Analysis of Direct Sensor-to-Embedded Systems Interfacing:
A Comparison of Targets' Performance

Lars E. Bengtsson, University of Gothenburg, Sweden

ABSTRACT

This paper is concerned with the direct interfacing of resistive sensors to different embedded targets. The author uses the idea of "direct sensor-to-microcontroller" technique where analog sensors are interfaced directly to inherently digital controllers and we compare the performance of this technique when applied to a typical microcontroller (PIC18), a CPLD and an FPGA. Experimental results show that 5 V systems, like the PIC18 controller, have an advantage over 3.3 V systems in terms of better precision performance, while the CPLD outperforms both the microcontroller and the FPGA in terms of accuracy. The accuracy depends mainly on the output impedance of the system's I/O ports and the precision depends mainly on trigger level noise. The PIC18 controller also has the best performance in terms of linearity and sensitivity. A lot of works have been published concerning direct interfacing to microcontrollers, but little attention has been paid to alternative targets like CPLD and FPGA. This work will benchmark these different kinds of targets and prove that the direct interfacing technique can also be applied to CPLDs and FPGAs.

Keywords: *Accuracy, Complex Programmable Logic Device (CPLD), Field Programmable Gate Array (FPGA), Micro-Controller, Precision, Sensors*

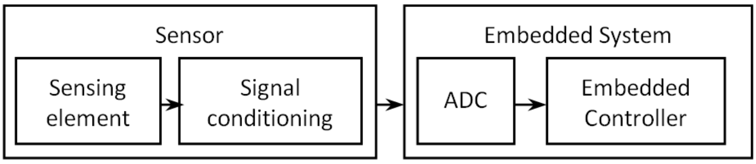
INTRODUCTION

Traditionally, analog sensors are interfaced to embedded systems via a combination of signal conditioning electronics and an analog-to-digital converter (ADC). Typically, some, or all, of the signal conditioning electronics is integrated in the sensor circuit and the ADC is typically integrated in the embedded controller. Figure 1 illustrates a typical distribution of components between the "sensor" circuit and the "system" circuit.

During the last decade, there's been a trend towards interfacing sensors directly to microcontrollers, without the use of ADCs (Cox, 1997; Merritt, 1999; Custodio et al., 2001; Jordana et al., 2003; Reverter et al., 2005; Lepowski, 2004; Reverter & Pallàs-Areny, 2006). The major advantage is that this reduces the overall cost of the acquisition system. The main idea is to transfer the analog signal variation caused by the sensor, into a *quasi* digital signal that can be measured by one of

DOI: 10.4018/ijimr.2012010103

Figure 1. Traditional data acquisition system



the controller's embedded timers. A variation in an analog voltage is transferred into an analog variation in either frequency, time duration or duty cycle ("quasi digital") as illustrated in Figure 2 (Reverter & Pallàs-Areny, 2005; Viorel, 2006).

Quasi digital signals are easily measured by any embedded system with a standard timer; measuring analog variations in time is easier (and more cost efficient) than measuring variations in an analog voltage. Also, performance parameters like dynamic range and noise immunity, typically improve when you measure a variation in time instead of a variation in analog voltage (Pallàs-Areny & Webster, 2001).

Exactly how this direct sensor interfacing is implemented depends on the sensor, whether it is resistive, capacitive or part of a Wheatstone bridge or not. We treat only resistive sensors (not being part of a bridge circuit) here. This includes, for example, common RTDs (Resistive Thermal Devices) like the Pt-1000 sensor.

Figure 2. Analog signals are transferred into quasi digital signals

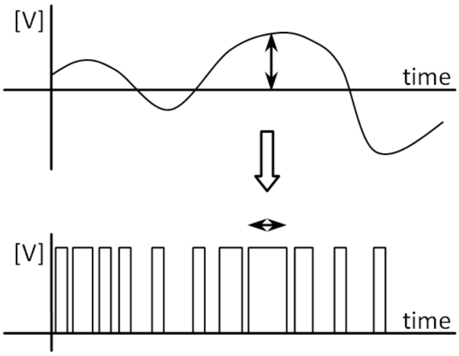


Figure 3. Direct interfacing of resistive sensor

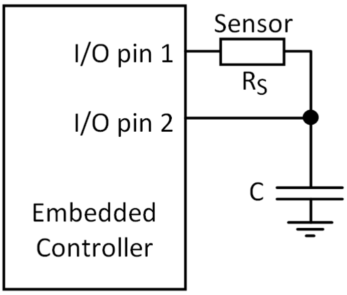


Figure 3 illustrates the basic idea of how a resistive sensor is interfaced directly to a controller.

During the first stage, I/O pin 2 is configured as output and set high while I/O pin 1 is configured as input (high impedance). I/O pin 2 will charge the capacitor to V_{OH} (I/O pin output high level). During the second stage, the pins are reconfigured; pin 2 becomes high impedance and pin 1 is configured as an output pin and set low. Hence the capacitor will discharge through the sensing element's resistance R_S. The discharging continues until the voltage on pin 2 reaches the threshold level for input logic low (V_{IL}). Figure 4 shows the charge/discharge timing diagram on I/O pin 2.

An internal timer measures the discharging time T_S in Figure 4, resulting in an integer N_S which is proportional to T_S. This number is proportional to the resistance of the sensing element (Merritt, 1999):

$$R_S = kN_S \tag{1}$$

where the constant k depends on the trigger level threshold of the embedded timer, the capacitor C and also on the controller's speed and timer setup parameters (clock rate, pre-scale).

