INTRODUCTION
Welcome to the PIC24FJ128GC010 Analog Design Guide. The PIC24FJ128GC010 family combines a high-speed, 16-bit PIC24 MCU with analog peripherals. The “PIC24FJ128GC010 Family Data Sheet” (DS30009312) is located on the Microchip web site: www.microchip.com/pic24fj128gc010. Along with digital blocks, such as USB and a Liquid Crystal Display (LCD) Controller, the PIC24FJ128GC010 has a 12-bit, 10 Mspips Pipeline ADC, a 16-bit Sigma-Delta ADC, plus two op amps and a 2-channel, 10-bit Digital-to-Analog Converter (DAC).

This Design Guide highlights the different types of applications ideally suited for the PIC24FJ128GC010 family of intelligent analog microcontrollers. The design examples are a starting point reference and are not intended to represent a finished product. Some of the applications shown here will require agency approval (FDA, UL, CE or others) and they are not fully tested to meet these approvals.

The following section will discuss the features of each analog module in the PIC24FJ128GC010. Detailed descriptions are found in each module’s FRM section in the “dsPIC33/PIC24 Family Reference Manual”.

ANALOG MODULES IN THE PIC24FJ128GC010

16-Bit Sigma-Delta A/D Converter (SD_ADC) FRM (DS30687)
The SD_ADC is a high-accuracy, high-resolution ADC for signals in the DC-4 kHz range. The converter has an inherent, low-pass filter characteristic ideal for lower bandwidth sources, such as AC power line monitoring, temperature probes, weight scales, pressure transducers and bridge measurements. There are two input channels routed to a fully differential PGA (Programmable Gain Amplifier), which greatly simplifies measurements in single supply, high common-mode conditions. In most cases, a one-pole filter is sufficient for anti-aliasing.

Dither may improve AC measurements when using the SD_ADC.(1) There are three dithering levels selectable in the SD1CON1 Special Function Register (SFR). Dither has the effect of spreading out the spectrum of the input signal, which reduces spectrum peaks, and thereby, improving overall SNR (Signal-to-Noise Ratio). The adding of low-level noise to the input signal will counter the effect of the quantization noise inherent in all ADCs.(2)

Sigma-Delta ADCs have higher gain and offset errors which must be compensated. The PIC24FJ128GC010 has special measurement modes to measure the gain and offset errors, which are then used in a simple inline calculation by the MCU firmware to produce the final ADC reading. See the “Basic Building Blocks” section for a detailed firmware example using the gain and offset calculations.

12-Bit, High-Speed Pipeline A/D Converter FRM (DS30686)
The pipeline ADC is a very high-speed, 12-bit ADC that can achieve 10 million conversions a second. The PIC24FJ128GC010 has 50 individual input channels that can be either capacitive touch pads or standard voltage inputs. Due to the higher noise floor, samples must be averaged to get the best results.

Operational Amplifier (Op Amp) (DS30505)
There are two independent op amp modules that have programmable speed/power settings. The Power-on Reset (POR) state is the low-power/low-speed configuration. The op amps have very low input bias current (<10 pA), but the input offset voltage may need compensation in the design.

10-Bit Digital-to-Analog Converter (DAC) (DS39615)
The DAC module has two independent voltage outputs. There are several options to select the DAC’s reference voltage. The DACs are very useful for DC biasing the analog ‘front end’ of sensor amplifiers/filters, and can be used for tone generation and Adaptive Differential Pulse Code Modulation (ADPCM) speech playback in products. See Application Note: AN643, “Adaptive Differential Pulse Code Modulation Using PIC® Microcontrollers”(11) for more information.
Other Useful Modules

In addition to the above modules, the PIC24FJ128GC010 has internal band gap voltage references, a Charge Time Measurement Unit (CTMU) with Threshold Detect (DS39743), three comparators, an LCD Controller and USB OTG.

SECTIONS AND DESIGNS

There are four major sections as follows:

- “Basic Building Blocks”
- “Industrial Applications”
- “Consumer/Appliance Applications”
- “Medical Applications”

Each section has several design examples showing the various analog peripherals. The last section is “Further Reading”, and has references and links to additional information.

BASIC BUILDING BLOCKS

The circuits presented in this section are not a complete product, but rather ‘building blocks’ used in combination with other circuitry. They perform specific, but commonly used, tasks found in most analog ‘front end’ measuring and generating circuits.

When using these design ideas, keep the following things in mind:

- Additional circuitry may be needed for ESD protection, EMI filtering or other types of interference seen in an actual product.
- Do not exceed the Common-Mode Voltages (VCM) for each module. These are listed in the “Electrical Characteristics” section of the “PIC24FJ128GC010 Family Data Sheet”.
- Low-current/battery operated products may require special attention to component values and selection.

Building Block #1: Current Monitoring

Perhaps the simplest and lowest parts count application is to measure current using a shunt resistor, and the PGA of the SD_ADC, as shown in Figure 1.

FIGURE 1: MEASURING CURRENT USING A SHUNT RESISTOR AND THE SD_ADC PGA
This configuration uses a differential input channel of the SD_ADC. The value of the resistor is usually selected as an even decade (1Ω, 0.1Ω, etc.) to make the current calculation easier. Lower resistances will dissipate lower wattage and can measure current in the tens of amps. The SD_ADC can measure AC or DC current as long as the voltage drop across the resistor is within the common-mode range of the PGA. The true differential input can detect positive or negative current flow reference to its input pins. Based on the current range expected and the resistor value, the gain of the PGA should be such that output voltage into the SD_ADC never exceeds the reference voltage used.

