

**8-BIT MICRO CONTROLLER WITH
INTEGRATED PROGRAMMABLE
GAIN AMPLIFIER IN 10-BIT A/D
SIGNAL PATH**

Abstract

Many of today's typical analog signal processing systems are built around a microcontroller with integrated A/D converter. However, the maximum amplitude of the analog signals to be processed is often not large enough to utilize the full dynamic range of the integrated A/D. To address this problem, designers typically use external operational amplifiers to amplify those signals. Using this approach, one OpAmp per amplified analog signal is required. The concept presented in this paper takes an alternate approach. A programmable gain operational amplifier is integrated between the analog input multiplexer and a 10-bit successive approximation A/D. The microcontroller software can program different gain factors for each of the analog inputs, thus eliminating the need for any external OpAmps. This approach can result in significant component count, board space and cost savings. The performance characteristics of such an implementation are discussed together with approaches of how to address common design challenges, like offset correction. The paper also discusses the challenges of emulating the analog functionality integrated on a microcontroller, and presents a concept that allows superior analog emulation capability.

The Traditional way

Figure 1 shows the block diagram of a typical analog signal processing system built around a microcontroller. The microcontroller features an integrated A/D converter and is used to process a variety of signals. In our example various analog sensor outputs (battery voltage change, battery

charge current, and battery temperature) are measured. The microcontroller's A/D voltage reference is typically tied to the chip's Vcc voltage (5V in example) and the A/D converter thus has a full scale dynamic range of 0V...5V. Assuming a 10-bit resolution, this translates into a voltage step of 4.88mV per bit.

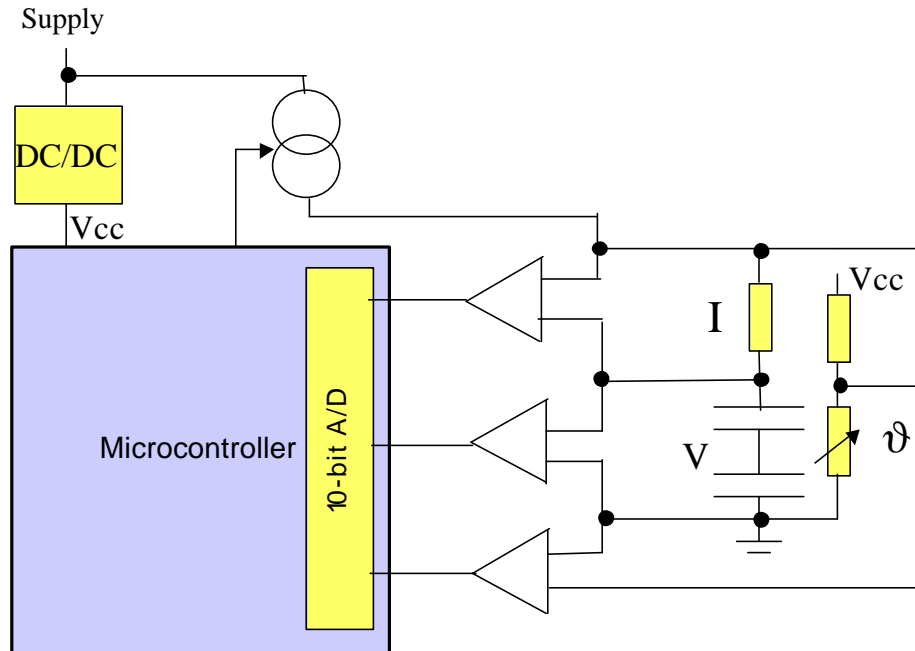


Figure 1: Example analog signal processing system

Now let's look at the output characteristics of each of our analog sensors in more detail.

One parameter, to be measured in our example system, is the charge current for the battery and the current flowing into the system powered by the battery. This is usually done using the shunt resistor and the trade-off that has to be made is minimizing the voltage drop over that resistor (to minimize heat dissipation and measurement error), while at the same time getting a reasonable resolution of the measured current.