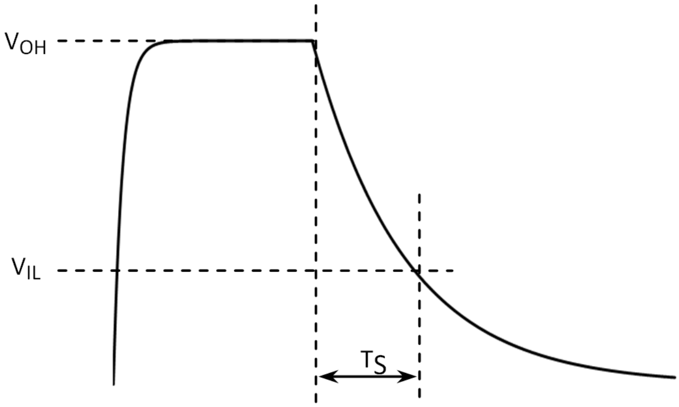
In this work we will compare the performance of three different kinds of embedded targets when they are used in direct (resistive) sensor-to-controller applications. The first target is an 8-bit PIC18F458 microcontroller from Microchip (Microchip, 2003). The second one is a CPLD (Complex Programmable Logic Device) EMP3064A from Altera (2006) and the third target is an FPGA (Field Programmable Gate Array) EP2C20, also from Altera (2008).

Any embedded target used for direct sensor-to-controller purposes, must have the following properties:

- Bidirectional I/O pins and the ability to change the direction of the I/O pins in software.
- I/O-pins must have tri-state capability; a high impedance (high-Z) state must be available for I/O-pins (in order to prevent current sinking during some stages of the charge/discharging process).
- A timer must exist (with at least 16 bits resolution)
- Preferably it needs some serial interface (RS-232 or USB) in order to transmit data to a host PC.

For conventional microcontrollers like the PIC18 family, this is not a problem since all I/O pins' data direction is easily changed by manipulating the controller's TRIS registers (Microchip, 2003). The CPLD and the FPGA circuits are programmed in VHDL and I/O pins

Figure 4. Charging/Discharging at I/O-pin 2



are made bidirectional by defining them as "inout" ports in the entity and they can be set to a high-Z state if they are declared as std_logic types. An std_logic type signal in VHDL has no less than eleven logic states (0, 1 and nine more) (Hamblen et al., 2008; Pedroni, 2004). Typically most of these nine extra states are not synthesizable in most targets, but one of them, the "high output impedance" state ('Z') is, which makes CPLDs/FPGAs potential targets for direct sensor-to-controller applications.

RELATED WORK

Cox (1997) was one of the first to suggest how to interface resistive sensors directly to microcontrollers, without the use of intermediate ADCs. He proposed an extra calibration resistor R_C and a "protection" resistor R_P, as illustrated in Figure 5.

By timing the discharging of C both via the sensor R_S and via the calibration resistor R_C, the unknown sensor resistance R_S can be estimated as a quotient (Cox, 1997):

$$\hat{R}_S = \frac{T_S}{T_C} R_C = \frac{N_S}{N_C} R_C \tag{2}$$

where N_S and N_C are the integer numbers produced by the controller's timer during the discharging through R_S and R_C, respectively.

Compared to Eq. (1), the expression is now independent of the capacitor C (Still, the *overall performance* is *not* independent of C) (Jordana et al., 2003; Reverter & Pallàs-Areny, 2006). The measurement time is of course increased, but the introduction of a calibration cycle removes/reduces all first order errors in offset, gain and temperature (Cox, 1997). Cox (1997) also suggested a protection resistor R_P, in order to protect the sinking I/O pin from ESD (Electro-Static Discharge) related breakdowns.

When it comes to selecting component values, Cox (1997) suggested that you should pick an R_C value that is one half of the maximum expected R_S, R_P should be a "small" resistor (100-200 ohms) and the charging capacitor should be

$$C = \frac{-T}{R_S \cdot \ln\left(1 - \frac{V_T}{V_R}\right)} \tag{3}$$

where T is the charging time necessary to get the desired resolution and V_R and V_T are the start and stop levels of the charging process, respectively. Reverter, Gasulla, and Pallàs-Areny (2005), would later show that the R_P resistor is also necessary in order to reduce power supply interferences.

Figure 5. The one-point calibration technique

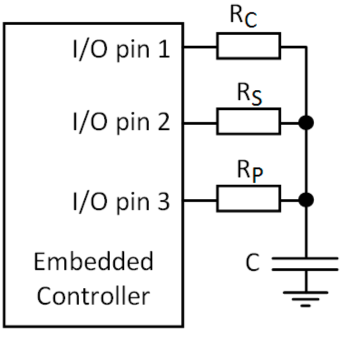
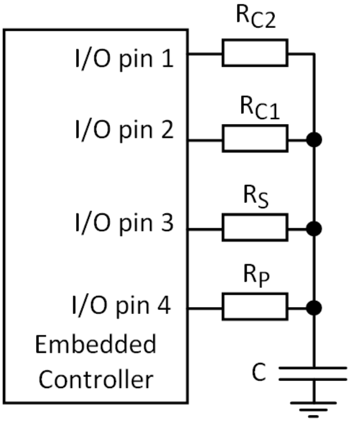


Figure 6. The two-point calibration technique



Previous to Cox (1997), Bierl (1996) had suggested a two-point calibration technique, with one extra calibration resistor as in Figure 6.

In this case, the sensor's resistance is estimated from

$$\hat{R}_S = \frac{N_S - N_{C1}}{N_{C2} - N_{C1}} \cdot \left(R_{C2} - R_{C1}\right) + R_{C1} \tag{4}$$

Reverter, Jordana, Gasulla, and Pallàs-Areny (2005) would later analyze both the one- and two-point calibration technique and conclude that the accuracy of the R_S measurements in both cases depends on the I/O-ports' internal resistances R_i. However, in the one-point calibration case, the error is of the order of R_i, while in the two-point calibration case, the error is of the order of ΔR_{ij}, i.e., the *difference* in internal resistance of the I/O-pins. Hence, from an accuracy point of view, the two-point calibration method is preferred (but the measurement time increases and hence the system bandwidth is reduced). As far as precision (uncertainty) is concerned, Reverter, Jordana, Gasulla, and Pallàs-Areny (2005) concluded that the one- and two-point calibration techniques yield similar results. The same work also suggested a "three-signal" calibration technique, which basically has the same performance as the two-point calibration technique but has better cost-performance ratio, since only one calibration resistor is required.

