circuit, shown in Figure 7-1. Diode D1 prevents current from flowing into the battery from VEXT when the external power is supplied. D2 prevents current from flowing into any external components from the battery if VEXT is removed. As long as the external source is present and higher voltage than the battery, no current from the battery will be used. When VEXT is removed and the voltage drops below VBAT, the battery will start powering the MCU. Low forward voltage Schottky diodes can be used in order to minimize the voltage dropout from the diodes. Additionally, inputs can be referenced to VEXT and VBAT in order to monitor the voltage levels of the battery and the external supply. This allows the micro to enter lower power modes when the supply is removed or the battery is running low. In order to avoid glitches on V_{DD} caused by the diode turn-on delay when switching supplies, ensure enough decoupling capacitance is used on V_{DD} (C1).

Figure 7-1:



Dynamic Operation Tips n' Tricks

The following tips and tricks apply to methods of improving the dynamic operating current consumption of an application. This allows an application to get processing done quicker which enables it to sleep more and will help reduce the current consumed while processing.

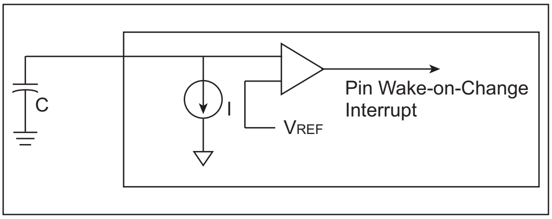
TIP #8 Enhanced PIC16 Mid-Range Core

The Enhanced PIC16 mid-range core has a few features to assist in low power. New instructions allow many applications to execute in less time. This allows the application to spend more time asleep and less time processing and can provide considerable power savings. It is important not to overlook these new instructions when designing with devices that contain the new core. The Timer1 oscillator and WDT have also been improved, now meeting nanoWatt XLP requirements and drawing much less current than in previous devices.

TIP #22 Ultra Low-Power Wake-Up Peripheral

Newer devices have a modification to PORTA that creates an Ultra Low-Power Wake-Up (ULPWU) peripheral. A small current sink and a comparator have been added that allows an external capacitor to be used as a wake-up timer. This feature provides a low-power periodic wake-up source which is dependent on the discharge time of the external RC circuit.

Figure 22-1: Ultra Low-Power Wake-Up Peripheral

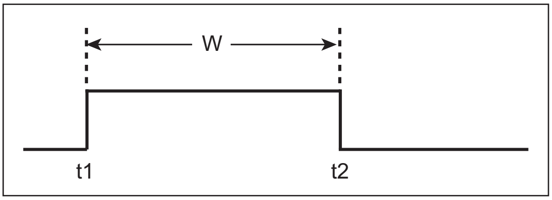


If the accuracy of the Watchdog Timer is not required, this peripheral can save a lot of current.

Visit the low power design center at: www.microchip.com/lowpower for additional design resources.

TIP #3 Measuring Pulse Width

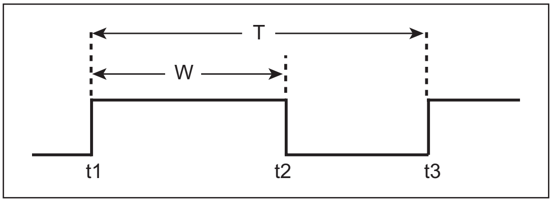
Figure 3-1: Pulse Width



1. Configure control bits CCPxM3:CCPxM0 (CCPxCON<3:0>) to capture every rising edge of the waveform.
2. Configure Timer1 prescaler so that Timer1 will run W_{MAX} without overflowing.
3. Enable the CCP interrupt (CCPxIE bit).
4. When CCP interrupt occurs, save the captured timer value (t1) and reconfigure control bits to capture every falling edge.
5. When CCP interrupt occurs again, subtract saved value (t1) from current captured value (t2) – this result is the pulse width (W).
6. Reconfigure control bits to capture the next rising edge and start process all over again (repeat steps 3 through 6).

TIP #4 Measuring Duty Cycle

Figure 4-1: Duty Cycle



The duty cycle of a waveform is the ratio between the width of a pulse (W) and the period (T). Acceleration sensors, for example, vary the duty cycle of their outputs based on the acceleration acting on a system. The CCP module, configured in Capture mode, can be used to measure the duty cycle of these types of sensors. Here's how:

1. Configure control bits CCPxM3:CCPxM0 (CCPxCON<3:0>) to capture every rising edge of the waveform.
2. Configure Timer1 prescaler so that Timer1 will run T_{MAX}(1) without overflowing.
3. Enable the CCP interrupt (CCPxIE bit).
4. When CCP interrupt occurs, save the captured timer value (t1) and reconfigure control bits to capture every falling edge.

Note 1:	T_{MAX} is the maximum pulse period that will occur.

5. When the CCP interrupt occurs again, subtract saved value (t1) from current captured value (t2) – this result is the pulse width (W).
6. Reconfigure control bits to capture the next rising edge.
7. When the CCP interrupt occurs, subtract saved value (t1) from the current captured value (t3) – this is the period (T) of the waveform.
8. Divide T by W – this result is the Duty Cycle.
9. Repeat steps 4 through 8.

TIP #7 Periodic Interrupts

Generating interrupts at periodic intervals is a useful technique implemented in many applications. This technique allows the main loop code to run continuously, and then, at periodic intervals, jump to the interrupt service routine to execute specific tasks (i.e., read the ADC). Normally, a timer overflow interrupt is adequate for generating the periodic interrupt. However, sometimes it is necessary to interrupt at intervals that can not be achieved with a timer overflow interrupt. The CCP configured in Compare mode makes this possible by shortening the full 16-bit time period.