Let's assume our charge/supply current that we want to measure is a maximum 1A,

however in certain modes current can drop below the 1mA level and we want to be able to measure this current as well. A 0.1Ω resistor in the supply line would result in a 0.1V voltage drop in the supply at 1A. Without amplification, our A/D would only see a maximum 100mV input voltage, giving us an A/D resolution of $100\text{mV}/4.88\text{mV} = 20.5$ (5 bits), which translates into a current resolution capability of $1\text{A}/20.5 = 48.8\text{mA}$, not very exciting. Amplifying the 0.1V voltage drop by a factor of 50, gives us a full 5V input voltage delta, resulting in a current resolution capability of 976.56μA. To measure the battery temperature, a negative temperature coefficient (NTC) resistor is used. A typical NTC resistor with

a β of 2988 and a value of $3k\Omega$ at 25°C , changes its resistive value from approximately $0.5k\Omega$ to about $7.7k\Omega$ over a temperature range of -10°C to $+50^\circ\text{C}$. If such a temperature sensor is used without additional amplification, a typical result

looks as follows (assuming the NTC's voltage change with temperature is linear, which of course it is not, but this is of no relevance for the point we are trying to illustrate.):

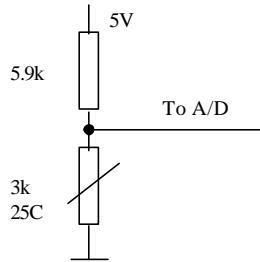


Figure 2: Temperature Measurement w/o amplification

The minimum voltage measured by the A/D at -10°C is:

$$5V * 0.5k\Omega / (5.9k\Omega + 0.5k\Omega) = 390.63mV$$

The maximum voltage measured by the A/D at $+50^\circ\text{C}$ is:

$$5V * 7.7k\Omega / (5.9k\Omega + 7.7k\Omega) = 2.831V$$

So the voltage delta over temperature is $2.440V$, utilizing only half of the A/D's full

dynamic range. Assuming, for a moment, the NTC's voltage change is linear, this translates into an A/D resolution of $2.440V / 4.88mV = 500$ (< 9 bits), which in terms of the temperature range to be measured would give us a temperature resolution of $60^\circ\text{C} / 500 = 0.12^\circ\text{C}$.

Taking the same configuration and amplifying the signal gives the following results:

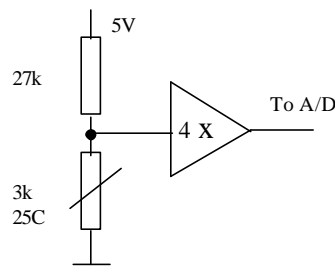


Figure 3: Temperature Measurement with Amplification

The minimum voltage measured by the A/D at -10°C is:

$$5V * 0.5k\Omega / (27k\Omega + 0.5k\Omega) * 4 = 363.6mV$$

The maximum voltage measured by the A/D at +50°C is:

$$5V * 7.7k\Omega / (27k\Omega + 7.7k\Omega) * 4 = 4.438V$$

The voltage delta over temperature is 4.074V. Assuming again, for a moment, the NTC's voltage change is linear, this would translate into an A/D resolution value of $4.074V / 4.88mV = 835$ (< 10 bits), which in terms of the temperature range to be measured gives us a temperature resolution of $60^\circ C / 835 = 0.072^\circ C$. So, by adding the amplifier into our system design, we have substantially improved our capability to do a finer grained temperature measurement.

Another parameter to be measured in our example system is the voltage change of the battery over time to determine charge termination. A rechargeable NiCd or NiMH battery has a typical cell voltage of 1.2V. The maximum voltage when fully charged is 1.45V, the minimum voltage when discharge is 0.9V. One of the methods to do charge termination is $-\Delta V$, i.e. at the end of charge the cell voltage drops slightly. Assuming a one cell configuration and no amplification, we utilize a resolution of the A/D of

$1.45V / 4.88mV = 297$ (< 9 bits). Amplifying the voltage by a factor of two we get $2.9V / 4.88mV = 594$ (< 10bit). This, in terms of the actual voltage we are trying to measure, gives us now, of course, twice the granularity: $1.45V / 594 = 2.44mV$

In summary, it can be stated that, with the traditional way of adapting the analog signals

to the full dynamic range of the A/D, external OpAmps are required. Depending on the application, the number of external OpAmps increases with the number of analog signals that need to be processed, resulting in a significant number of external components (OpAmps, resistors to set the amplification factors, and Vcc buffer caps) that increase system cost and required PCB real-estate area. Where PCB real estate area is a critical factor, it can be addressed in two ways: integration and sub-miniature packaging technology. Before we take a closer look at the possibilities that integration provides, we take a short look at new sub-miniature package technology available for microcontrollers: chip scale packaging.