In 2001, Custodio et al. (2001) published a work that analyzed the importance of selecting the right calibration resistor in more detail. They analyzed in detail the influence of the microcontroller's internal parameters on the uncertainty of the discharging time measurement. The parameters included not only the current leakage of the I/O pins, but also the influence of non-ideal input/output resistances. Their model of the microcontroller's I/O pins showed that there are accuracy errors proportional to the I/O pin's output resistance. They also suggested gain errors if the two I/O pins' output parameters were not identical, and also quantified the non-linearity errors. One conclusion was that all these errors only existed when $R_C \neq R_S$ and suggested that R_C is chosen as close to the midrange value of R_S as possible (which agrees with Cox's (1997) suggestion, even if Cox's motivation was less detailed). The selection of C, suggested by Custodio et al. (2001), is that if you want an n-bit resolution representing the maximum and minimum values of R_S, then

$$C > \frac{2^n}{f \Delta R_S \cdot \ln\left(\frac{V_{OH} - V_{OL}}{V_{OH} - V_{IH}}\right)} \tag{5}$$

where f = timer's counting rate, $\Delta R_S = R_{S,max} - R_{S,min}$, V_{OH} is the output voltage of an output pin set high, V_{OL} is the output voltage of an output pin set low and V_{IH}, is the high level threshold of an input pin.

Reverter et al. (2003a) reported that the program code influences the power consumption of the microcontroller. Certain code instructions (jump and call) cause spike interferences in the power supply. Since the I/O pins' input trigger levels depend on the power supply, the trigger levels are code dependent. Results showed that jump/call instructions executed regularly in loops with a period of τ, could cause a quantization error of $\pm\tau$ (compared to the timer's inherent ±1 uncertainty). This was only observed in slow slewing signals and some solutions were suggested; the use of an external Schmitt-trigger (ST) and the necessity of a decoupling capacitor for the power supply voltage. Later, Reverter and Pallàs-Areny (2004) and Reverter and Pallàs-Areny (2006) would suggest that the microcontroller should be put into low-power mode during the time measurement in order to reduce the firmware induced uncertainties.

The need for decoupling capacitors in direct sensor-to-microcontroller circuits was further investigated by Jordana et al. (2003). They concluded that a decoupling capacitor was necessary in order to reduce the trigger uncertainties caused by power supply fluctuations. The size of the decoupling capacitor was not critical; increasing the decoupling capacitor above 100 nF did not significantly improve their results. Jordana et al. (2003) also investigated the influence of the charging/discharging capacitor. While a large capacitor is attractive since it increases the charging time and therefore reduces the relative influence of the inherent quantization error of the timer, a small capacitor has the benefit of increasing the signal's slew rate and therefore reduces the uncertainties caused by internal trigger level noise. They tried to find a reasonable compromise between the error-uncertainty-time parameters influenced by the choice of C. They found that the standard deviation of the measured resistance increased with C but they found no distinct relationship between the relative error and C. They suggested $C = 1.5\ \mu$F as a reasonable compromise between error-uncertainty-time considerations in the case of a Pt-1000 RTD.

Reverter et al. (2003b) reported in 2003 the benefits of timing the discharging of the capacitor rather than the charging. The reason is that experimental results showed that the trigger level noise was smaller for the I/O-pin's lower threshold level (used at discharging) than for the higher threshold level (used at charging). This work also showed that the trigger level was more susceptible to signal noise than power supply noise. These results were later confirmed by Reverter et al. (2004), where also the time measurement's dependence on the noise frequency was studied. The trigger uncertainty was reported to increase when sinusoidal noise was added to either the input signal or the power supply and was particularly severe when the noise frequency was close to that of the measured signal. Temperature variations and time drifts were reported not to affect the measurement.

The effective number of bits (ENOB) of the time-to-digital conversion involved in the discharging time measurement was thoroughly analyzed by Reverter and Pallás-Areny (2004) and they found that the ENOB depends on the RC time constant only up to some particular time. For small RC values, the quantization uncertainties dominate and the ENOB improves with increasing RC values. However, the consequence of increasing RC is that the signal slew rate decreases at the trigger point and hence trigger level noise uncertainties increase. At some point, trigger level noise will dominate over quantization noise and from that point, increasing RC does not improve ENOB. Hence, there is an optimal measurement speed-ENOB combination for each sensor interface. In the reported experiment, this optimal combination was observed at $RC \approx$ 2-3 ms.

In Reverter, Jordana, Gasulla, and Pallàs-Areny (2005) the influence of power supply interferences on the discharging time measurement was investigated. It was shown that power supply interferences did not influence the accuracy as much as the resolution and that these interferences are most severe at low frequencies. It was recommended that the analog and digital power supplies be separated.

AIM OF THIS WORK

The aim of any measurement is to produce an estimate $\hat{x}$ of some unknown quantity x_0. The deviation of $\hat{x}$ from the true value x_0 is the *accuracy* (or *error*) of the estimate:

$$\text{Accuracy} = \hat{x} - x_0 \tag{6}$$

The statistical spread of a number of estimators, $\hat{x}_i$, is the *uncertainty* or *precision* of the estimator and we will use the standard deviation $s\left(\hat{x}\right)$ to quantify the estimator's precision:

$$\text{Precision} = s\left(\hat{x}\right) \tag{7}$$

The aim of this work is to find out what the best approach is to direct sensor-to-embedded system in terms of what kind of embedded target to use. We will compare accuracy and precision of a typical 8-bit microcontroller, a CPLD and an FPGA.

Accuracy Analysis

According to Custodio et al. (2001), the main contribution to the estimator's accuracy in direct sensor-to-controller applications is the I/O-pins' output resistance (since it add directly to the sensor's resistance during discharging). The output resistance of a typical I/O pin on our three targets was measured as described by Reverter, Jordana, Gasulla, and Pallàs-Areny, (2005) and is summarized in Table 1.

R_{OH} and R_{OL} is the output resistance of an I/O pin when set "high" and "low", respectively. Since we measure the discharging time, it is R_{OL} that is our main concern here and that will determine the accuracy. From Table 1 it is clear that in terms of accuracy, we expect the CPLD to outperform both the PIC controller and the FPGA.