When measuring DC current, the following SD_ADC control bits should be programmed as follows:

- DITHER – Set low or high, based on the reference voltage (low if the reference is within 0.5V of SVDD, high if the reference is more than 0.5V below SVDD)
- PWRLVL = 0 (normal bandwidth)
- CHOP = 0b11 (enabled)
- SDOSR = 0 (ratio is 1024)

AC current may require a smaller SDOSR value depending on the application and nature of the current waveform.

A commonly used current sense resistor is 0.1Ω at 1% tolerance in a package size of 0805. These resistors can measure currents up to 1A, which generates a 100 mV voltage drop. The PGA is then programmed for a gain to place the SD_ADC voltage within the range of the reference voltage. For non-critical measurement, SVDD is derived from VDD using a passive filter (C1, C2 and ferrite bead, L1). For this example, if SVDD is 3.3V, a gain of 32 can be used to measure currents up to 100 mA with the one shunt resistor shown.

Applications that require full 16-bit accuracy may require an external, low noise, high stability (<9 ppm) voltage reference which connects to the CH1+ pin and selecting it in the SD1CON register. Accuracy can also be improved by averaging a large number of samples (16, 32 or 64).

Typical values for the SVDD filter are:
- C1: 2.2 μF/25V
- C2: 0.1 μF/50V
- L1: 100Ω Ferrite Bead

**CODE EXAMPLE**

The following C routines demonstrate how to initialize the SD_ADC for gain and offset calculation, read a ‘raw’ SD_ADC result and then perform the gain/offset calculation for the current shunt example. The calculation assumes a 3.3V reference voltage for the SD_ADC (using the SVDD pin) and a 1Ω shunt resistor.

The routines used to interface to the SD_ADC for this example is shown in Example 1.
EXAMPLE 1: SD_ADC GAIN AND OFFSET CALCULATIONS

static signed long int offset;
static double SDGain;
//----------------------------------------------------------------------------
//Public prototypes
//----------------------------------------------------------------------------
void ADC_SD_Init();
float ADC_SD_GetCalibratedResult();
signed short int ADC_SD_GetRawResult();
//----------------------------------------------------------------------------
// Description: Initializes the sigma delta ADC module and performs initial
// measurements to establish gain and offset errors, so that they can be used
// later to get calibrated results. This API should also be called first,
// prior to calling the other sigma delta ADC API functions. The measurement
// channel is CH1 (only option in this demo, but can be expanded to include CH0).
//----------------------------------------------------------------------------
void ADC_SD_Init()
{
  unsigned long i;
  unsigned char count;
  signed long int maxValue;
  #define EXPECTED_MAX_VALUE (double)10430
  // 32.83mV static input, 32 gain, 3300mV ref, 32767 count register
  //Disable ADC while configuring it.
  SD1CON1 = 0x0000;
  //Setup Sigma Delta ADC Module, so that it is ready to perform gain and
  //offset measurements (so we get proper gain and offset calibration values to use later).
  SD1CON1bits.PRLVL = 1;  // High power mode
  SD1CON1bits.SDREFP = 0;  // Positive Voltage Reference is SVDD
  SD1CON1bits.SDREFN = 0;  // Negative Voltage Reference is SVSS
  SD1CON1bits.VOSCAL = 1; // Internal Offset Measurement Enable
  SD1CON1bits.DITHER = 1; // Low Dither, because using SVDD as reference
  SD1CON1bits.SDGAIN = 32; // Gain is 32:1 for offset measurement and application
  SD1CON2bits.W Rem = 2;  // Round result to 16-bit
  SD1CON2bits.SDMN = 1;    // SDxRESH/SDxRESL updated on every Interrupt
  SD1CON2bits.SDINT = 3;   // Interrupt on every data output
  SD1CON2bits.CHOP = 3;    // Chopping should be enabled always
  SD1CON3bits.SDCH = 1;    // Channel 1 is used for this demo code
  SD1CON3bits.SDCS = 1;    // Clock Source is 8MHz FRC
  SD1CON3bits.SDOSR = 0;   // Oversampling Ratio (OSR) is 1024 (best quality)
  SD1CON3bits.SDIV = 1;    // Input Clock Divider is 2 (SD ADC clock is 4MHz)
  // Enable the ADC module now that it is configured
  SD1CON1bits.SDON = 1;
  // Wait for a minimum of five interrupts to be generated. Need to throw at least
  // the first four away when using interrupt every period option, since the
  // SINC filter needs to be flushed with new data when we change
  // ADC channel or initialize the ADC.
  for (i = 0; i<6; i++) // (value must be >= 5)
  {
    IFS6bits.SDA1IF = 0;  //Clear interrupt flag
    while(IIFS6bits.SDA1IF == 0); //Wait until hardware says we have a result ready.
    IIFS6bits.SDA1IF = 0; //Clear interrupt flag
  }
  while(IIFS6bits.SDA1IF == 0); //Wait until hardware says we have a result ready.
  IIFS6bits.SDA1IF = 0; //Clear interrupt flag
  offset = (signed short int)SD1RESH; // cast to accommodate the sign bit extend
  // Switch off offset measurement mode, now that we have the value
  SD1CON1bits.SDON = 0;
  SD1CON1bits.VOSCAL = 0;
  // Now reconfigure ADC so as to measure the SVDD through channel 1,
  // and compare it against the expected result (after offset correction).
  // Use the comparison to compute the optimum gain calibration factor, and
  // save the value, so that it may be used later.
  SD1CON3bits.SDCH = 1; // point to the input voltage
  SD1CON1bits.SDON = 1;  // SD_ADC back on to make next measurement
  // Wait for a minimum of five interrupts to be generated.
  for(count=0; count<6; count++)
EXAMPLE 1:  SD_ADC GAIN AND OFFSET CALCULATIONS (CONTINUED)