Sub-miniature packaging

National Semiconductor started offering its microcontrollers in a new package, called Chip Scale Package (CSP) about a year ago. The package provides significant space saving when compared to traditional SO packages, while at the same time giving a larger number of I/Os per available mm^2 of board area. Especially when it comes to analog performance the package offers distinct advantages, as pin inductance and capacitance are minimized, due to the very short leads. At the same time thermal performance of the package is superior, due to a large heat conductive contact surface with the PCB. Compared with a 44 pin plastic quad flat package (PQFP), a 44 pin CSP package is 66% smaller ($7mm * 7mm * 0.8mm$). The 28pin CSP package is 68% smaller than a 28 pin TSSOP package and has about the same physical size as an 8pin SO, but give the user 24 I/Os versus the 4 I/Os of the SO8 package.

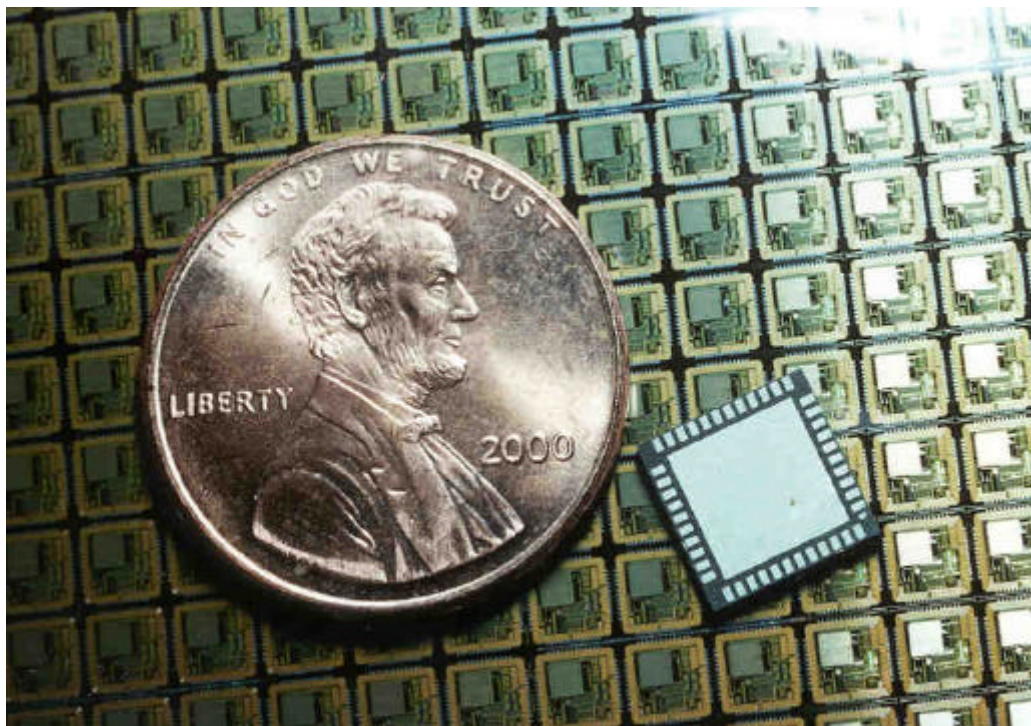
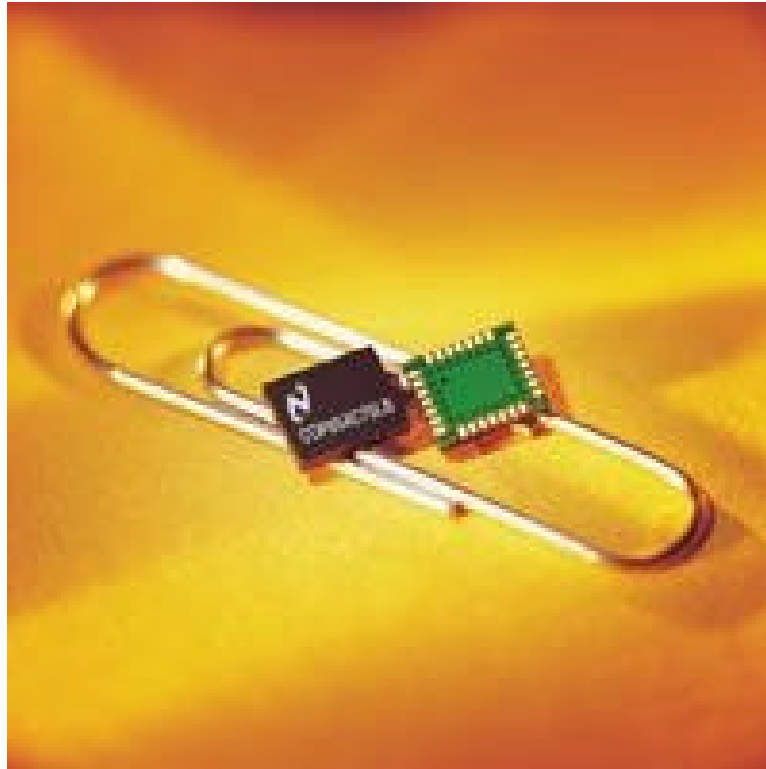