Precision Analysis

Quantization and trigger level noise have been reported to be the main uncertainty sources (Jordana et al., 2003; Reverter et al., 2004a; Reverter, Jordana, Gasulla, & Pallàs-Areny, 2005) and for slow slewing signals, as in direct sensor-to-controller applications, the trigger noise dominates (Jordana et al., 2003). According to Bouwens (1984) the trigger noise of digital counters is inversely dependent on the signal slew rate:

$$u_{trigg} = k\frac{1}{SR} \tag{8}$$

The signal slew rate is the gradient at the V_{IL} level in the discharging stage in Figure 4. Assuming $V_{OL} = 0$ (Sedra & Smith, 1991), the voltage across the capacitor during discharging is

$$V_C\left(t\right) = V_{OH} \cdot \mathrm{e}^{-t/RC} \tag{9}$$

and the time to reach the V_{IL} level is

$$t_{IL} = -RC \cdot \ln\frac{V_{IL}}{V_{OH}} \tag{10}$$

The signal slew rate affecting the trigger noise is the discharging signal's gradient at the V_{IL} level, i.e., at time t_{IL}:

$$\frac{dV_C}{dt} = -\frac{1}{RC}V_{OH} \cdot \mathrm{e}^{-t/RC} = SR\left(t\right) \tag{11}$$

$$\left.\frac{dV_C}{dt}\right|_{t=t_{IL}} = -\frac{1}{RC} \cdot V_{OH} \cdot \frac{V_{IL}}{V_{OH}} = -\frac{1}{RC} \cdot V_{IL} = SR\left(t_{IL}\right) \tag{12}$$

Table 1. Output resistance of our embedded targets

Embedded target	R_{OH} [Ω]	R_{OL}[Ω]
PIC18F458 (PIC)	57.15	21.79
EMP3064A (CPLD)	14.58	7.84
Cyclone II EP2C20 (FPGA)	46.67	73.72

Table 2. Voltage levels of typical I/O pins

Embedded target	V_{IL} [V]
PIC18F458	1.2810
MAX3064A	1.0322
Cyclone II EP2C20	1.0263

From Eq. (12) we can see that the slew rate depends directly on V_{IL}. This parameter was measured for each one of our three targets and the results are illustrated in Table 2. (Notice that the CPLD and FPGA are both 3.3 V systems.)

It is obvious from Table 2 that the use of +5 V supply voltage in the PIC controller system seems to be favorable when it comes to precision performance since this will result in a higher slew rate at the threshold level. As far as precision is concerned we expect the PIC controller to offer the best performance while we don't expect to see any major difference between the other targets.

METHODS AND EQUIPMENT

Hardware

Since the purpose of this work only was to establish the best type of target to use in direct sensor-to-embedded controller applications, either the one-point (Figure 5) or the two-point (Figure 6) calibration technique could be used. Since the one-point calibration technique is less complex and hence easier to implement, the one-point calibration technique was implemented in all three targets. In order to simulate a Pt-1000 RTD sensor, fixed resistors were used, ranging from 817.41 Ω to 2193.95 Ω. For a Pt-1000 RTD, these resistor values correspond to a temperature range of approximately −47 °C to +310 °C. For all targets, the register contents of the embedded timer measuring the discharging time were transferred to a LabVIEW program on a PC computer via an RS-232 interface (COM-port). This requires a level converter circuit (TTL-to-RS232), a MAX232 (Maxim, 2011), and in accordance to recommendations by Reverter et al. (2003b), the MAX232 circuit and the embedded controller were always supplied from separate sources. Both supply voltages were decoupled with a 1 µF capacitor which agrees with recommendations by Jordana et al. (2003).

The R_P resistor was 150 Ω (Reverter, Gasulla, & Pallàs-Areny, 2005) and the calibration resistor R_C was 1495.03 Ω, which is close to the midrange value of R_S (Cox, 1997; Custodio et al., 2001).

For all targets a 16-bit timer, updated at a rate of 1 MHz, was invoked to measure the discharging time. The exact value of the charging/discharging capacitor was chosen after careful considerations. Using Eq. (3), suggested by Cox (1997), we get C ≈ 10 µF. Aiming for a resolution of at least 10 bits, Eq. (5) (Custodio et al., 2001) results in a capacitor value of at least 3 μF. Reverter and Pallàs-Areny (2004) found in a similar setup, that an *RC* constant of 2-3 ms should be appropriate, indicating a *C* value of at least 2 μF (however they used a timer clock rate of 4 MHz).

Taking all these previously reported suggestions into account, a *C* value of 4.7 μF was finally used.

The PIC18F458 controller came in a 40-pin DIP package (Dual In Package), the CPLD was a 44-pin PLCC package (Plastic Leaded Chip Carrier) and the FPGA was a 484-pin Fine Line BGA package (Ball Grid Array). The FPGA used in this work was the one that comes on the Altera's DE1 Development and Education board (Altera, 2011) and the components in Figure 5 were connected to the FPGA's I/O-pins via an available I/O-expansion connector. For the PIC18 and the CPLD, PCBs (printed circuit boards) were designed at an in-house workshop.

Software

The software is uncomplicated and can be implemented in a number of different ways. Referring to Figure 5, the flowchart diagram in Figure 7 illustrates the software.

Both the CPLD and the FPGA were programmed in VHDL and in these cases the flowchart in Figure 7 was transferred into a simple state machine since this would minimize the VHDL code size (which is particularly important in the CPLD case since it only contains 64 macro cells) (Altera, 2006). The state machine is illustrated in Figure 8.

In the CPLD case this state machine occupied 38 macro cells of the 64 macro cells available in the EMP3064A. (In order to make it fit into the CPLD, the compiler needs to be configured to "optimize for area".) In all cases, some standard software was added for the purpose of transferring the discharging times to the LabVIEW program on the host PC via the RS-232 interface.