Example Problem:

A PIC16F684 running on its 8 MHz internal oscillator needs to be configured so that it updates a LCD exactly 5 times every second.

Step #1: Determine a Timer1 prescaler that allows an overflow at greater than 0.2 seconds

a) Timer1 overflows at: Tosc*4*65536* prescaler

b) For a prescaler of 1:1, Timer1 overflows in 32.8 ms.

c) A prescaler of 8 will cause an overflow at a time greater than 0.2 seconds.
8 x 32.8 ms = 0.25s

Step #2: Calculate CCPR1 (CCPR1L and CCPR1H) to shorten the time-out to exactly 0.2 seconds

a) CCPR1 = Interval Time/(Tosc*4*prescaler) = 0.2/(125 ns*4*8) = 5000 = 0xC350

b) Therefore, CCPR1L = 0x50, and CCPR1H = 0xC3

Step #3: Configuring CCP1CON

The CCP module should be configured in Trigger Special Event mode. This mode generates an interrupt when the Timer1 equals the value specified in CCPR1L and Timer1 is automatically cleared(1). For this mode, CCP1CON = ‘`b00001011`’.

Note 1:	Trigger Special Event mode also starts an A/D conversion if the A/D module is enabled. If this functionality is not desired, the CCP module should be configured in “generate software interrupt-on-match only” mode (i.e., CCP1CON = `b`‘`00001010`’). Timer 1 must also be cleared manually during the CCP interrupt.

TIP #9 Generating the Time Tick for a RTOS

Real Time Operating Systems (RTOS) require a periodic interrupt to operate. This periodic interrupt, or "tick rate", is the basis for the scheduling system that RTOS's employ. For instance, if a 2 ms tick is used, the RTOS will schedule its tasks to be executed at multiples of the 2 ms. A RTOS also assigns a priority to each task, ensuring that the most critical tasks are executed first. Table 9-1 shows an example list of tasks, the priority of each task and the time interval that the tasks need to be executed.

Table 9-1: Tasks

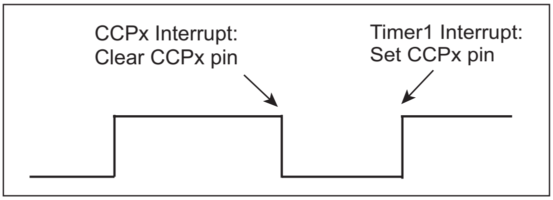
Task	Interval	Priority
Read ADC Input 1	20 ms	2
Read ADC Input 2	60 ms	1
Update LCD	24 ms	2
Update LED Array	36 ms	3
Read Switch	10 ms	1
Dump Data to Serial Port	240 ms	1

The techniques described in Tip #7 can be used to generate the 2 ms periodic interrupt using the CCP module configured in Compare mode.

> **Note:** For more information on RTOSs and their use, see Application Note AN777 "*Multitasking on the PIC16F877 with the Salvo™ RTOS*".

TIP #10 16-Bit Resolution PWM

Figure 10-1: 16-Bit Resolution PWM



1. Configure CCPx to clear output (CCPx pin) on match in Compare mode (CCPxCON <CCPSM3:CCPxM0>).
2. Enable the Timer1 interrupt.
3. Set the period of the waveform via Timer1 prescaler (T1CON <5:4>).
4. Set the duty cycle of the waveform using CCPRxL and CCPRxH.
5. Set CCPx pin when servicing the Timer1 overflow interrupt(1).

> **Note 1:** One hundred percent duty cycle is not achievable with this implementation due to the interrupt latency in servicing Timer1. The period is not affected because the interrupt latency will be the same from period to period as long as the Timer1 interrupt is serviced first in the ISR.

Timer1 has four configurable prescaler values. These are 1:1, 1:2, 1:4 and 1:8. The frequency possibilities of the PWM described above are determined by Equation 10-1.

Equation 10-1

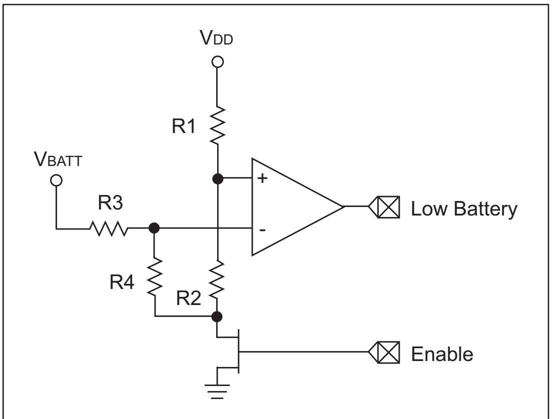
$$F_{PWM} = F_{OSC}/(65536*4*prescaler)$$

For a microcontroller running on a 20 MHz oscillator (Fosc) this equates to frequencies of 76.3 Hz, 38.1 Hz, 19.1 Hz and 9.5 Hz for increasing prescaler values.

TIP #1 Low Battery Detection

When operating from a battery power supply, it is important for a circuit to be able to determine when the battery charge is insufficient for normal operation of the circuit. Typically, this is a comparator-based circuit similar to the Programmable Low Voltage Detect (PLVD) peripheral. If the PLVD peripheral is not available in the microcontroller, a similar circuit can be constructed using a comparator and a few external components (see Figure 1-1 and Figure 1-2). The circuit in Figure 1-1 assumes that the microcontroller is operating from a regulated supply voltage. The circuit in Figure 1-2 assumes that the microcontroller supply is unregulated.