Figure 4: 44pin and 28pin CSP packages

Integrating the Operational Amplifiers

Figure 5 below illustrates an implementation on one of National Semiconductor's COP8 microcontrollers, where a programmable gain amplifier has been integrated between the output of the on-chip analog input multiplexers and the 10-bit successive approximation A/D. The OpAmp is software programmable with different gain factors of 1, 2, 5, 10, 20, 50 and 100. It can also be completely bypassed. The OpAmp is designed with high impedance, rail-to-rail inputs and rail-to-rail output. The input offset of the OpAmp can be as much as $\pm 7\text{mV}$. For this reason, circuitry for trimming out the offset, by the user, has been provided. By doing two measurements, one with an amplification factor of 1 and one with an amplification factor of 20, the user can trim the offset for both the N-channel and the P-channel input stage pairs in 0.5mV steps. At the end of this calibration process the input offset voltage is $500\mu\text{V}$ or less. The analog input multiplexer supports 7 inputs, of which

one is dedicated to the on-chip temperature sensor, providing a total of 6 external analog inputs. Rather than requiring up to 6 external OpAmps and passive components for designs utilizing all of those analog inputs, the implementation discussed here can handle all that functionality with the one integrated, programmable gain OpAmp.

A second "standard" operational amplifier is integrated with both inputs and its output accessible externally via the microcontroller's pins. This OpAmp can be utilized to implement an active filter function (e.g. noise filtering), an independent amplification or buffer stage, or other OpAmp function. In fact, the analog input multiplexer's output provides a path back to one of the input pins and a second input pin provides a direct connection to the 10-bit A/D input, so that another, external circuit can be inserted into the A/D conversion path. Using the second OpAmp for example, an active low-pass filter could be implemented that is applied to all of the analog inputs.

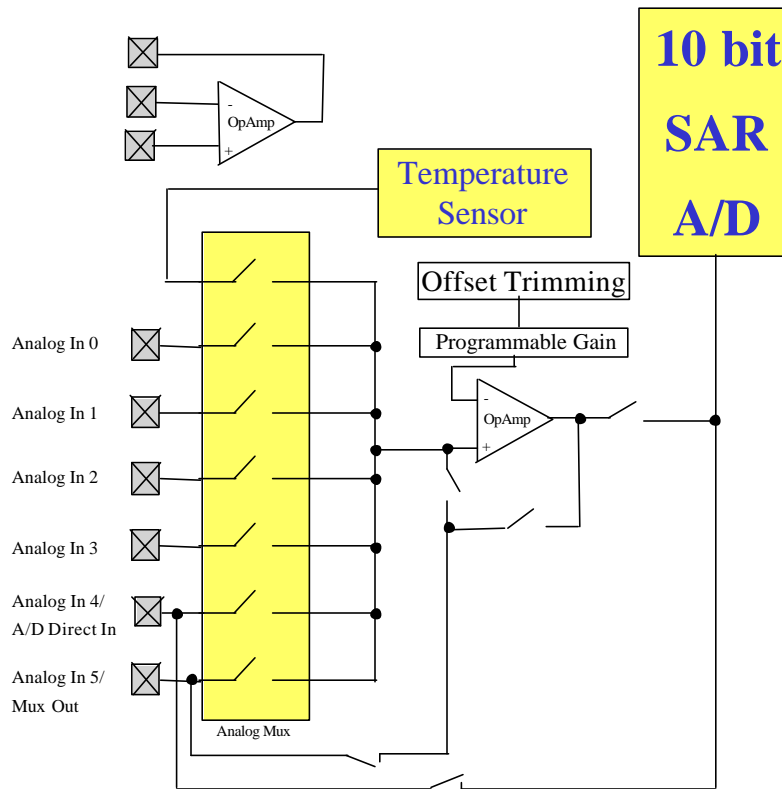


Figure 5: Block Diagram of On-Chip OpAmps and A/D Signal Path

Figure 6 shows the block diagram of the COP8AME microcontroller, a device with 8k bytes of FLASH memory, 512 bytes of RAM, three 16-bit multifunction timers - that support processor independent PWM, capture and event counter modes at a resolution down to 50ns, full duplex USART with integrated baud rate generator,