// Clear interrupt flag.
IFS6bits.SDA1IF = 0;
// Wait for the result ready.
while(IFS6bits.SDA1IF == 0);

// Save the maximum value to calculate the gain.
maxValue = (signed short int) SD1RESH; // must cast as signed as is declared unsigned in header
// Calculate gain.
SD_gain = EXPECTED_MAX_VALUE/((double)((signed long int)maxValue-offset));

// It is important to note that the gain error of the PGA is not corrected in this example, as
// the gain is 1 when measuring SVDD. You can always 'hard code' the gain correction to some
// value or other scheme. A good initial value is in the range of 1.0475 to 1.0575
// Disable ADC while re-configuring it (for normal operation mode, now that
// we have establish gain/offset calibration values for the ADC).
SDICON1bits.SDON = 0; //Setup Module for normal reads on CH1
SDICON3bits.SDCH = 1;
SDICON1bits.SDGAIN = 5; // Gain is back to 32:1 for the application code
SDICON1bits.SDON = 1;

// Wait for SYNC filter to flush, so the next result the application tries
//to read will be valid
for(count = 0; count < 6; count++) //throw away first >=4 samples for SINC filter delay
{

while(IFS6bits.SDA1IF == 0); //Wait until hardware says we have a result ready.
IFS6bits.SDA1IF = 0; //Clear interrupt flag
}
// The ADC is now ready, and the first sample of the newly selected channel
// should be available in SD1RESH.

// Function: signed short int ADC_SD_GetCalibratedResult(BYTE Channel)
// Description: Does what is needed to get an ADC sample from the channel
// specified. The value that is returned is the raw result generated by
// the hardware, and will therefore contain offset and gain error.
//----------------------------------------------------------------------
signed short int ADC_SD_GetRawResult()
{
// Wait for a new result
IFS6bits.SDA1IF = 0; //Clear interrupt flag
while(IFS6bits.SDA1IF == 0); //Wait until hardware says we have a result ready.
return SD1RESH;
}

// Function: signed short int ADC_SD_GetCalibratedResult(BYTE Channel)
// Description: Does what is needed to get an ADC sample from the channel
// specified, and then compensate it for offset and gain. The calibrated
// result is then returned.
//----------------------------------------------------------------------
float ADC_SD_GetCalibratedResult()
{
    signed long int OffsetCalibratedResult;
    double GainAndOffsetCorrectedResult;
    // Get the raw ADC result
    OffsetCalibratedResult = ADC_SD_GetRawResult();
    // Correct for offset error measured previously
    OffsetCalibratedResult -= offset;
    // Now compute the gain corrected result.
    GainAndOffsetCorrectedResult = (double)OffsetCalibratedResult * SD_gain;
    // ratio of adc result to max count of 15bits plus the sign bit
    GainAndOffsetCorrectedResult = GainAndOffsetCorrectedResult / 32767;
    // multiply by voltage in mV to get result in milliamps
    GainAndOffsetCorrectedResult = GainAndOffsetCorrectedResult * 3300.0;
    // scales back from 32x gain and divides by 1 (the shunt resistor)
    GainAndOffsetCorrectedResult = GainAndOffsetCorrectedResult / 32.0
    return (float) GainAndOffsetCorrectedResult; // current in ma
}
Building Block #2: Programmable Current Sources

Some measuring transducers, such as RTDs, require a constant current in order to generate the resultant voltage. The basic circuit for this is shown in Figure 2.

**FIGURE 2: BASIC CIRCUIT FOR CONSTANT CURRENT SOURCE**
One of the internal op amps is used to drive a transistor for current gain. One input of the op amp is tied to a reference voltage, which is commonly, a simple voltage divider from the VDD supply. An improvement to the circuit is shown in Figure 3, which uses the SD_ADC to measure the actual current in the load.

**FIGURE 3: SD_ADC MEASURING ACTUAL CURRENT IN THE LOAD**

The measured current can then generate an error value that the firmware uses to drive the DAC to the correct voltage. This is an example of a servo loop; the SD_ADC is used to change the DAC values up or down so that any offset or gain errors are canceled out. The exact DAC voltage needed does not have to be in a calibrated look-up table. The SD_ADC ramps up or down via firmware until the proper current is generated. If the VDD voltage changes, for example, the servo loop would detect the current change and then write a new value to the DAC, which then adjusts the current source. The speed of the servo loop response can easily be set in the firmware using timed ramping algorithms.
Building Block #3: Input Filtering/DC Offset Correction

The ADCs in the PIC24FJ128GC010 sample the input signal at a very high rate. The SD_ADC uses a 1 MHz, 2 MHz or 4 MHz sample clock (not to be confused with the variable data rate or OSR clock) and the pipeline converter is variable from a 1 MHz to 10 MHz sample rate. In any sampled system, input low-pass filters are required to prevent the aliasing components to add to the result of the conversion.