DISCUSSION OF RESULTS

Figure 9 shows a typical charging/discharging diagram in the PIC18 case, when R_S = 2193.95 ohms. The "long" discharging times correspond to the R_S resistor and the "short" discharging times correspond to the R_C resistor (1495.03 ohms). Compare this diagram with Table 2. The corresponding diagrams for the FPGA and the CPLD looked very similar and agreed very well to the numbers in Table 2.

For all targets, the discharging time was measured approximately 500 times and the mean and standard deviation were calculated and the mean was transferred to resistance using equation (2). Tables 3, 4 and 5 show the results.

From these tables, all our predictions are verified; the PIC18 microcontroller has the smallest standard deviation in all measurements, i.e., the overall best precision. This confirms that the PIC18's higher slow rate at the V_{IL} level is indeed favorable since it reduces the noise effects on the trigger level. This is also illustrated in Figures 10 and 11, where we compare the histograms of the PIC18 and the CPLD when R_S = 1502.12 ohms. The narrower histogram of the PIC18's data was verified in all measurements.

From tables 3 through 5 we can also see that our prediction about the accuracy performance is also indeed verified; in terms of accuracy, the CPLD outperforms both the PIC18 and the FPGA.

The results are summarized in Figure 12. However, poor precision performance due to random noise is only a matter of averaging enough samples. Figure 12 shows the results for 500 samples in each case. The CPLD can achieve the same precision as the PIC18 controller, but it would take twice as many samples (1.98) and hence the system's overall bandwidth would be reduced by a factor of two. Hence, choosing between a PIC18 controller and a CPLD is only an accuracy-precision-bandwidth tradeoff.

It should be noted that Reverter and Pallàs-Arreny (2006) have suggested further methods in order to reduce the uncertainty when using

Figure 7. Software flowchart

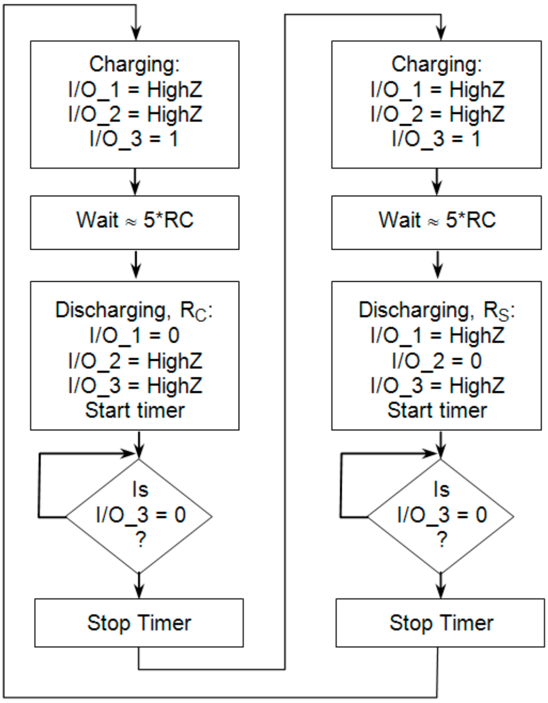
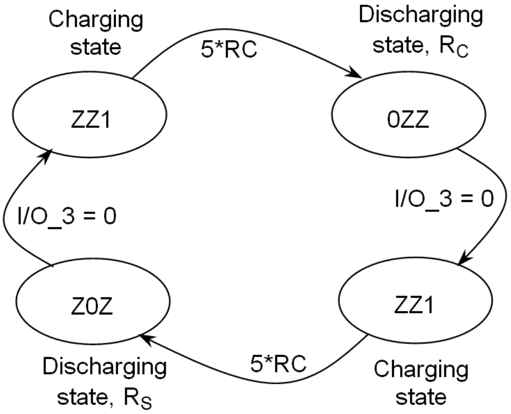


Figure 8. Direct sensor-to-embedded system state machine



microcontrollers; the use of a low-power mode that doesn't stop the embedded timer and that can be recovered by external interrupts. The PIC controller used here has indeed that option; Timer1 can run during sleep mode and it can also be revitalized by an interrupt generated by the capture module (Microchip, 2003). Hence, the width of the PIC18's distribution diagram in Figure 12 could probably be reduced even more. This suggests that it would take even more samples in the CPLD case to reach the precision of the PIC18.

Figure 9. A snapshot of the charging/discharging process

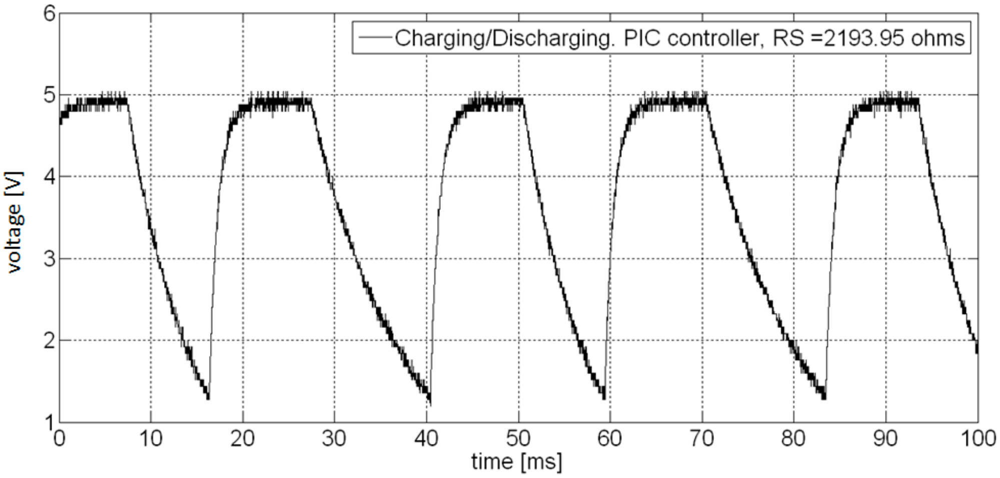


Table 3. PIC18F458 performance data

R_0 [Ω]	mean [μs]	std [μs]	$\hat{R}$ [Ω]	$\hat{R} - R_0$ [Ω]	Rel. error
817.41	4895.7	5.63	823.37	+5.96	0.73%
998.17	5958.2	7.20	1002.07	+3.90	0.39%
1193.95	7109.6	8.31	1195.71	+1.76	0.15%
1502.12	8922.4	10.07	1500.59	−1.53	0.10%
1797.44	10658.5	12.68	1792.57	−4.87	0.27%
2193.95	12992.2	14.87	2185.06	−8.89	0.41%

