Getting Started In PC-Based Data Acquisition

By Roger Lockhart
DATAQ Instruments

Originally published in the June, 1995 special Data Acquisition Issue of Sensors Magazine

I hate to use clichés, but suppliers of data acquisition system (DAS) products need a reality check. Too commonly educated in electrical or software engineering, we pack our sales material with technical jargon only persons with a similar background can understand. This is great for impressing the competition, but we leave many potential customers drifting in our wake. Too often, the customer for DAS products is a mechanical or chemical engineer, a seismologist, a medical researcher, or another who works in one of hundreds of other disciplines not even remotely related to computers and electronics. Turnabout is fair play, so I propose some rhetorical questions for the EEs in the audience: When was the last time you had to spec a pinion gear for your car; or concern yourselves with the intricate details of even one of the many forms of electrophysiology?

The goal of this article is to relate DAS concepts and alternatives to mere mortals. It uses a divide-and-conquer approach to segment the selection of a complete DAS product into a more easily manageable set of independent yet related subsystems. These are signal conditioning, A/D converters, software, and the computer.

Signal Conditioning

That there are analog signals to be acquired is a basic premise of DAS applications. It's also a good bet that a transducer exists somewhere in the chain that converts a physical phenomenon into a continuously variable (analog) voltage that is directly proportional to the magnitude of the phenomenon. For example, you use thermocouples to convert heat, and strain gauges to convert stress and pressure into voltages directly.

You may also use a magnetic or optical sensor to measure RPM, or an LVDT to sense distance. These examples produce intermediate signals that must be further conditioned to derive a signal suitable for connection to a DAS. In the case of RPM measurement, the sensor produces a pulse train whose frequency is proportional to the speed of rotation. The LVDT produces an AC signal that must be demodulated.

There are more examples, but you can begin to get the idea that there are many ways to measure an equally wide array of physical quantities. For all these alternatives there exists a signal conditioner that's right for the application. You can reduce the selection by applying these common sense guidelines:

Buy transducers with built-in signal conditioning when possible. Especially for pressure measurements, transducers featuring built-in amplifiers have been gaining in popularity. These products, requiring only a single 5 or 10VDC power source, produce a high-level output suitable for direct-connection to a DAS. You'll pay more for such a transducer, but the savings in connection time and elimination of an amplifier could more than compensate for this disadvantage. Later on in this section I'll talk about the direct connection of strain gauge and thermocouple signals to a DAS and the considerations involved with this.

Determine how much flexibility you need. In applications where integral amplifiers are not appropriate or not available (I've not seen one for thermocouples, for example), consider the kind of measurements you'll be making. It helps me to generalize using two broad classifications: production and research.

A production application is where the same measurements are made day after day over a long term. For example, perhaps you need to monitor several process points on an extrusion line to ensure product quality or enhance productivity. In such an application, the need for flexibility is slight. You're connecting the DAS to a well-defined range of signal types with similarly well-defined measurement ranges. Since you pay for flexibility, you can cut your costs by selecting a signal conditioning approach that targets the measurements you'll be making on a per channel basis.

An excellent solution for production measurements are the 5B Series products manufactured by Analog Devices and Dataforth. These products feature low-cost modules (about $200 per channel on average) that connect to a common back plane, usually in 16-channel groups (see DATAQ Instruments' DI-500 Series products). Each hermetically-sealed module is designed to measure a specific function over a defined measurement range. As a consequence, they don't have any balance pots or sensitivity switches. One module, for example, accepts a J-type thermocouple for temperature measurements over a 0 to 760°C range. Another is designed for temperatures within a -1000 to +300°C range. There are over 40 modules in the 5B family and each supplies a high-level 0 to 5V or ±5V signal suitable for connecting directly to a DAS.

Research applications are the complete opposite of the production counterpart. The term "research" suggests an investigation of the unknown resulting in the implied need for flexibility. Research measurements are short term and varied in nature. The measurements you are making today will not be the same as those you'll be making tomorrow or the next day. Picture a computer on a cart that is wheeled to one measurement site and the next, perhaps used by more than one person for troubleshooting. Or an application where the measurement functions are well defined but signal levels can vary widely.

In this setting, where you need flexibility to adapt to ever-changing measurement demands, consider using an amplifier with features designed for adaptability. Like the 5B modules, such amplifiers are supplied by function (i.e., thermocouple, RTD, DC volts, rms, high voltage, etc.). However, they differ from the 5Bs in that one amplifier can connect to a wide range of signal amplitudes. One DC amp, for example, may have a measurement range from millivolts to hundreds of volts. Other useful features may be selectable high-pass and/or low-pass filters for noise elimination. Or zero suppression to enhance the amplifier's ability to measure dynamic signals riding on large, static offsets.