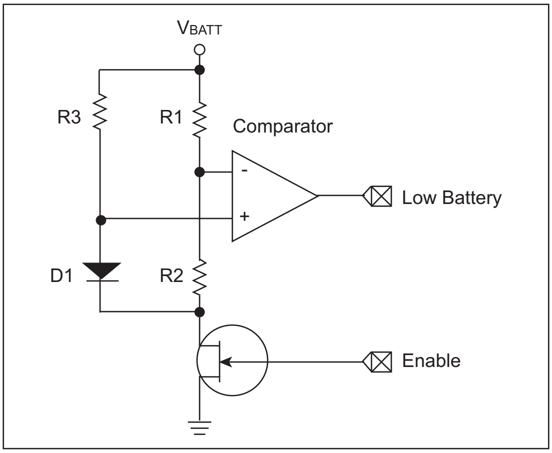
Figure 1-1: Regulated Supply



The comparator will trip when the battery voltage, V_{BATT} = 5.7V: R1 = 33k, R2 = 10k, R3 = 39k, R4 = 10k, V_{DD} = 5V.

In Figure 1-1, resistors R1 and R2 are chosen to place the voltage at the non-inverting input at approximately 25% of V_{DD}. R3 and R4 are chosen to set the inverting input voltage equal to the non-inverting input voltage when the battery voltage is equal to the minimum operating voltage for the system.

Figure 1-2: Unregulated Supply



Comparator will trip when V_{BATT} = 3V: R1 = 33k, R2 = 10k and R3 = 470Ω.

In Figure 1-2, resistor R3 is chosen to bias diode D1 above its forward voltage when V_{BATT} is equal to the minimum battery voltage for the system. Resistors R1 and R2 are chosen to set the inverting input voltage equal to the forward voltage of D1.

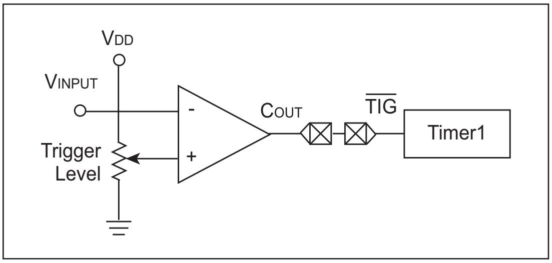
TIP #4 Pulse Width Measurement

To measure the high or low pulse width of an incoming analog signal, the comparator can be combined with Timer1 and the Timer1 Gate input option (see Figure 4-1). Timer1 Gate acts as a count enable for Timer1. If the input is low, Timer1 will count. If the $\overline{\text{T1G}}$ input is high, Timer1 does not count. Combining $\overline{\text{T1G}}$ with the comparator allows the designer to measure the time between a high-to-low output change and a low-to-high output change.

To make a measurement between a low-to-high and a high-to-low transition, the only change required is to set the CINV bit in the comparator CMCON register which inverts the comparator output.

Because the output of the comparator can change asynchronously with the Timer1 clock, only comparators with the ability to synchronize their output with the Timer1 clock should be used and their C2SYNC bits should be set.

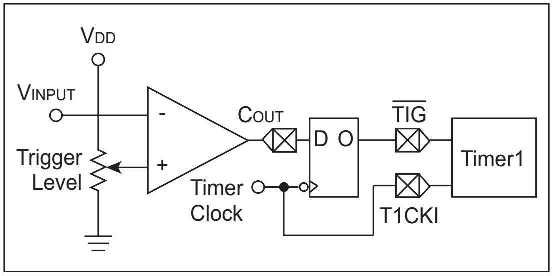
Figure 4-1: Comparator with Timer1 and $\overline{\text{T1G}}$



If the on-chip comparator does not have the ability to synchronize its output to the Timer1 clock, the output can be synchronized externally using a discrete D flip-flop (see Figure 4-2).

Note:	The flip-flop must be falling edge triggered to prevent a race condition.

Figure 4-2: Externally Synchronized Comparator

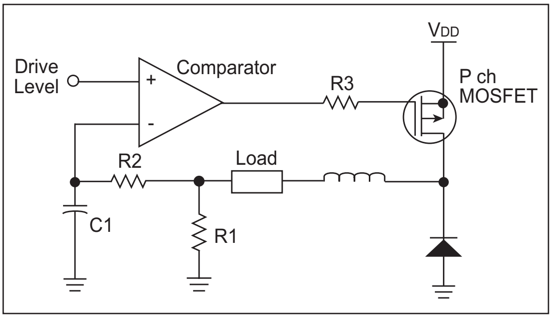


TIP #13 PWM High-Current Driver

This tip combines a comparator with a MOSFET transistor and an inductor to create a switch mode high-current driver circuit. (See Figure 13-1).

The operation of the circuit begins with the MOSFET off and no current flowing in the inductor and load. With the sense voltage across R1 equal to zero and a DC voltage present at the drive level input, the output of the comparator goes low. The low output turns on the MOSFET and a ramping current builds through the MOSFET, inductor, load and R1.