Table 4. CPLD EMP3064A performance data

R_0 [Ω]	mean [μs]	std [μs]	$\hat{R}$ [Ω]	$\hat{R} - R_0$ [Ω]	Rel. error
817.41	3700.6	7.93	819.10	+1.69	0.21%
998.17	4517.4	9.86	999.89	+1.72	0.17%
1193.95	5404.0	10.93	1196.14	+2.19	0.18%
1502.12	6787.8	14.63	1499.32	−2.80	0.19%
1797.44	8122.1	17.41	1794.04	−3.40	0.19%
2193.95	9925.6	49.35	2192.40	−1.55	0.07%

Table 5. Cyclone II EP2C20 performance data

R_0 [Ω]	mean [μs]	std [μs]	$\hat{R}$ [Ω]	$\hat{R} - R_0$ [Ω]	Rel. error
817.41	3322.5	11.09	838.39	+20.90	2.57%
998.17	4003.9	17.92	1010.29	+12.12	1.21%
1193.95	4753.4	20.36	1199.41	+5.46	0.46%
1502.12	5941.6	21.59	1499.22	−2.90	0.19%
1797.44	7070.7	30.10	1784.12	−13.32	0.74%
2193.95	8590.8	41.36	2167.68	−26.27	1.20%

Figure 10. Typical histogram of PIC18 data

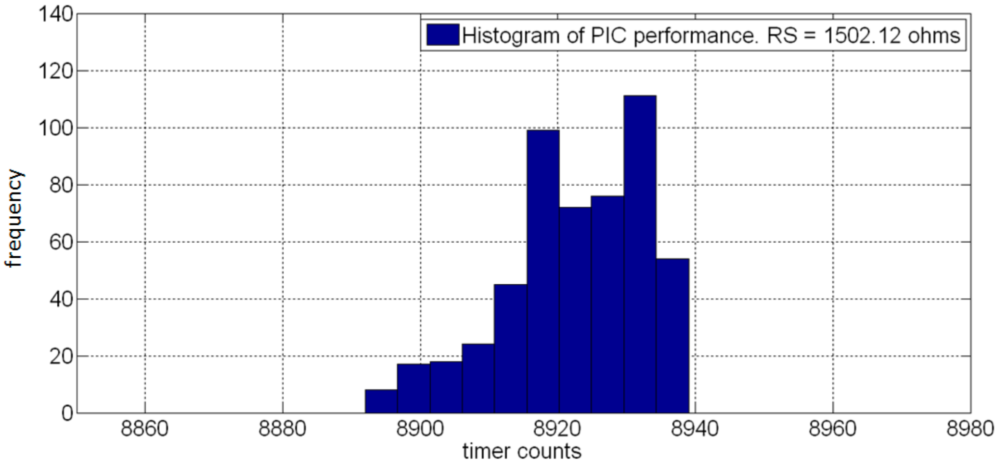


Figure 11. Typical histogram of CPLD data

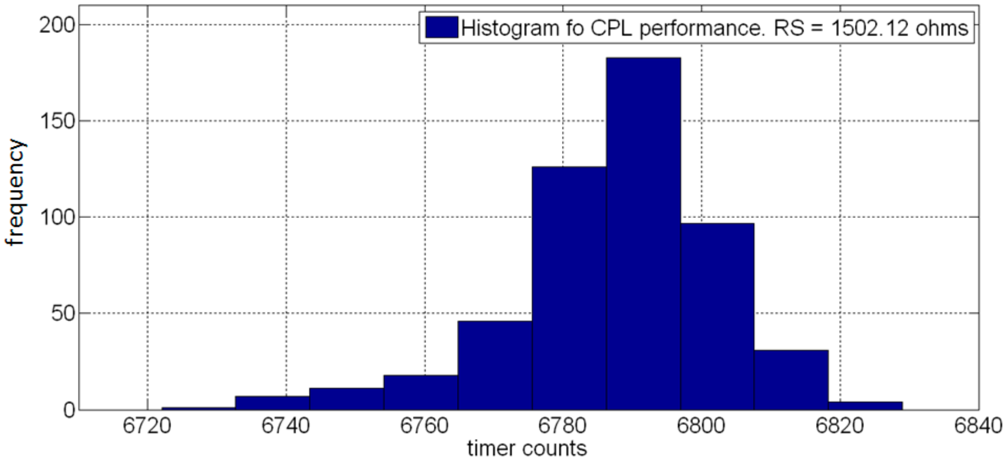
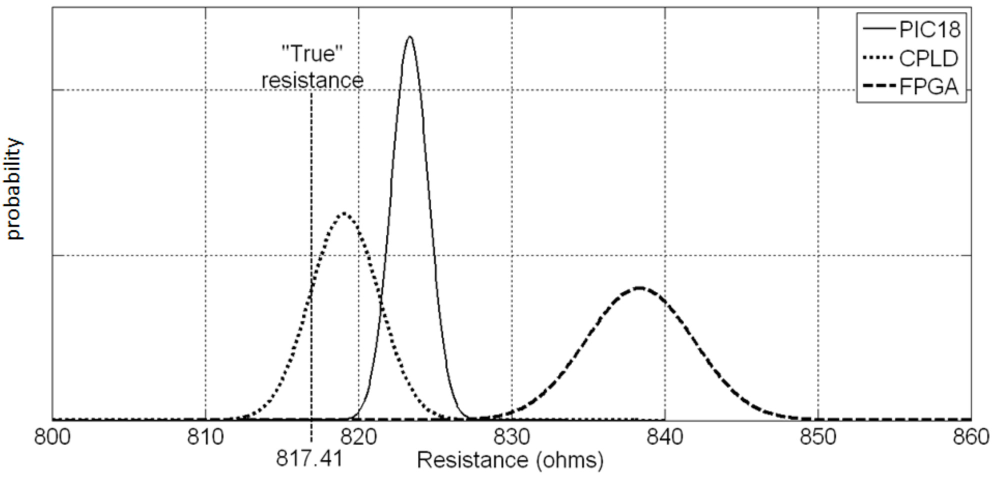