There are many vendors of so-called research-oriented amplifiers. One I recommend often is Gould Instrument Systems, Inc. Gould manufactures many amplifier lines, but one relatively new line is their 6600 Series products. These feature a wide range of plug-in amplifiers for industrial and medical applications, each designed to measure a specific function. Each amplifier in the line plugs into an amplifier cage allowing you to mix and match the amps to address almost every conceivable application. The cage contains and powers up to 8 amps. Gould even supplies an optional, built-in ADC card that connects directly to the printer port of any PC. It provides a convenient method for digitizing, storing, and analyzing amplifier signals with a PC.

Some common signal conditioning pitfalls. Now that we've covered two broad classifications of signal conditioners, its appropriate to examine some common pitfalls.

Consider the need for isolation when choosing a signal conditioner. This feature plays an important role when making off-ground measurements — where a potential difference (often referred to as a common mode voltage) exists between the ground (or common) of your signal and the common of your DAS.

For example, consider the need to make measurements on the electrical system of an automobile. In this environment, you have a multitude of electrical activity ranging from a whirling alternator to the thousands of volts generated by the engine's ignition system — all in a self-contained system resting on rubber tires. What's the relationship between the auto's ground and the power ground of your PC or DAS? There isn't one. Connect one lead of a hand-held DVM to the computer's chassis and the other to any so-called "ground" on the car and you'll measure a voltage. Such potential differences, ranging from millivolts to hundreds of volts depending upon the measurement, can destroy your ability to make accurate measurements. Off-ground measurements attempted in the absence of isolation can cause noise that can completely swamp the signal of interest. But worst-case off-ground measurements can actually cause sparks to fly the instant you connect your signal to a non-isolated amplifier if sufficient voltage and current exists in the off-ground portion of the signal.

Isolation allows the input stage of an amplifier to "float" with the off-ground portion of the signal of interest. The amplifier maintains the potential difference that exists between its "floating common" and the common of the signal, preventing current from flowing through the connection. The probability of noise is greatly minimized, along with the hazardous and humbling experience of electrical fireworks.

Although isolation is not a universal requirement, you should be aware of the circumstances where it will be necessary and factor these into your decision-making process. All the amplifiers mentioned above provide isolation.

You should use signal conditioning to your advantage in areas of dynamic signal analysis and avoid the temptation to let the computer do the job. For example, to calculate RPM from the pulse train generated by a magnetic pickup, or to calculate rms and frequency deviation from an AC voltage or current waveform. Although the computer and supporting DAS products have the capacity to perform this type of analysis, the realities of real time data acquisition dictate otherwise. Of all the tasks performed by computers in the modern world, real time data acquisition ranks among the most demanding. The time pressures placed on a processor for data acquisition, display, and storage can become burdensome at faster sampling rates and will limit the effectiveness of even the fastest machines. With this in mind, you should exploit opportunities to minimize sample rates whenever possible.

Let's take the example of engine RPM a step further. You have two choices when making this measurement. You can connect the DAS directly to a pickup, and sample at about 15,000 samples per second to provide adequate time resolution. Then write or buy a piece of incredibly efficient computer code capable of performing a real time frequency calculation. Be satisfied with just this one measurement, however, because with the computer chasing its tail in this manner there is precious little time left for anything else. There are DAS products capable of real time frequency measurements using on-board timers, but few if any allow simultaneous analog measurements.

You can scour the universe of DAS products, minimizing your options and maximizing the price tag in the process. Or you can simply place a tachometer amplifier between the pickup and your DAS. Essentially a frequency-to-voltage converter, the tach receives and conditions the rather noisy pickup signal and delivers a voltage directly proportional to RPM to the DAS. Sample this signal at a rate that's a function of how fast you expect RPM to change with respect to time — I'd say 10 samples per second would be fine. In this example, the application of a signal conditioner saved over 3 orders-of-magnitude in sample rate, and expanded the universe of acceptable DAS solutions to include almost any product. And in raising the computer from its knees you have the potential to make many other simultaneous measurements.