In addition, special consideration is required when using the SD_ADC, due to the inherent low-pass response of the decimation filter used to convert the modulator’s serial data to a parallel 16-bit value. The decimation filter’s frequency response is based on the selected OSR, as seen in the following Figure 4.

**FIGURE 4: DECIMATION FILTER FREQUENCY RESPONSE BASED ON SELECTED OSR**

The filter has a gradual roll-off characteristic near the -3 dB point. It is important to note that the 'flat' portion of the decimation filter response (less than a 0.1 dB loss) is approximately 1/8th of the bandwidth. In order to accurately sample AC signals, the frequency content of the input signal must fit within the overall flat pass band of the decimation filter. The external filter must prevent harmonics of the input signal to lie outside the decimation filter’s total pass band or severe distortion will occur. Note that this is a different paradigm than traditional ADCs, where the requirement is to assure the harmonics are down ~6 dB/bit at the sample frequency, such as the pipeline ADC. This is why the SD_ADC in the PIC24FJ128GC010 is best suited for either AC sine wave or DC measurements. The sigma-delta architecture can be used for higher bandwidth signals (audio is a good example), but those sigma-delta ADCs implement specific design elements not found in the PIC24F design.

The low-pass filter response of the SD_ADC decimation filter is of great advantage when measuring DC signals from transducers. High-frequency noise is greatly attenuated by the decimation filter, but at the highest OSR setting of 1024, the -3 dB bandwidth of the input is still ~325 Hz, which will contain AC power line harmonics. If these harmonics are an issue, additional filtering may be required (see the next section for details).

Anti-aliasing filters are usually used in the context of digitizing AC signals. But what if the measurement is a DC signal, like temperature? Input filtering is still required because noise will be picked up by the sensor leads, noise in the power supply/voltage reference lines or other sources, such as the PC board layout. It is difficult to state in specific terms what is needed for all cases, so some guidelines will be shown in the following examples and applications.
A single pole RC filter is all that is required when using the SD_ADC, as the requirement is to reduce by >20 dB any signals above the Nyquist frequency. Note that this is very different than the requirement for the pipeline ADC, which requires a signal reduction of >74 dB at the Nyquist frequency or aliasing will occur. The modulator in the SD_ADC pushes the noise spectra out in frequency, where it is then removed by the decimation filter. Figure 5 shows a simple RC filter suitable for the SD_ADC.

FIGURE 5: SIMPLE RC FILTER

![Simple RC Filter Diagram]

This filter has a flat response to 8 kHz, has a 48 kHz bandwidth and attenuates by more than 20 dB frequencies above 500 kHz. Frequencies are attenuated by more than 32 dB above 2 MHz for the desired 4 MHz sample rate of the modulator.

For the 12-bit pipeline ADC, the number of poles in the anti-aliasing filter will depend on the bandwidth required by the application. The fact that the lowest sample rate of the P_ADC is 1 MHz relaxes the requirements somewhat for the slope of the anti-aliasing filter. Table 1 is a simple guideline for Butterworth filters. The exact nature of the design will be application-specific. This data is for a 1 MHz sample rate.

TABLE 1: BUTTERWORTH FILTERS FOR THE PIPELINE ADC

<table>
<thead>
<tr>
<th>Number of Poles</th>
<th>Input Signal Bandwidth (Before Aliasing)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>90 Hz</td>
</tr>
<tr>
<td>2</td>
<td>6 kHz</td>
</tr>
<tr>
<td>3</td>
<td>24 kHz</td>
</tr>
<tr>
<td>4</td>
<td>50 kHz</td>
</tr>
</tbody>
</table>

A detailed explanation of anti-aliasing filters and their design is found in Microchip Application Note: AN699, “Anti-Aliasing, Analog Filters for Data Acquisition Systems” (DS00699).

FILTER EXAMPLE #1: PGA INPUT PINS WITH LONG SENSOR LEADS

In many cases, the sensor to be measured is physically located at a remote location from the PC board containing the PIC24FJ128GC010. The leads from the sensor to the board can pick up noise from AC power lines, RF signals from AM radio broadcasts and other interferences. The PGA input impedance is high (>1M), so care is needed to keep spurious signals from causing jitter in the measurement.

Figure 6 shows typical input signal conditioning for the differential PGA input on the SD_ADC.

FIGURE 6: INPUT SIGNAL CONDITIONING FOR DIFFERENTIAL FILTER NOISE

![Input Signal Conditioning Diagram]

The input resistors are of equal value, as are common-mode capacitors, C1. Differential-mode capacitor, C2, is usually set to 10 times C1 in order to attenuate the differential noise greater, since the PGA input has very high common-mode rejection of the noise. The -3 dB corner frequencies of the filter are given by Equation 1:

EQUATION 1:

\[
\text{FREQ}_{\text{diff}} = \frac{1}{4\pi R(C2 + \frac{C1}{2})}
\]

\[
\text{FREQ}_{\text{cm}} = \frac{1}{2\pi RC1}
\]