Figure 12. Performance summary of our three targets



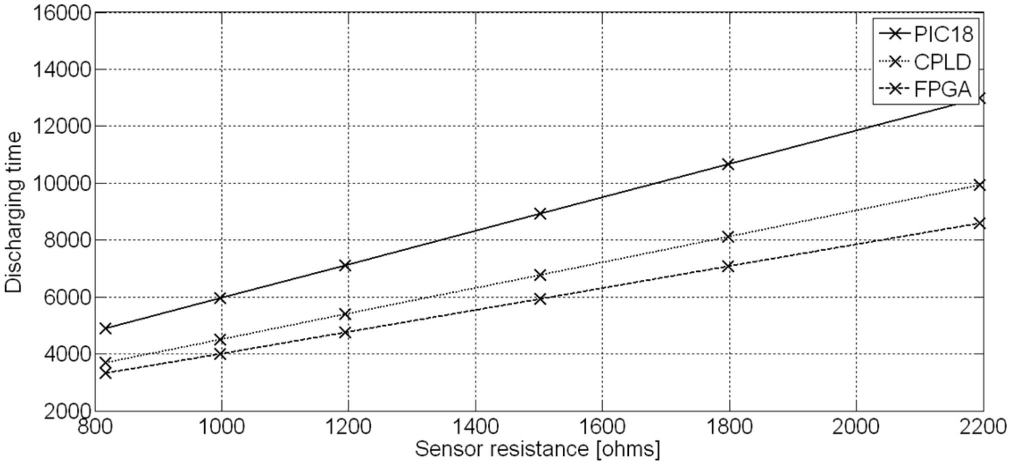
So far we have only considered accuracy and precision. Other parameters of interest in this kind of measurements are sensitivity and linearity. In Figure 13 we have plotted the discharging times produced by each system as a function of the sensor's resistance.

From this diagram it is obvious that the +5 V powered microcontroller has yet another advantage over the other targets; it has better sensitivity; 5.88 counts/ohm compared to 4.52 and 3.83 counts/ohm for the CPLD and the FPGA respectively. Hence the PIC18 will be able to detect smaller changes in the sensor's output. By fitting each set of data to a straight line, the linearity of each sensor was also measured and the linearity of the PIC18 data was a factor of 10 better than the CPLD/FPGA data (0.01% and 0.10% respectively).

The reason for the differences in, for example accuracy performance, lies of course in the manufacturing technology and manufacturers aren't very detailed about this in their data sheets (Microchip, 2003; Altera, 2006, 2008). In general though, even if microcontrollers, CPLDs and FPGAs have very different architectures (Brown & Rose, 2000), the digital I/O pins are almost always designed as the output of a flip-flop. However, they can be implemented in a number of different technologies (NMOS, CMOS, TTL, etc.) (Sedra & Smith, 1991) and the manufacturing process itself may differ; the production of a semiconductor is a multi-step process including crystal growth, wafer manufacturing, photolithography, silicon oxidizing, oxide deposition, etching, diffusion, ion implantation etc. (Quirk & Serda, 2001). Hence, even if all targets have flip-flop outputs, there are plenty of reasons why the performance and parameters of the final semiconductor would differ, and the actual parameters need to be examined for each unique situation. The results obtained in this work are not claimed to be general, but will serve as a guideline when selecting embedded targets for direct sensor-to-embedded system applications.

It appears that the PIC18 system (or similar +5 V powered microcontrollers) is the best choice of embedded target for direct sensor-to-embedded system implementations. In this work it was only outperformed by the CPLD in accuracy. In any other respect, it showed the overall best performance. When it comes to accuracy, the two-point calibration method (Reverter, Jordana, Gasulla, & Pallàs-Areny, 2005) should be considered if the loss of bandwidth is acceptable. The accuracy will then be proportional to the *difference* ΔR_{ij} in output impedance between two I/O pins rather

Figure 13. Comparison of sensitivity and linearity



than the output impedance R_i of individual I/O pins. ΔR_{ij} was measured for a set of three I/O pins on each target and the result is presented in Table 6.

From Table 6 it is clear that PIC18 would most likely have a better accuracy performance than the CPLD if the two-point calibration method was used.

Finally a comment on the asynchronous serial data link to the host computer; In this work a standard RS-232 interface was used with a MAX232 circuit as a TTL-to-RS-232 level shifter. The need of an extra circuit requires an extra power supply since the controller should be powered separately (Reverter, Jordana, Gasulla, & Pallàs-Areny, 2005). Also, modern Windows PCs are not equipped with an RS-232 COM port anymore.

The simplest solution to this is to use a “TTL-RS232 to USB” converter cable (FTDI, 2009). This eliminates the need for the MAX232 circuit and therefore eliminates the need for an extra power supply. This approach was tested in this work and no degradation in performance was observed.

CONCLUSION

Any measurement is primarily concerned with accuracy and precision. This work has shown that in direct sensor-to-embedded system applications, +5 V powered embedded systems should be used if precision is the first priority since they in general have higher slew rate at the discharging threshold level and therefore have a better immunity against trigger level

Table 6. ΔR_{ij} for different targets

	ΔR_{20} [Ω]	ΔR_{10} [Ω]	ΔR_{21} [Ω]
FPGA	0.946	0.797	0.149
CPLD	0.533	0.019	0.514
PIC18	0.137	0.027	0.111

noise compared to +3.3 V powered systems like CPLDs/FPGAs.

However, the *accuracy* in direct sensor-to-embedded systems applications are mostly depending on the output impedance of the embedded system's I/O-pins, and this work has shown that CPLDs exist that outperform a typical microcontroller in this respect.