In my experience dealing with DAS users, there seems to be a lot of confusion over when to use single-ended (SE) as opposed to differential (DI) input configurations. Most DAS products allow you to configure analog inputs for one or the other. And many users, having heard one place or another that DI inputs are preferred, universally use this configuration. The following statement applies to most DAS applications: If the DAS is connected to the output of a signal conditioner, including those built into transducers, use a SE input. In the overwhelming majority of applications where a signal conditioner is present, connecting its output to a differential input on the DAS buys you nothing in terms of performance, cuts down your channel count, and can even cause measurement problems.

Having said that, when should you use a differential input? Whenever you connect to a balanced signal source, or have good reason to suspect the presence of a common mode voltage as described above. You encounter balanced signal sources when connecting directly to strain gauges, or grounded thermocouples. Connecting these to a SE input configuration essentially short circuits the signal source leaving you measuring nothing but noise. A thermocouple that is grounded (e.g., by attaching the metal thermocouple junction directly to a metal boiler that is grounded) is short circuited when one of its two leads is connected to a SE input which by definition has one side also connected to ground. However, connecting the thermocouple to a differential input removes both leads from ground and allows the measurement to be made.

The presence of a common mode voltage (CMV) also dictates the use of a differential input. Again, connecting one leg of such a signal to the grounded half of the SE input causes current to flow in the circuit resulting in degraded signal fidelity. Using a differential input removes both leads from ground and preserves the potential difference between the DAS input and the signal source. But be wary of the CMV's magnitude. Most DAS products can tolerate only a few volts of CMV. Exceed the spec and you're faced with the need for a fancy resistor divider scheme. Better yet — here it comes again — insert an isolated amplifier between the signal source and DAS and just make the measurement.

Another consideration is to ensure that the output voltage matches the full scale input voltage of the analog to digital converter (ADC) you'll be using. I'll talk more about this in the following section, but the important point to make while we're still talking about signal conditioners involves the selection of gain. When measuring low-level signals that require the application of high gain factors, you should always take the majority of the gain as close to the transducer as possible. Assume you're measuring a transducer that produces a low-level signal of 1mV (0.001V) at full scale. Assume further that you want to connect the signal to an ADC that requires 10V full scale, implying the need for a gain of 10,000 (10/0.001) which you can take anywhere between the ADC and the transducer. Your ADC has programmable gain factors of 1, 10, 100, and

1000, and you have two external amplifiers capable of delivering gains of 10 and 1000 respectively. Which amplifier do you use?

Either amplifier, used in tandem with the ADC's gain factors will produce the desired gain of 10,000. But when you know that all amplifiers are also sources of noise, and that amplifiers can't distinguish noise from signal, it makes sense to use the external amplifier with a gain of 1000 and choose a gain of 10 at the ADC. The other choice would produce a smaller and less desirable signal to noise ratio since any noise generated by the lower gain external amplifier is amplified by a factor of 1000 by the ADC.

The final thought for this section is directed to those already making the measurements they need with traditional instruments (DSO, chart recorder, tape recorder, etc.), but who are seeking a way to make the same measurements using a DAS for flexibility and lower cost. Don't change anything from the signal conditioners forward to the transducers you're using. Instead, place the DAS in parallel with the traditional instrument. In virtually all situations, the instrumentation in which you've already invested to support your traditional instrument will work fine with the DAS. Such an approach will save you time, funds, and frustration. It will insulate you from the need to learn a completely new signal conditioning scheme. And access to your traditional instrument will provide you with a fallback position and a reference point as you become familiar with the new, computer-based instrument.

Selecting an Analog to Digital Converter (ADC).

Now that we've properly conditioned the signals we want to acquire, its time to examine the elements that govern your selection of a suitable ADC:

A fixed or mobile application. Consider how you will use the DAS. Are the signals you need to connect to centrally located? Or will you need to move the DAS from one location to another? In the past, your options were limited in terms of mobility. Most DAS solutions were limited to ADCs in the form of plug-in cards that needed to be inserted into expansion slots of large, clumsy desktop PCs. A wheeled cart was the extent of your portability options.

Two events have changed this picture forever: The emergence of powerful and low-cost battery-powered laptop and notebook computers; and the subsequent availability of ADC solutions designed to be used with these alternative products.

Where DAS portability is a consideration, your choices fall into two categories: parallel port products; and those that connect to the PCMCIA slots provided by some computer models. Parallel port DAS products connect to the parallel (or printer) port available on any PC — from notebooks to desktop models. PCMCIA products require a special port available on a small but growing number of computers. Parallel port DAS products offer the advantage of being able to connect to any PC, and the potential for a self-contained power source in the form of rechargeable batteries. The latter feature makes the DAS independent of the PC's power source and extends the operation time of both in purely portable, battery-operated applications. PCMCIA products offer generally higher sample rates since they connect directly to the bus of the computer's processor. They also offer a sleek, low-profile package that inserts directly into the mating slot of the host PC. However, this advantage is compromised somewhat since the fragile connectors of the PCMCIA design require the use of an external signal connection box in most applications.