MICROWIRE™ 3-wire synchronous interface, brown-out reset, clock doubler, and dual clock function. The virtual EEPROM feature allows the user to dynamically allocate any desired amount of FLASH program memory for data storage.

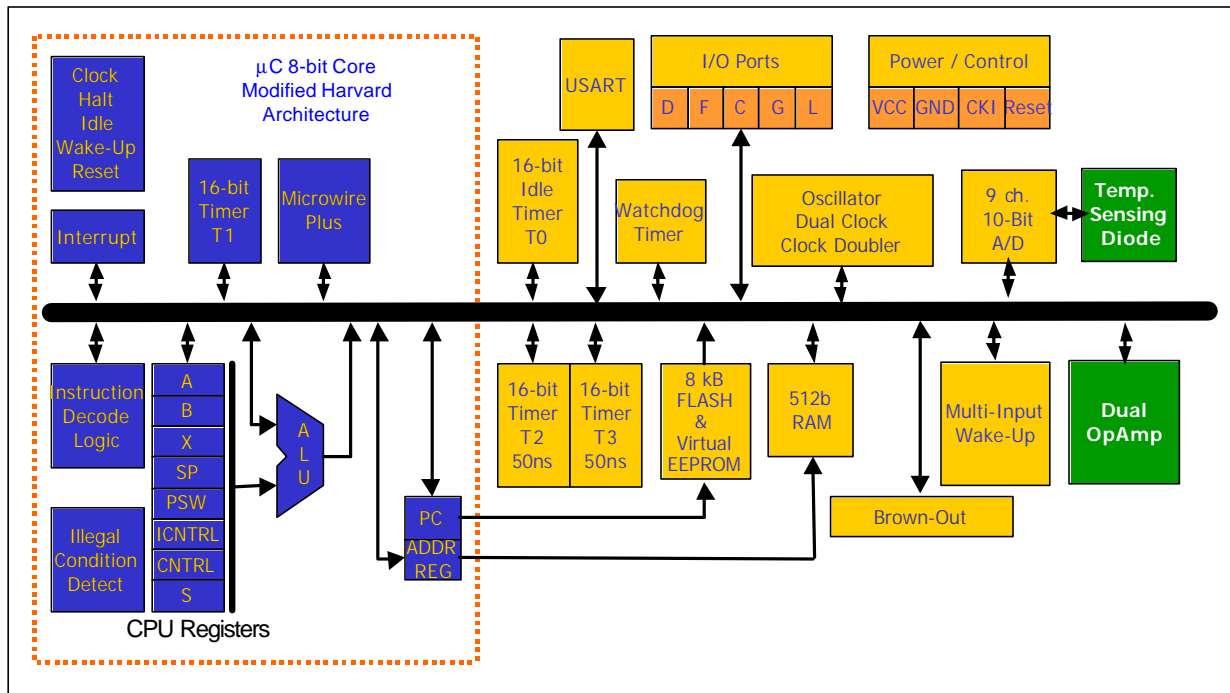


Figure 6: Block Diagram of COP8AME Microcontroller with integrated OpAmps and Temperature Sensor

System Development challenges

As beneficial as the integration of a programmable OpAmp and temperature sensors onto a microcontroller is in terms of external component count reduction, it creates a series of new challenges when it comes to application development, which usually involves the writing and debugging of code. The debugging of code usually involves a real-time, in-system emulator for the microcontroller, which allows the software developer to rapidly make changes to the code during the development phase and to check it out in the final end application. At the same time, the emulator provides powerful PC based debugging tools that support capabilities like examining on-chip registers, setting breakpoints at defined points in the program flow and looking at a

history of executed code (trace). Existing emulators today use one of two approaches, which for the purpose of this discussion we would like to call the “probecard” approach and the “debug board” approach.