A typical application is measuring heart pulse rate using skin galvanic contacts. The resistors, R, not only form the low-pass noise filter, but add ESD protection and limits the current to the skin. A typical application would be: R = 100K, C1 = 220 pF, C2 = 2.2 nF. This results in the corner frequencies of 345 Hz differential and 7.2 kHz for common-mode.
FILTER EXAMPLE #2: LOW-PASS FILTERS
AND BIASING USING THE DAC

Figure 7 shows a 2-pole Sallen-Key 1 kHz, low-pass filter with a gain of 12 using an internal op amp. Microchip has a no-cost filter design tool (FilterLab® 2.0) which supports many different filter types. (9)

FIGURE 7: 2-POLE Sallen-Key 1 kHz, LOW-PASS FILTER

Since this is a single supply system, a Common-Mode Voltage, \( V_{CM} \), is required. A typical circuit for generating a Common-Mode Voltage is shown in Figure 8. The resistors, \( R_1 \) and \( R_2 \), form a divider, and \( C_1 \) is additional filtering. The op amp voltage follower provides a low-impedance source for the rest of the circuitry. For these examples, this circuit is replaced by one of the PIC24FJ128GC010 DAC output channels. Since the Common-Mode Voltage is usually \( V_{DD}/2 \), this is easily supplied by the DAC with the advantage that it can be adjustable in firmware to also compensate for DC offsets in the internal filter op amps.

FIGURE 8: CIRCUIT FOR GENERATING COMMON-MODE VOLTAGE
If there are available MCU cycles in the application, a special software algorithm can be used to remove DC offsets. (4)

Another common filter needed for data acquisition systems is an AC power line interference notch filter. This is a common issue in medical equipment, where long patient leads and high input impedance amplifiers are susceptible to AC hum from both power line wiring, and overhead fluorescent lighting ballast (that will interfere at 2x the line frequency). Figure 9 shows a passive notch filter with greater than 50 dB of stop-band attenuation. (5)

![Passive Notch Filter with 50 dB of Stop-Band Attenuation](image)

A more effective comb filter is possible to implement in firmware that can remove both the fundamental and the first harmonic, which solves both AC line and lighting ballast interference. (6)

Where all 3 Capacitors are the Equal Value of C and RF = 12R.

$$f_{\text{NOTCH}} = \frac{1}{2\pi RC\sqrt{3}}$$
FILTER EXAMPLE #3: SELF-BIASING USING THE PGA INPUT

Interfacing external signals to ADCs requires forcing the resulting signal into the common-mode range of the input pins. A common problem is resolving a small AC signal with a large DC offset. The usual solution of AC coupling may not be the best solution, especially if the AC signal is low frequency (<5 Hz). Since the MCU is available for post-processing, there are ‘tricks’ which involve the combination of analog filtering and MPU math routines to get the final result.

This example of a self-bias scheme is shown below in Figure 11. This is an interface to a Passive Infra-Red (PIR) receiver used in home security and motion detection.

FIGURE 11: INTERFACE TO A PASSIVE INFRA-RED (PIR) RECEIVER

The output stage of a typical PIR module is a FET source follower, biased by the 47K resistor to ground. This configuration is very sensitive to power supply ripple/noise, so an RC filter of 100Ω/10 μF is used. The output of the PIR is a DC voltage of ~2V with a small AC signal, proportional to the motion of the heat source, in the range of 10 mV. The DC voltage varies as the ‘background’ heat changes, due to lighting, vibration and other factors.

The circuit works by using 2 RC filters with a ‘fast’ time constant for the positive input and a much slower (10,000x) time constant filter on the negative (reference) pin. The differential amplifier subtracts the background DC bias from the actual motion detection signal. The PGA is set for a gain of 32. This gain is too small to achieve a close to full-scale signal swing into the SD_ADC, but the 16 bits of resolution, combined with sample averaging, is adequate to resolve motion detection.
INDUSTRIAL APPLICATIONS

Bridging Measurements

A common industrial process control measurement involves a bridge measurement. This is a specific topology of 4 resistances (called a Wheatstone bridge), 3 of which are known and of the same value, and an unknown 4th resistor. The 4th resistor is what changes value as a result of the applied phenomenon: air pressure, compression/strain, humidity and many others. A typical bridge transducer is modeled in Figure 12.

FIGURE 12: TYPICAL BRIDGE TRANSDUCER

These types of transducers use precise ratio matched resistors to generate the output voltage according to Equation 2:

**EQUATION 2:**

\[ V_{OUT} = V_E \left( \frac{R}{R + RG} - \frac{1}{2} \right) \]

Where \( V_E \) is called the Excitation Voltage (can be AC or DC) and \( RG \) is the varying element to the other resistors, \( R \). The resistors, \( R \), are typically 5K to 20K in value. The change in resistance over the transducer range for \( RG \) is small, which in turn, generates a small differential output voltage: a 5-50 mV full-scale change is commonly specified. This needs to be amplified and the DC bias removed before the ADC input pin.

Many bridge measurements only have 5-10 mV full-scale outputs, so a gain of 32 (the maximum in the PIC24FJ128GC010 device’s PGA) is insufficient to have enough dynamic range to give the required readings. The two internal op amps can be configured as an instrumentation amplifier, whose gain is set by a single resistor. The amplifiers also act as the anti-aliasing filter by adding 2 capacitors across the feedback resistors. The gain of the PGA can be set to a lower value (in the 2x-8x range), decreasing the overall input noise to the SD_ADC. Since the data taken from bridging transducers tends to be slowly changing, many samples are averaged to reduce noise and improve accuracy. See Figure 13 for an example using the internal op amps.