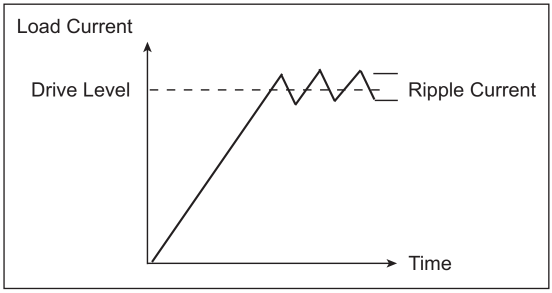
Figure 13-1: High Current Driver



When the current ramps high enough to generate a voltage across R1 equal to the drive level, the comparator output goes high turning off the MOSFET. The voltage at the junction of the MOSFET and the inductor then drops until D1 forward biases. The current continues ramping down from its peak level toward zero. When the voltage across the sense resistor R1 drops below the drive level, the comparator output goes low, the MOSFET turns on, and the cycle starts over.

R2 and C1 form a time delay network that limits the switching speed of the driver and causes it to slightly overshoot and undershoot the drive level when operating. The limit is necessary to keep the switching speed low, so the MOSFET switches efficiently. If R2 and C1 were not present, the system would run at a speed set by the comparator propagation delay and the switching speed of the MOSFET. At that speed, the switching time of the MOSFET would be a significant portion of the switching time and the switching efficiency of the MOSFET would be too low.

Figure 13-1: Current Through the Load



To design a PWM high current driver, first determine a switching speed (F_{SWX}) that is appropriate for the system. Next, choose a MOSFET and D1 capable of handling the load current requirements. Then choose values for R2 and C1 using Equation 13-1.

Equation 13-1

$$F_{SWX} = \frac{2}{R2 * C1}$$

Next determine the maximum ripple current that the load will tolerate, and calculate the required inductance value for L1 using Equation 13-2.

Equation 13-2

$$L = \frac{V_{DD} - V_{LOAD}}{I_{RIPPLE} * F_{SWX} * 2}$$

Finally, choose a value for R1 that will produce a feedback ripple voltage of 100 mV for the maximum ripple current I_{RIPPLE}.

Example:

- F_{SWX} = 10 kHz, R2 = 22k, C1 = .01 µF
- I_{RIPPLE} = 100 mA, V_{DD} = 12V, V_L = 3.5V
- L = 4.25 mH

TIP #18 Logic: OR/NOR Gate

This tip shows the use of the comparator to implement an OR gate, and its complement, the NOR gate.

Resistors R1 and R2 drive the non-inverting input of the comparator with 1/3 V_{DD}. Resistors R3 and R4 average the voltages of the inputs A and B at the inverting input. If either A or B is high, the average voltage is 1/2 V_{DD} and the output of the comparator is high. Only if both A and B are low does the average voltage at the non-inverting input drop below 1/3 the supply voltage, causing the comparator output to go low. The operation of the NOR gate is identical to the OR gate, except the output is inverted due to the swap of the inverting and non-inverting inputs.

> **Note:** Typical propagation delay for the circuit is 250-350 ns using the typical on-chip comparator peripheral of a microcontroller. Delay measurements were made with 10k resistance values.

While the circuit is fairly simple, there are a few requirements for correct operation:

1. The inputs A and B must drive from ground to V_{DD} for the circuit to operate properly.
2. The combination of R1 and R2 will draw current constantly, so they must be kept large to minimize current draw.
3. All resistances on the inverting input react with the input capacitance of the comparator, so the speed of the gate will be affected by the source resistance of A and B, as well as the size of resistors R3 and R4.
4. Resistor R1 must be 2 x R2.
5. Resistor R3 must be equal to R4.

Figure 18-1: OR Gate

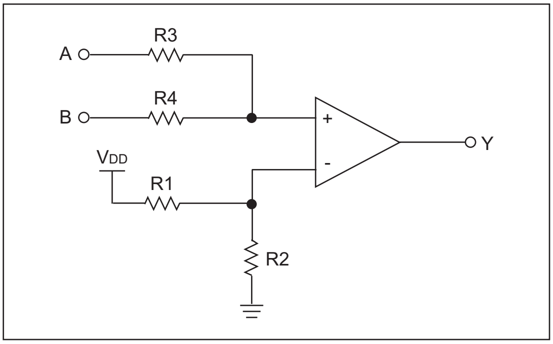
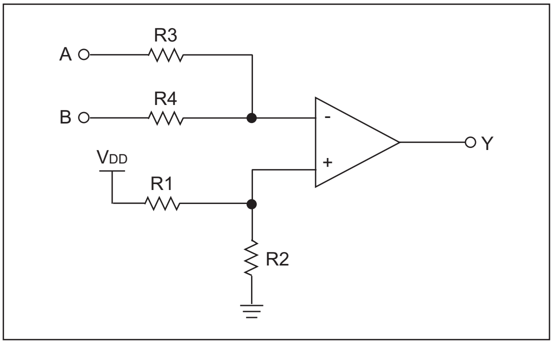


Figure 18-2: NOR Gate



Example:

- V_{DD} = 5V, R3 = R4 = 10k
- R1 = 10k, R2 = 5.1k

TIP #4 Drive Software

Pulse-Width Modulation (PWM) Algorithms

Pulse-Width Modulation is critical to modern digital motor controls. By adjusting the pulse width, the speed of a motor can be efficiently controlled without larger linear power stages. Some PIC devices and all dsPIC DSCs have hardware PWM modules on them. These modules are built into the Capture/Compare/PWM (CCP) peripheral. CCP peripherals are intended for a single PWM output, while the Enhanced CCP (ECCP) is designed to produce the complete H-Bridge output for bidirectional Brushed DC motor control. If cost is a critical design point, a PIC device with a CCP module may not be available, so software generated PWM is a good alternative.