REFERENCES

Altera. (2006). *MAX3000A programmable logic device family*. Retrieved May 18, 2011, from http://www.altera.com/literature/ds/m3000a.pdf

Altera. (2008). *Cyclone II device handbook.* Retrieved May 31, 2011, from http://www.altera.com/literature/hb/cyc2/ cyc2_cii5v1.pdf

Altera. (2011). *DE1 development and education board.* Retrieved June 24, 2011, from http://www.altera.com/education/univ/ materials/boards/de1/unv-de1-board.html

Bierl, L. (1996). *Precise measurements with the MSP430*. Waltham, MA: Texas Instruments.

Bouwens, A. J. (1984). *Digital instrumentation*. New York, NY: McGraw-Hill.

Brown, S., & Rose, J. (2011). *Architecture of CPLDs and FPGAs: A tutorial*. Retrieved August 14, 2011, from http://www.eecg. toronto.edu/~jayar/pubs/brown/survey.pdf

Cox, D. (1997). *Implementing ohmmeter/temperature sensor (Application Note No. AN512)*. Chandler, AZ: Microchip Technology.

Custodio, A., Pallàs-Arreny, R., & Bragós, R. (2001). Error analysis and reduction for a simple sensor-microcontroller interface. *IEEE Transactions on Instrumentation and Measurement*, *50*(6). doi:10.1109/19.982960

FTDI. (2009). *TTL to USB serial converter generic cable datasheet*. Retrieved August 15, 2011, from http://www.ftdichip.com/ Support/Documents/DataSheets/Cables/DS_TTL-232RG_CABLES.pdf

Hamblen, J. O., Hall, T. S., & Furman, M. D. (2008). *Rapid prototyping of digital systems*. New York, NY: Springer. doi:10.1007/978-0-387-72671-7

Jordana, J., Reverter, F., & Pallàs-Areny, R. (2003, May 20-22). Uncertainty in resistance measurements based on microcontrollers with embedded time counters. In *Proceedings of the Instrumentation and Measurement Technology Conference*, Vail, CO.

Lepkowski, J. (2004). *Temperature measurement circuits for embedded applications (Application Note No. AN929)*. Chandler, AZ: Microchip Technology.

Maxim. (2010). *+5V-Powered, Multichannel RS-232 Drivers/Receivers, 19-4323, Rev 16*. Sunnyvale, CA: Maxim Integrated Products. Retrieved June 11, 2011, from http://datasheets. maxim-ic.com/en/ds/MAX220-MAX249.pdf

Merritt, B. (1999). *MPS430 based digital thermometer* (Application Report No. SLAA038). Waltham, MA: Texas Instruments.

Microchip. (2003). *PIC18FXX8 Data Sheet* (Report No. DS41159C). Chandler, AZ: Microchip Technology.

Pallás-Areny, R., & Webster, J. G. (2001). *Sensors and signal conditioning* (2nd ed.). New York, NY: John Wiley & Sons.

Pedroni, V. A. (2004). *Circuit Design with VHDL*. Cambridge, MA: MIT Press.

Quirk, M., & Serda, J. (2001). *Semiconductor manufacturing technology*. Upper Saddle River, NJ: Prentice Hall.

Reverter, F., Casas, O., Jordana, J., & Pallás-Areny, R. (2004). Trigger uncertainty in period-to-code converters based on counters embedded in microcontrollers. *Sensors and Actuators. A, Physical*, *110*, 439–446. doi:10.1016/j.sna.2003.09.003

Reverter, F., Gasulla, M., & Pallàs-Areny, R. (2005, May 17-19). Analysis of power supply interference effects on direct sensor-to-microcontroller interfaces. In *Proceedings of the Instrumentation and Measurement Technology Conference*, Ottawa, ON, Canada.

Reverter, F., Jordana, J., Gasulla, M., & Pallàs-Areny, R. (2005). Accuracy and resolution of direct resistive sensor-to-microcontroller interfaces. *Elsevier: Sensors and Actuators A*, *121*, 78–87. doi:10.1016/j.sna.2005.01.010

Reverter, F., Jordana, J., & Pallàs-Areny, R. (2003a, May 20-22). Program-dependent uncertainty in period-to-code converters based on counters embedded in microcontrollers. In *Proceedings of the Instrumentation and Measurement Technology Conference*, Vail, CO.

Reverter, F., Jordana, J., & Pallàs-Areny, R. (2003b, June 22-27). Internal trigger errors in microcontroller-based measurement. In *Proceedings of the 17th IMEKO World Congress*, Dubrovnik, Croatia.

Reverter, F., & Pallàs-Areny, R. (2004). Effective number of resolution bits in direct sensor-to-microcontroller interfaces. *Measurement Science & Technology*, *15*, 2157–2162. doi:10.1088/0957-0233/15/10/028

Reverter, F., & Pallàs-Areny, R. (2005). *Direct sensor-to-microcontroller interface circuits*. Barcelona, Spain: Maracombo.

Reverter, F., & Pallàs-Areny, R. (2006). Uncertainty reduction techniques in microcontroller-based time measurements. *Elsevier: Sensors and Actuators A*, *127*, 74–79. doi:10.1016/j.sna.2005.11.015

Sedra, A. S., & Smith, K. C. (1991). *Microelectronic circuits* (3rd ed.). Orlando, FL: Saunders College.

Viorel, C. P. (2006, October 16-17). Microcontroller based measurements: how to take out the best we can of them. In *Proceedings of the 8th WSEAS International Conference on Mathematical Methods and Computational Techniques in Electrical Engineering*, Bucharest, Hungary.

Lars E. Bengtsson, born 1963, received his MSc Degree in Physics in 1987 and a research degree in Physics in 1991 (PhL, Licentiate of Philosophy in Physics) at the University of Gothenburg. LB was an ass. prof. at Chalmers University of Technology in Sweden from 1997 to 2007 and transferred to Gothenburg University in 2007. His major teaching/research areas are embedded systems and electrical measurement systems.