FIGURE 13: USING INTERNAL OP AMPS WITH A BRIDGE TRANSDUCER
The gain of the op amp input filter is given by:

**EQUATION 3:**

\[
\text{Gain} = 1 + \frac{2R}{R_G}
\]

With the values shown, the first stage gain is ~9.8, and with a PGA gain of 4, gives an overall gain of ~39. Resistor, RG, can be adjusted to give the required dynamic range based on the type of transducer used. The 40.1K feedback resistor and 10 nF capacitor roll the amplifier off at 400 Hz. The actual gains and filter requirements will be application dependent. The altimeter shown has a 60 mV, full-scale range from 0 to 20,000 ft., so the input voltage to the SD_ADC is 0-2.4V.

If the CMRR performance of the internal op amps is insufficient, Microchip offers a stand-alone instrumentation amplifier, the MCP6N11. This part is used for the blood pressure meter example in the “Blood Pressure Cuff” section.

Additional application information on bridge measurements can be found in Microchip Application Note: AN695, “Interfacing Pressure Sensors to Microchip’s Analog Peripherals” (DS00695).

**AC Power Line Monitoring**

AC power line monitoring is divided into two areas: revenue versus non-revenue metering. AC power providers have adopted stringent standards (IEC 62052, IEC 62053 and ANSI C12.20) for power meters that are used to bill the customer. These meters are bound by a 0.2% or 0.5% accuracy specification and have over 30 individual tests/calibrations required in order to be compliant. Microchip has several reference designs and application notes for revenue accurate metering. They can be found at: [www.microchip.com/powermonitoring](http://www.microchip.com/powermonitoring).

This following example is for a non-revenue AC power monitor, which would be used for equipment testing, fault finding, general information or other uses. The actual power calculations needed are shown in the sample code and design guide for the “PIC18F87J72 Single-Phase Energy Meter Reference Design” (DS51931).

The schematic for the measurement portion of the meter is shown below in Figure 14.

**Figure 14: SINGLE-PHASE 120VAC POWER MONITOR**
The interface for the AC power monitor is straightforward. For the voltage measurement, a simple resistor divider is used from a neutral combined with an anti-aliasing filter. Note that the differential input of the SD_ADC is used, since the signal is AC and will cause the sign bit of the result to change. The AC current measurement uses a low resistance, high-powered, cabinet mounted current shunt. The voltage drop is measured directly, with the PGA amplifier’s gain set to maximize the dynamic range. This setting will depend on the specific shunt used.

The 5-cycle pipeline delay, caused by switching the SD_ADC channel MUX between voltage and current measurements, means the instantaneous AC power cannot be directly calculated for revenue use. The pipeline delay is fixed at 5 data rate periods, so the faster the data rate, the smaller the ‘phase error’. Table 2 shows the error as a function of OSR. Note that the lower the OSR, the lower the accuracy of the converter without using data averaging or other filtering algorithm.

### Table 2: OSR Functions

<table>
<thead>
<tr>
<th>OSR Value</th>
<th>Phase Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1024</td>
<td>30.7%</td>
</tr>
<tr>
<td>512</td>
<td>15.4%</td>
</tr>
<tr>
<td>256</td>
<td>7.7%</td>
</tr>
<tr>
<td>128</td>
<td>3.8%</td>
</tr>
<tr>
<td>64</td>
<td>1.9%</td>
</tr>
<tr>
<td>32</td>
<td>1.0%</td>
</tr>
<tr>
<td>16</td>
<td>0.5%</td>
</tr>
</tbody>
</table>

Smaller settings of the OSR value will result in increasing the bandwidth of the decimation filter, which may add noise to the resulting measurement. Since the final calculations can be done at a slow rate (a display update rate of 1 Hz is common in this application), hundreds of samples can be filtered/averaged in order to more accurately calculate the power used by the load.

For more accurate or revenue grade metering, the 12-bit pipeline ADC can be used for voltage measurement, triggered by the SD_ADC interrupt flag. The SD_ADC is used for the current measurement. The application software must accurately compensate for the delay in time between the 12-bit ADC measurement and the inherent latency in the SD-ADC decimation filter. This will require placing measurements into a RAM buffer and aligning the calculations to the cycle delay of the MCU, and with the correct samples for voltage and current.

For a dual channel SD_ADC, with phase delay compensation specifically designed for revenue power meters, see Microchip’s MCP3901. Information for this device is located here: [www.microchip.com/mcp3901](http://www.microchip.com/mcp3901).
CONSUMER/APPLIANCE APPLICATIONS

Full Comfort Thermostat

Modern thermostats have evolved past temperature-only measurements on a rotary dial indicator. They can be entire miniature weather stations, logging temperature, humidity, CO₂ concentration and barometric pressure. The data can be sent wirelessly over Wi-Fi or Bluetooth to your cell phone or computer. They can employ motion detection to record if you are home or away and do other tasks as well. The following example shows how to interface three of these sensors to the PIC24FJ128GC010. The example is shown below in Figure 15.