The following algorithms are designed to efficiently produce an 8-bit PWM output on the Mid-Range family of PIC microcontrollers. These algorithms are implemented as macros. If you want these macros to be a subroutine in your program, simply remove the macro statements and replace them with a label and a return statement.

Example 4-1: 1 Output 8-Bit PWM

```
pwm_counter equ xxx ;variable
pwm         equ xxx ;variable

set_pwm macro A     ;sets the pwm
                    ;setpoint to the
                    ;value A
 MOVLW A
 MOVWF pwm
 endm

update_PWM macro    ;performs one update
                    ;of the PWM signal
                    ;place the PWM output
                    ;pin at bit 0 or 7 of
                    ;the port
 MOVF pwm_counter,w
 SUBWF pwm, w       ;if the output
                    ;is on bit 0
 RLF        PORTC,f ;replace PORTC with
                    ;the correct port if
                    ;the output is on bit
                    ;7 of the port
                    ;replace the rlf with
                    ;rrf incf
                    ;pwm_counter,f
```

Example 4-2: 8 Output 8-Bit PWM

```
pwm_counter equ xxx     ;variable
pwm0        equ xxx     ;
pwm1        equ pwm0+1
pwm2        equ pwm1+1
pwm3        equ pwm2+1
pwm4        equ pwm3+1
pwm5        equ pwm4+1
pwm6        equ pwm5+1
pwm7        equ pwm6+1
output      equ pwm7+1
set_pwm macro A,b       ;sets pwm b with
                        ;the value A
 MOVLW pwm0
 ADDLW b
 MOVWF fsr
 MOVLW a
 MOVWF indf
 endm

update_PWM macro        ;peforms one
                        ;update of all 8
                        ;PWM signals
                        ;all PWM signals
                        ;must be on the
                        ;same port
 MOVF       pwm_counter,w
 SUBWF      pwm0,w
 RLF        output,f
 MOVF       pwm_counter,w
 SUBWF      pwm1,w
 RLF        output,f
 MOVF       pwm_counter,w
 SUBWF      pwm2,w
 RLF        output,f
 MOVF       pwm_counter,w
 SUBWF      pwm3,w
 RLF        output,f
 MOVF       pwm_counter,w
 SUBWF      pwm4,w
 RLF        output,f
 MOVF       pwm_counter,w
 SUBWF      pwm5,w
 RLF        output,f
 MOVF       pwm_counter,w
 SUBWF      pwm6,w
 RLF        output,f
 MOVF       pwm_counter,w
 SUBWF      pwm7,w
 RLF        output,w
 MOVWF      PORTC
 INCF       pwm_counter,f
endm
```


TIP #5 Writing a PWM Value to the the CCP Registers With a Mid-Range PIC® Microcontroller

The two PWM LSb’s are located in the CCPCON register of the CCP. This can make changing the PWM period frustrating for a developer. Example 5-1 through Example 5-3 show three macros written for the mid-range product family that can be used to set the PWM period. The first macro takes a 16-bit value and uses the 10 MSb’s to set the PWM period. The second macro takes a 16-bit value and uses the 10 LSb’s to set the PWM period. The last macro takes 8 bits and sets the PWM period. This assumes that the CCP is configured for no more than 8 bits.

Example 5-1: Left Justified 16-Bit Macro

```
pwm_tmp    equ xxx  ;this variable must be
                    ;allocated someplace
setPeriod  macro a  ;a is 2 SFR’s in
                    ;Low:High arrangement
                    ;the 10 MSb’s are the
                    ;desired PWM value
 RRF       a,w      ;This macro will
                    ;change w
 MOVWF     pwm_tmp
 RRF       pwm_tmp,w
 ANDLW     0x30
 IORLW     0x0F
 MOVWF     CCP1CON
 MOVF      a+1,w
 MOVWF     CCPR1L
```

Example 5-2: Right Justified 16-Bit Macro

```
pwm_tmp    equ xxx  ;this variable must be
                    ;allocated someplace
setPeriod  macro a  ;a is 2 bytes in
                    ;Low:High arrangement
                    ;the 10 LSb’s are the
                    ;desired PWM value
 SWAPF     a,w      ;This macro will
                    ;change w
 ANDLW     0x30
 IORLW     0x0F
 MOVWF     CCP1CON
 RLF       a,w
 IORLW     0x0F
 MOVWF     pwm_tmp
 RRF       pwm_tmp,f
 RRF       pwm_tmp,w
 MOVWF     CCPR1L
```

Example 5-3: 8-Bit Macro

```
pwm_tmp    equ xxx  ;this variable must be
                    ;allocated someplace
setPeriod  macro a  ;a is 1 SFR
 SWAPF     a,w      ;This macro will
                    ;change w
 ANDLW     0x30
 IORLW     0x0F
 MOVWF     CCP1CON
 RRF       a,w
 MOVWF     pwm_tmp
 RRF       pwm_tmp,w
 MOVWF     CCPR1L
```


TIP #6 Current Sensing

The torque of an electric motor can be monitored and controlled by keeping track of the current flowing through the motor. Torque is directly proportional to the current. Current can be sensed by measuring the voltage drop through a known value resistor or by measuring the magnetic field strength of a known value inductor. Current is generally sensed at one of two places, the supply side of the drive circuit (high side current sense) or the sink side of the drive circuit (low side current sense). Low side sensing is much simpler but the motor will no longer be grounded, causing a safety issue in some applications. High side current sensing generally requires a differential amplifier with a common mode voltage range within the voltage of the supply.