In situations where you know the DAS will be in a fixed application, plug-in cards still rule the day. They offer the most variety in resolution and sample rate, and are available from a wider array of vendors. Since batteries and packaging add cost, plug-in cards offer generally better price/performance.

Let’s talk sample rate. There are many issues here, but the best place to begin is with the infamous Nyquist theory. I’ve labeled it “infamous” because I don’t believe any DAS concept is more widely used and less understood. The Nyquist theory states that you can reconstruct any bandwidth-limited analog waveform from its digitized equivalent if the sample rate is at least 2 times the highest frequency component. Stated another way, you cannot apply the Nyquist theory to the waveforms you sample unless you limit their bandwidth to less than one-half of your sample rate. The confusion stems from the interpretation of the theory where it is easy to conclude that if the signal of interest is “about 100 Hz” you don’t need to sample any higher than 200 Hz. Such conclusions lead to disappointing results at best.

We were all exposed to this danger as kids. Remember watching western movies where the wheels of a speeding stagecoach seemed to be rotating impossibly slow, or even going backwards! This effect is an alias, or ghost frequency caused by the slow shutter (or sample) rate of the camera relative to the fast rotational speed of the spoked wheels. The same phenomenon can haunt your best data acquisition efforts in the form of signal “ghosts” if you don’t take steps to prevent it.

The operative phrase of the Nyquist theory is bandwidth-limited and there is only one way to bandwidth-limit your signals: Apply an anti-alias filter to the analog waveform before digitizing. An ideal anti-alias filter is a low-pass filter that passes all frequencies below a defined corner frequency and blocks all frequencies above it. In theory, if we apply an anti-alias filter with a corner frequency of 100 Hz to the waveform mentioned in the previous paragraph, then we can safely sample at 200 Hz. In practice, we’d have to sample a little higher since no filter is ideal — it will always pass some frequencies a limited range beyond the theoretical corner.

Purists will claim that all signals you acquire should first pass through an anti-alias filter. The reality is that anti-alias filters are expensive, bulky and simply not required for every application. Most uses of a DAS are for static, near DC signals (i.e., temperatures, pressures, flows, etc.) where use of a differential amplifier or a simple low pass filter will reduce unwanted AC to an acceptable margin. DATAQ Instruments’ DI-400 Series instruments, for example, offer signal averaging that simulates a low pass filter suitable for low frequency applications and eliminates the need for external components. Higher frequency applications — especially those where the frequency content of the signal needs to be examined like in shock and vibration applications are the domain of true anti-alias filters. For example, if you’re interested in a frequency range of 500 Hz per channel for 10 channels, it probably makes sense to apply anti-alias filters and sample at about a 15,000 Hz throughput rate. Without the filters, you’re faced with a throughput rate upwards of 100,000 Hz to minimize the effects of alias frequencies.

Channel skew is another issue that falls within the sample rate category. This undesirable effect is a time-shifting (or skewing) of sampled data points that otherwise should be time-aligned. Let’s assume you are sampling 32 thermocouples connected to various points on a boiler and you need to sample them once every second. Sequential sampling of these points results in a time skew equal to 31 seconds between the first and last channel — not a desirable situation. A technique called burst sampling, if supported by your ADC, would allow you to sample all 32 thermocouples at the maximum rate of the converter once every second. Assuming your ADC has a maximum sample rate of 50,000 samples per second, the maximum time skew between the first and last channel reduces to 620µs (31*1/50,000). The need to minimize channel skew is paramount in virtually all DAS applications which dictates that you should only consider ADC products that support burst sampling.

The final thought on sample rate is a cautionary one. Don't assume that the maximum sample rate spec for an ADC product applies to expansion channels. Most ADC products offer 16 built-in channels. Channel expanders are added to the products, usually in 16- or 32-channel increments, to increase the number of connected points. Some ADC products that spec high speed sample rates for the built-in channels are not able, because of their design, to apply these rates to expansion channels. Indeed, products of this type sample expansion channels at a relative crawl that can severely penalize your flexibility. The reasons for this get technical and are beyond the scope of this article. But if you are considering the purchase of an ADC where expansion channels are required, ask the manufacturer if there is a sample rate penalty for the additional, external channels. The answer should be a simple "yes" or "no".