The probecard approach uses a, so called, base unit that is connected to a PC via the serial port. The base unit contains the majority of the emulator electronics, with the exception of the emulation chip itself. The emulation chip is a special bond-out version of the actual microcontroller and is mounted on a separate small PCB, called a probecard. This probecard connects via a ribbon cable to the base unit and has different socket adapters at the bottom, which allow the probecard to be plugged into the actual target application board. What this means for analog signals that connect to the microcontroller is, that they have to travel from the target board, via a chip-socket, via another pin adapter socket on the probecard,

via the PCB traces of the probecard to the actual microcontroller pins. One can easily envision that this approach introduces additional noise, capacitance and inductance that make it impossible to obtain the same analog signal behavior during emulation as in

the final production version where the microcontroller sits directly on the target board.

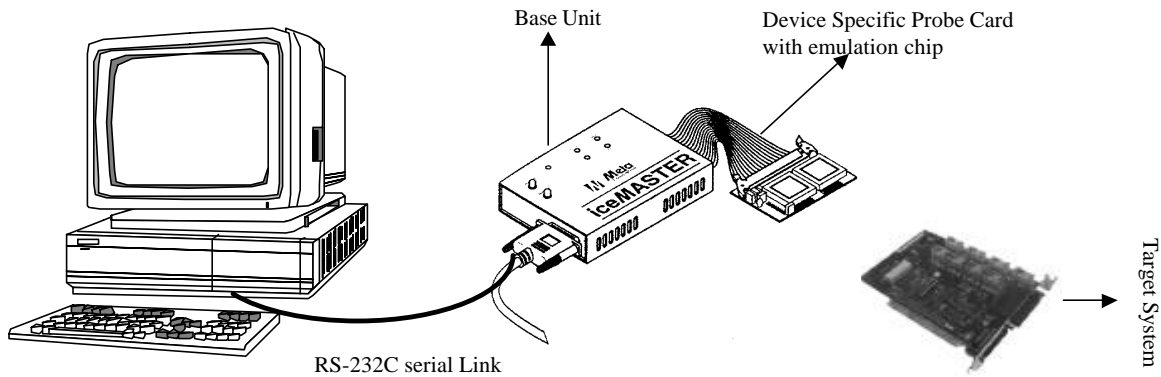


Figure 7: Probecard based emulator

The “debug board” approach combines all of the emulator electronics and the actual emulation chip onto a single, larger sized PCB. The connection to the target system is accomplished by ribbon cables, which provide, on one end, a connector that can plug into an actual chip package socket. This means that all pin signals for the microcontroller to be emulated are now routed via this ribbon cable that connects the target system with the debug board, including any analog signals to be measured by the microcontroller. This is bad enough if already amplified analog signals are routed via such a cable, as additional noise, capacitance and inductance are introduced that falsify the original analog signal, but it becomes even worse, if the amplification is now done within the microcontroller and

analog signals in the mV or even uV range are routed via such a cable- the results become unusable. Also, as analog functions, such as a temperature sensor, get integrated into a microcontroller, you want to measure the temperature at a specific point in the end application and **not** the ambient temperature of the debug board!

Another draw-back of both the probecard and debug board approaches is, that they require a specialized bond-out chip for each new microcontroller derivative. This makes the debug board or probecard specific to a given device, a new probecard or debug board needs to be purchased if new family members with new integrated features are introduced.