Figure 6-1: Resistive High Side Current Sensing

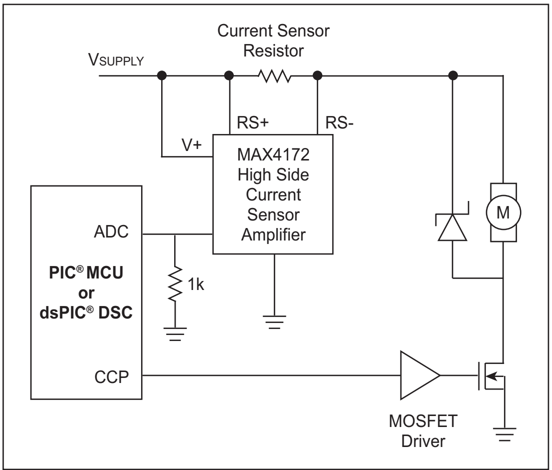
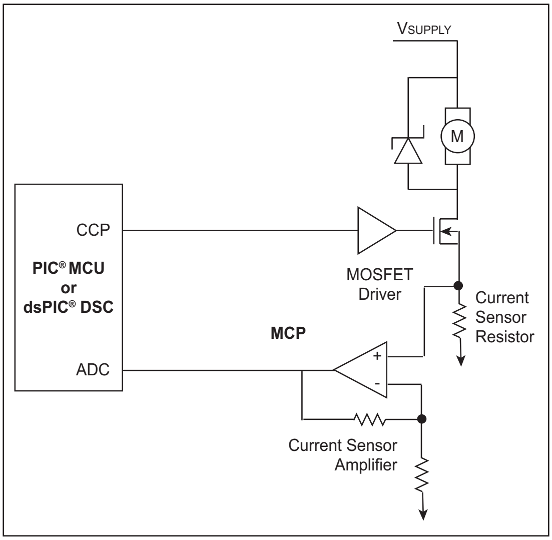


Figure 6-2: Resistive Low Side Current Sensing



Current measurement can also be accomplished using a Hall effect sensor to measure the magnetic field surrounding a current carrying wire. Naturally, this Hall effect sensor can be located on the high side or the low side of the load. The actual location of the sensor does not matter because the sensor does not rely upon the voltage on the wire. This is a non-intrusive method that can be used to measure motor current.

Figure 6-3: Magnetic Current Sensing

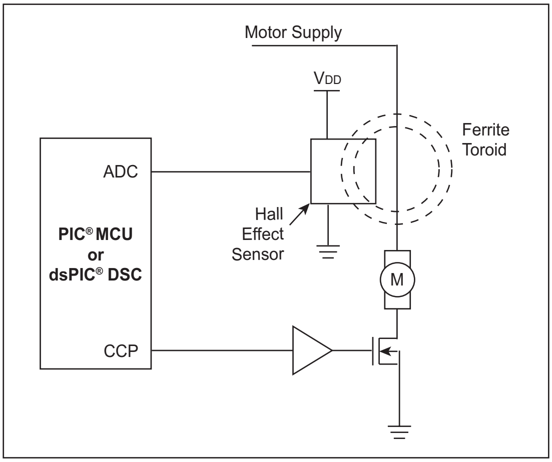
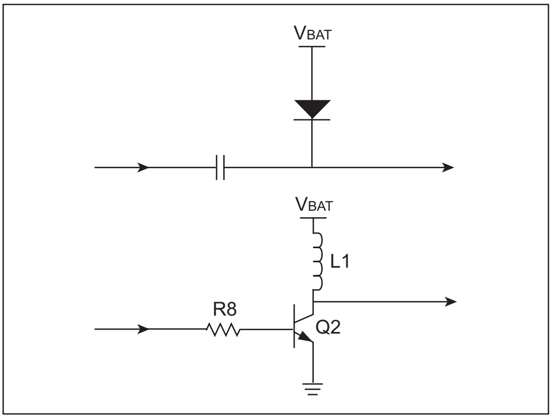


Figure 5-2: Two Types of Boost Converter

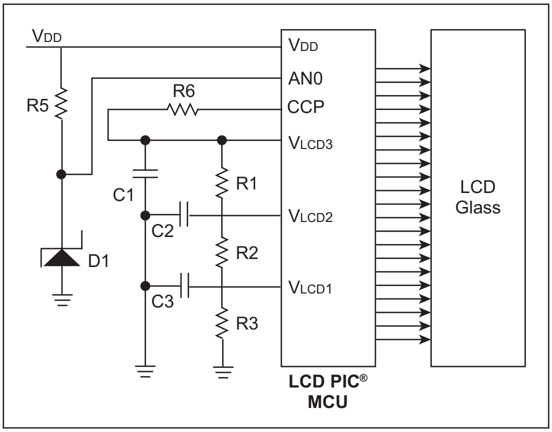


The two methods of producing a boost converter are shown above. The first circuit is simply a switched capacitor type circuit. The second circuit is a standard inductor boost circuit. These circuits work by raising V_{DD}. This allows the voltage at V_{LCD} to exceed V_{DD}.

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 V_{DD} supply. The PWM will then be used to compensate for variations in the supply voltage.