FIGURE 15: COMFORT THERMOSTATS INTERFACED TO THE PIC24FJ128GC010

Studies have shown that people report air quality as ‘stale’ or ‘stuffy’ when the CO₂ concentration exceeds ~1200 ppm, compared to a baseline ‘clean outdoor’ concentration of 350 ppm. This example uses a specialized sensor to record the CO₂ level. The sensor requires a power supply of 6V at 200 mA. This is derived from the system 3.3V supply by boost regulator, MCP1650. The sensor’s output stage requires very high input impedance, which the internal CMOS op amp can provide. The op amp is configured as a voltage follower by setting the NINSEL<2:0> bits to ‘110’ in the SFR register, AMP1CON.

The thermostat measures temperature with an NTC thermistor using the CTMU circuitry of the PIC24FJ128GC010 and the relative humidity using a DC ratiometric sensor connected to the SD_ADC. The data gathered by the PIC24FJ128GC010 can be shown on a local LCD, transmitted over USB, or sent out via a wireless adapter module (such as the MRF24WG0MA using an SPI port).
Audio Using the DAC Outputs

The two DAC channels can be used to generate various types of audio, from speech playback using ADPCM, to audio alert/warning tones. With the proper buffer amplifiers, they can drive standard 16Ω ear bud headphones. See Figure 16 for a simple way to drive small speakers or headphones using the DAC channels.

FIGURE 16: DRIVING HEADPHONES USING THE TWO DAC CHANNELS

The audio tones generated by the DAC are low-pass filtered and sent to the MCP6022 buffer amplifier. This op amp has a high-current output stage, capable of driving ±20 mA. This is sufficient for adequate volume levels in standard ear buds. The series 15Ω resistor limits the volume delivered and may not be needed in all applications. The AC audio will have a DC bias, which is removed by the 100 μF capacitor. The low-pass filter shown has a cutoff frequency of ~16 kHz. For speech output, this should be lowered to ~4 kHz. When driving speakers for alert tones in appliances, the 15Ω resistor can be reduced to ~3Ω, depending on the volume required.
MEDICAL APPLICATIONS

The section describes two different medical applications suitable for the PIC24FJ128GC010: a blood pressure cuff and a glucose measurement device. The LCD, USB and XLP features of the part, coupled with precision measurements using the 16-bit SD_ADC, allow many small, handheld design possibilities.

**Note:** These examples are basic guidelines only and final end-use application may be subject to agency approvals. They are for demonstration purposes and are not intended for medical diagnosis.

Blood Pressure Cuff

Modern blood pressure measurements use a digital readout versus the older column of mercury in a glass tube approach. The pressure reading is still given as mmHg (millimeters of mercury), over the range of 0-250 mmHg. Available pressure transducers use a resistive bridge topology to output a small (on the order of 25 mV, full scale) DC signal as the cuff pressure changes. The transducers are specified in either psi (pounds per square inch) or in kPa (kilo-Pascals), so a conversion to mmHg is needed. The conversion factors are as follows:

- \(1 \text{ kPa} = 7.5 \text{ mmHg}\)
- \(1 \text{ psi} = 51.7 \text{ mmHg}\)

The pressure transducer is not located in the arm cuff itself; it is located on the PC board, and connected to the cuff and air pump via tubing. The DC air pump inflates the cuff to a preset pressure and a solenoid valve releases pressure in a linear fashion over a period of a few seconds. Changes in the pressure are detected and an algorithm, called MWI (Measurement While Inflating), is executed by the PIC24FJ128GC010 to calculate pulse rate, systolic and diastolic pressure. The data is displayed on an LCD screen and can be saved in Flash memory, and downloaded via USB to a health monitoring program. See Microchip Application Note: AN1556, “Blood Pressure Meter Design Using Microchip’s PIC24F Microcontroller and Analog Devices” (DS00001556) for additional information.

The blood pressure cuff example schematic is shown in Figure 17. The pressure transducer (OMRON 2SMPP-02) is rated for 0-37 kPa, which seems like an odd scale factor as most transducers are specified in even ranges, such as 0-50 kPa. This particular transducer is specifically made for blood pressure cuffs, because the uncalibrated output is very close to 1 mV = 10 mmHg. The full range span is 0-310 mmHg, which covers all possible readings. The pressure transducer is designed to be driven by a 100 μA current source, which is provided by a DAC channel and one of the internal op amps. The DAC drives ~1V into the circuit to produce the 100 μA needed.

The 100 μA constant current generates voltages too small to be read by the PGA/SD_ADC. Therefore, an external instrumentation amplifier, MCP6N11, is used with a fixed gain of 31. This signal is AC coupled and amplified by a high-pass filter with a gain of 4.6, using the second op amp in the PIC24FJ128GC010. The combined gain is ~143, which feeds 2.85V into the SD_ADC when the pressure is 200 mmHg. The second DAC channel is used to provide DC offset for the input signal to lie within the common-mode range of the PGA (whose gain is set to 1 to keep the overall noise voltage lower).

After calculation, the data can be sent to an LCD display over the USB module or sent wirelessly using a Microchip wireless module. The readings can easily be time/date-stamped by the RTCC in the PIC24FJ128GC010 for data logging of blood pressure readings over time.
FIGURE 17: BLOOD PRESSURE CUFF SCHEMATIC

![Blood Pressure Cuff Schematic Diagram](image_url)
Glucose Measurement

The glucose meter is a key element of the Home Blood Glucose Monitoring device used daily by diabetes patients. The technique uses a special ‘test strip’ with specific chemicals added to cause a chemical reaction to glucose when excited by electrical current. The reaction causes electron migration to a sensing electrode, which is read by the meter over a period of time. The concentration of glucose is calculated in units of milligram per deciliter (mg/dl) or millimole per liter (mmol/L). For a detailed implementation example, see Application Note: AN1560, “Glucose Meter Reference Design” (DS00001560).