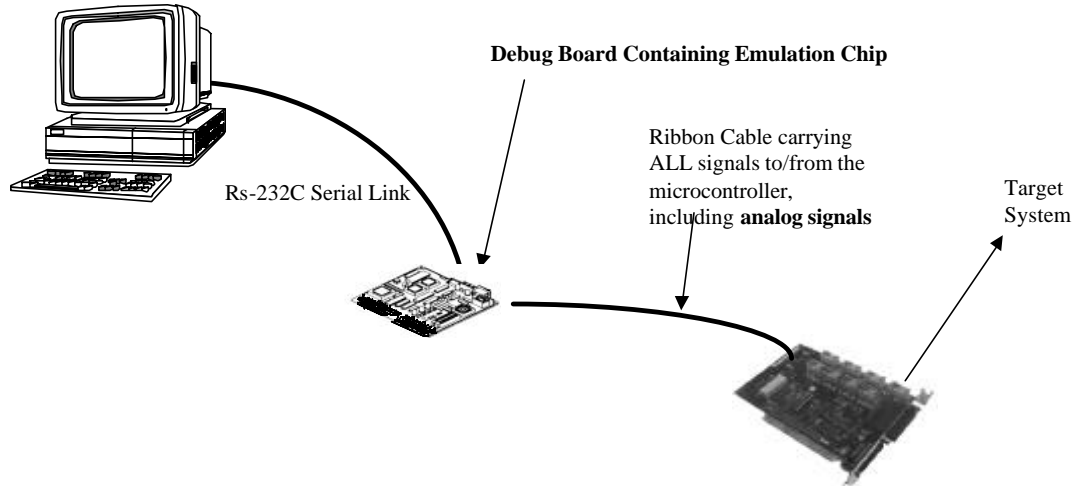


Figure 8: Debug Board based emulator

The solution to the above problems comes with the advent of FLASH program memory and a new approach in microcontroller emulation pioneered by National Semiconductor: 100% precise analog emulation. The electrical erase/write capability of FLASH program memory, in combination with emulation support circuitry and code embedded in the microcontroller, each chip has the capability to be its own emulator. This means that a sample of the microcontroller chip, regardless of package, can be directly soldered into the target application. This, of course, means that now the microcontroller chip “sees” the exact same analog environment as the finalized application. Temperatures, for example, are measured where they are supposed to be measured and voltages in the μV to mV range that need to be amplified only need to travel

the shortest possible distance. The only connection now required to enable PC based software debugging are four fully digital I/O pins of the controller, whose digital functions are “recreated” on a small piece of hardware that sits between the PC and the target system. This connection from the target board to the base is accomplished via a small 2×7 header on the target board that, at the same time, can be used to program and re-program the controller in-system without the need for any additional programming support circuitry. As mentioned earlier, the chip is its own emulator, so that when a new family member is introduced, no new emulator or probecard needs to be purchased- all that is required is a sample of the new chip and eventually a software update for the debugger software that runs on a PC.

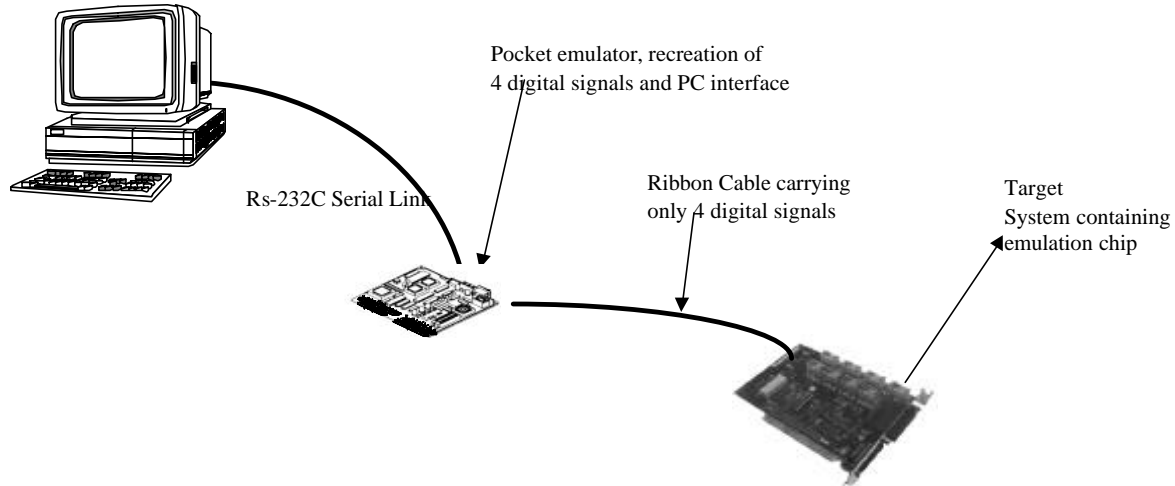


Figure 9: The chip is it's own emulator, 100% precise analog emulation

Conclusion

The integration of a programmable gain amplifier into the on-chip A/D signal path of a microcontroller, can provide significant system cost savings. A number of external operational amplifiers and passive components, which are required today to utilize the full dynamic range of an A/D converter, can be eliminated. This in combination with new sub-miniature packaging technology, not only reduces system cost, but at the same time can shrink the required board space by as much as 66%. However, as more and more analog functions are integrated into microcontrollers, traditional microcontroller emulation methodologies, which were constructed for "digital" systems, become less and less suitable for the task at hand. Only if the actual microcontroller chip has the capability to act as its own emulator, can analog signal integrity be maintained throughout the development phase all the way to production.