Figure 6-1: Software Controlled Voltage Generator



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	(FOSC/4)/8192	No
PIC18F8X90	T1OSC	Yes
PIC18F6XJ90	INTRC/32	Yes
PIC18F8XJ90		

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!

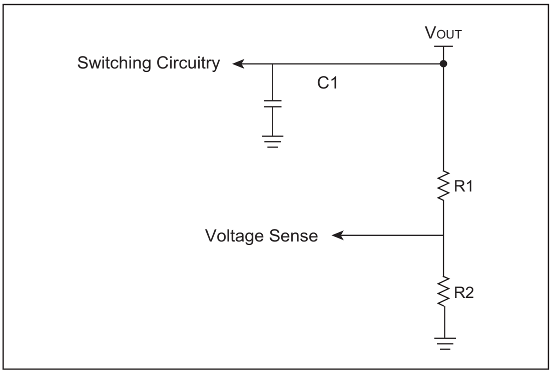
When should you select the internal RC oscillator (or LFINTOSC) over the Timer1 oscillator? It depends on whether your application is time-sensitive enough to require the accuracy of a crystal on the Timer1 oscillator or not. If you have a timekeeping application, then you will probably have a 32 kHz 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 #21 Using Output Voltage Monitoring to Create a Self-Calibration Function

A PIC microcontroller can be used to create a switching power supply controlled by a PID loop (as described in Tip #16). This type of power supply senses its output voltage digitally, compares that voltage to the desired reference voltage and makes duty cycle changes accordingly. Without calibration, it is sensitive to component tolerances.

Figure 21-1: Typical Power Supply Output Stage



The output stage of many power supplies is similar to Figure 21-1. R1 and R2 are used to set the ratio of the voltage that is sensed and compared to the reference.

A simple means of calibrating this type of power supply is as follows:

1. Supply a known reference voltage to the output of the supply.
2. Place the supply in Calibration mode and allow it to sense that reference voltage.

By providing the supply with the output voltage that it is to produce, it can then sense the voltage across the resistor divider and store the sensed value. Regardless of resistor tolerances, the sensed value will always correspond to the proper output value for that particular supply.

Futhermore, this setup could be combined with Tip #20 to calibrate at several temperatures.

This setup could also be used to create a programmable power supply by changing the supplied reference and the resistor divider for voltage feedback.

TIPS ‘N TRICKS INTRODUCTION

Overview - the 3.3 Volt to 5 Volt Connection

One of the by-products of our ever increasing need for processing speed is the steady reduction in the size of the transistors used to build microcontrollers. Up-integration at cheaper cost also drives the need for smaller geometries. With reduced size comes a reduction in the transistor breakdown voltage, and ultimately, a reduction in the supply voltage when the breakdown voltage falls below the supply voltage. So, as speeds increase and complexity mounts, it is an inevitable consequence that the supply voltages would drop from 5V to 3.3V, or even 1.8V for high density devices.

Microchip microcontrollers have reached a sufficient level of speed and complexity that they too are making the transition to sub-5V supply voltages. The challenge is that most of the interface circuitry is still designed for 5V supplies. This means that, as designers, we now face the task of interfacing 3.3V and 5V systems. Further, the task includes not only logic level translation, but also powering the 3.3V systems and translating analog signals across the 3.3V/5V barrier.

This Tips ‘n Tricks book addresses these challenges with a collection of power supply building blocks, digital level translation blocks and even analog translation blocks. Throughout the book, multiple options are presented for each of the transitions, spanning the range from all-in-one interface devices, to low-cost discrete solutions. In short, all the blocks a designer is likely to need for handling the 3.3V challenge, whether the driving force is complexity, cost or size.

Additional information can be found on the Microchip web site at www.microchip.com/3volts.

> **Note:** The tips ‘n tricks presented here assume a 3.3V supply. However, the techniques work equally well for other supply voltages with the appropriate modifications.

Power Supplies

One of the first 3.3V challenges is generating the 3.3V supply voltage. Given that we are discussing interfacing 5V systems to 3.3V systems, we can assume that we have a stable 5 V_{DC} supply. This section will present voltage regulator solutions designed for the 5V to 3.3V transition. A design with only modest current requirements may use a simple linear regulator. Higher current needs may dictate a switching regulator solution. Cost sensitive applications may need the simplicity of a discrete diode regulator. Examples from each of these areas are included here, with the necessary support information to adapt to a wide variety of end applications.

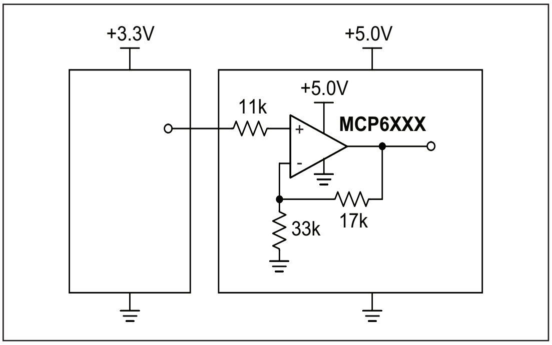
Table 1: Power Supply Comparisons

Method	V_{REG}	I_Q	Eff.	Size	Cost	Transient Response
Zener Shun Reg.	10% Typ	5 mA	60%	Sm	Low	Poor
Series Linear Reg.	0.4% Typ	1 µA to 100 µA	60%	Sm	Med	Excellent
Switching Buck Reg.	0.4% Typ	30 µA to 2 mA	93%	Med to Lrg	High	Good

TIP #14 3.3V → 5V Analog Gain Block

To scale analog voltage up when going from 3.3V supply to 5V supply. The 33 kΩ and 17 kΩ set the op amp gain so that the full scale range is used in both sides. The 11 kΩ resistor limits current back to the 3.3V circuitry.