There are many available types of test strips with their own mating connectors and interface requirements. This example is a generic, ‘3-terminal’ strip with ground, input drive signal and output current leads. Specific implementation will be based on the actual strip selected. See Figure 18 below.

FIGURE 18: GLUCOSE MEASUREMENT

The activation current for the chemical reaction is set by DAC1. The resulting current is converted to a voltage by OA1 and fed to the SD_ADC via a 2-pole, 16 Hz noise filter. DAC2 adjusts for any DC offset to reduce common-mode noise. Note that the SD_ADC input pins see the same impedance, which keeps the noise at a minimum. The software sets the OSR to 1024 for the narrowest bandwidth. Since the reaction takes several seconds, the SD_ADC can average many readings for an accurate calculation. The readings use a linear regression method for the final glucose reading.
FURTHER READING

Note the following details of the code protection feature on Microchip devices:

• Microchip products meet the specification contained in their particular Microchip Data Sheet.

• Microchip believes that its family of products is one of the most secure families 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 Microchip products in a manner outside the operating specifications contained in Microchip’s Data Sheets. Most likely, the person doing so is 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 continually improving the code protection features of our products. Attempts to break Microchip’s code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer’s risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights.

Trademarks
The Microchip name and logo, the Microchip logo, dsPIC, FlashFlex, KEELLOQ, KEELLOQ logo, MPLAB, PIC, PICmicro, PICSTART, PIC18 logo, rPIC, SST, SST Logo, SuperFlash and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

FilterLab, Hampshire, Hi-TECH C, Linear Active Thermistor, MTP, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Silicon Storage Technology is a registered trademark of Microchip Technology Inc. in other countries.

Analog-for-the-Digital Age, Application Maestro, BodyCom, chipKIT, chipKIT logo, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, dsSPEAK, ECAN, ECONOMONITOR, FanSense, Hi-TIDE, In-Circuit Serial Programming, ICSP, Mindi, MiWi, MPSAS, MPF, MPLAB Certified logo, MPLIB, MPLINK, mTouch, Omniscient Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICkit, PICtail, REAL ICE, rFLAB, Select Mode, SQI, Serial Quad I/O, Total Endurance, TSHARC, UniWinDriver, WiperLock, ZENA and Z-Scale are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

GestIC and ULPP are registered trademarks of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

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

© 2013, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

Printed on recycled paper.

ISBN: 978-1-62077-754-1

Microchip received ISO/TS-16949:2009 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.
### AMERICAS
Corporate Office  
2355 West Chandler Blvd.  
Chandler, AZ 85224-6199  
Tel: 480-792-7200  
Fax: 480-792-7277  
Technical Support:  
http://www.microchip.com/support  
Web Address:  
www.microchip.com

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

**Austin, TX**  
Tel: 512-257-3370

**Boston**  
Westborough, MA  
Tel: 508-257-3370

**Chicago**  
Itasca, IL  
Tel: 630-285-0087  
Fax: 630-285-0088

**Cleveland**  
Independence, OH  
Tel: 216-447-0464  
Fax: 216-447-0083

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

**Detroit**  
Novi, MI  
Tel: 248-848-4000

**Houston, TX**  
Tel: 713-281-9583

**Indianapolis**  
Noblesville, IN  
Tel: 317-773-8323  
Fax: 317-773-5453

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

**New York, NY**  
Tel: 631-435-6000

**San Jose, CA**  
Tel: 408-735-9110

**Canada - Toronto**  
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: 86-10-8569-7000  
Fax: 86-10-8528-2104

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

**China - Chongqing**  
Tel: 86-29-8980-9588  
Fax: 86-29-8980-9500

**China - Hangzhou**  
Tel: 0571-2819-3187  
Fax: 0571-2819-3189

**China - Hong Kong SAR**  
Tel: 852-2943-5100  
Fax: 852-2941-3431

**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: 86-755-8864-2200  
Fax: 86-755-8203-1760

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

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

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

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

### ASIA/PACIFIC
India - Bangalore  
Tel: 91-80-3090-4444  
Fax: 91-80-3090-4123

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

India - Pune  
Tel: 91-20-3019-1500

Japan - Osaka  
Tel: 81-6-6152-7160  
Fax: 81-6-6152-9310

Japan - Tokyo  
Tel: 81-3-6880-3770  
Fax: 81-3-6880-3771

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: 60-3-6021-9857  
Fax: 60-3-6021-9859

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: 886-3-5778-365  
Fax: 886-3-5770-955

Taiwan - Kaohsiung  
Tel: 886-7-213-7830

Taiwan - Taipei  
Tel: 886-2-2508-8600  
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 - Dusseldorf  
Tel: 49-2129-3766400

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

Germany - Pforzheim  
Tel: 49-7231-424750

Italy - Milan  
Tel: 39-02-748200  
Fax: 39-0331-466781

Italy - Venice  
Tel: 39-049-7625286

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

Poland - Warsaw  
Tel: 48-22-3325737

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

Sweden - Stockholm  
Tel: 46-8-5090-4654

UK - Wokingham  
Tel: 44-118-921-5800  
Fax: 44-118-921-5820

10/28/13