Figure 14-1: Analog Gain Block

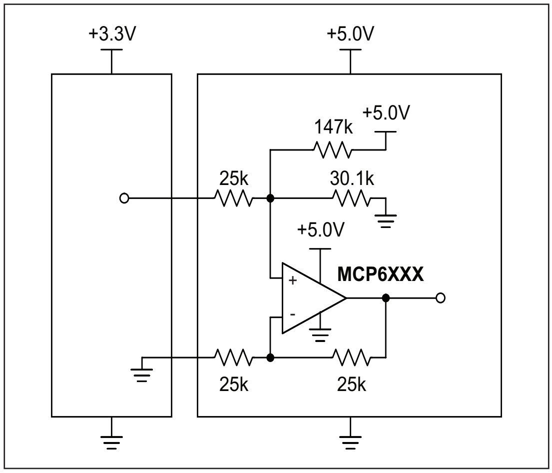


TIP #15 3.3V → 5V Analog Offset Block

Offsetting an analog voltage for translation between 3.3V and 5V.

Shift an analog voltage from 3.3V supply to 5V supply. The 147 kΩ and 30.1 kΩ resistors on the top right and the +5V supply voltage are equivalent to a 0.85V voltage source in series with a 25 kΩ resistor. This equivalent 25 kΩ resistance, the three 25 kΩ resistors, and the op amp form a difference amplifier with a gain of 1 V/V. The 0.85V equivalent voltage source shifts any signal seen at the input up by the same amount; signals centered at 3.3V/2 = 1.65V will also be centered at 5.0V/2 = 2.50V. The top left resistor limits current from the 5V circuitry.

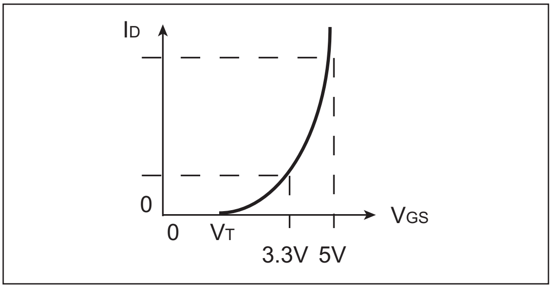
Figure 15-1: Analog Offset Block



TIP #19 Driving N-Channel MOSFET Transistors

Care must be taken when selecting an external N-Channel MOSFET for use with a 3.3V microcontroller. The MOSFET gate threshold voltage is an indication of the device’s capability to completely saturate. For 3.3V applications, select MOSFETs that have an ON resistance rating for gate drive of 3V or less. For example, a FET that is rated for 250 µA of drain current with 1V applied from gate-to-source is not necessarily going to deliver satisfactory results for 100 mA load with a 3.3V drive. When switching from 5V to 3V technology, review the gate-to-source threshold and ON resistance characteristics very carefully as shown in Figure 19-1. A small decrease in gate drive voltage can significantly reduce drain current.

Figure 19-1: Drain Current Capability Versus Gate to Source Voltage



Low threshold devices commonly exist for MOSFETs with drain-to-source voltages rated below 30V. MOSFETs with drain-to-source voltages above 30V typically have higher gate thresholds (VT).

Table 19-1: R_{DS}(ON) and V_{GS}(th) Specifications for IRF7467

R_{DS}(on)	Static Drain-to-Source On-Resistance	–	9.4	12	mΩ	V_{GS} = 10V, I_D = 11A
		–	10.6	13.5		V_{GS} = 4.5V, I_D = 9.0A
		–	17	35		V_{GS} = 2.8V, I_D = 5.5A
V_{GS}(th)	Gate Threshold Voltage	0.6	–	2.0	V	V_{DS} = V_{GS}, I_D = 250 µA

As shown in Table 19-1, the threshold voltage for this 30V, N-Channel MOSFET switch is 0.6V. The resistance rating for this MOSFET is 35 mΩ with 2.8V applied gate, as a result, this device is well suited for 3.3V applications.

Table 19-2: R_{DS}(ON) and V_{GS}(th) Specifications for IRF7201

R_{DS}(on)	Static Drain-to-Source On-Resistance	–	–	0.030	Ω	V_{GS} = 10V, I_D = 7.3A
		–	–	0.050		V_{GS} = 4.5V, I_D = 3.7A
V_{GS}(th)	Gate Threshold Voltage	1.0	–	–	V	V_{DS} = V_{GS}, I_D = 250 µA

For the IRF7201 data sheet specifications, the gate threshold voltage is specified as a 1.0V minimum. This does not mean the device can be used to switch current with a 1.0V gate-to-source voltage as there is no RDS(ON) specification for V_{GS}(th) values below 4.5V. This device is not recommended for 3.3V drive applications that require low switch resistance but can be used for 5V drive applications.

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:
http://support.microchip.com
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Kokomo
Kokomo, IN
Tel: 765-864-8360
Fax: 765-864-8387

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Hong Kong SAR
Tel: 852-2401-1200
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4080

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Yokohama
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-6578-300
Fax: 886-3-6578-370

Taiwan - Kaohsiung
Tel: 886-7-536-4818
Fax: 886-7-536-4803

Taiwan - Taipei
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

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

UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

03/26/09