Many ADC products offer a built-in amplifier that can apply gain to the analog signals you acquire. This capability offers greater flexibility in adapting to the wide range of signal levels that generally accompany real world DAS applications. Gain is provided in two forms: switch-selectable gain; and programmable gain. Switch-selectable gain generally requires that the same gain factor be applied to all acquired channels, which compromises your flexibility in situations where signal levels vary enough to require a different gain factor for each. ADC products with programmable gain allow you to apply gain on a channel-by-channel basis. For this reason, ADC products with programmable gain are generally more popular.

Whether built into the ADC or applied externally, gain is used for one reason: to amplify signals to a level as close as possible, but not exceeding the volts-full-scale (VFS) rating of the ADC. A gain factor of 10 should be applied to a 1 VFS signal to match it to an ADC with a 10 VFS input range. Assuming the ADC has a resolution of 12 bits, and you don't apply the gain, your measurement is yielding only slightly better than 8-bit resolution — over 10 times less than what the ADC is capable of delivering.

At the other extreme, be careful not to over-specify resolution. On countless occasions I've talked to users who want to connect a pressure transducer with ±1% inaccuracy to a 16-bit ADC. The benchmark inaccuracy of the 16-bit ADC is 0.0015% while that of a 12-bit product (the workhorse of the industry) is 0.024% — 41 times better than the signal they want to measure. Higher resolution ADCs cost more, increase the noise sensitivity of the measurement, and burden the design criteria of all DAS elements placed ahead of them. Unless you genuinely need to measure extremely minute signal variations (i.e., greater than 1 part in 4096 for a 12-bit ADC), selecting a higher resolution ADC should be avoided.

Software Selection

The engine that drives DAS applications is software. It is the weak link in the DAS chain. Software can enhance the performance of strong hardware, or amplify the deficiencies of weak hardware. In many if not most instances, software is the difference between a successful and unsuccessful data acquisition project.

The universe of almost endless software alternatives reduces to just two fundamental forms: programmable or turnkey of which all software products are one, the other, or a hybrid containing elements of each.

Programmable approaches to DAS software usually involve the use of a programming language (e.g., BASIC, Visual BASIC, C, Pascal, etc.). Most ADC manufacturers include a software development kit (SDK) which allows your language-of-choice to access the ADC hardware and thus perform various data acquisition tasks. The advantage of this approach is flexibility. A limitless number of data acquisition tasks can be programmed and completed in precisely the manner needed by the users. The disadvantage is complexity. The person writing the software needs to be familiar with the complex and often arcane characteristics of the programming language. The SDK contains a library of 20 or more functions pertaining to the use of the ADC that further increases complexity and the slope of the learning curve.

These disadvantages are resolved by a graphical programming language (GPL). It eliminates the need for a conventional programming language and function library in favor of its own collection of objects — graphical icons that each perform a specific data acquisition task. By manipulating the objects on the screen, you can create a data acquisition application without delving into the particulars of a programming language and the ADC’s function library. However, often referred to as an aid for non-programmers, these products are more accurately described as a programmer’s aid. If you’re not familiar with the fundamentals of programming, you’ll find it difficult to apply a GPL with satisfactory results. Also, GPLs come with a lot of excess baggage. The finished product is so code-laden that applications often run very sluggishly. Fast and even moderate sample and display rates can be compromised.

The second class of DAS software is turnkey products. In contrast to their counterpart, this alternative does not require programming of any kind. You’ll be acquiring and, depending upon the product, analyzing data minutes after installing the software. Turnkey products leverage the experience and expertise of professional software engineers and insulate you not only from the need to design the software yourself, but also from the need to support the product. You also gain access to an almost endless stream of product upgrades that can further enhance the way you can acquire and analyze data.

However, as most who have spent time looking at turnkey software products will attest, you probably won’t find exactly what you’re looking for. All DAS applications are unique. Each requires it own special approach to data display, storage, or analysis to satisfy the best expectations of its users. A turnkey product’s solution may not be compatible with the way you need to approach your particular data acquisition problem. To address this need for customization, many turnkey products allow you to attach your own code to modify the way the software works. You start with the software’s ability to program the ADC, acquire data, display it, record it to disk, etc. Your code can take over from there by perhaps generating a readout of minimum, maximum, or average values. You can even produce real time control processes by toggling digital I/O ports or scaling an analog output in response to data acquired by the turnkey software. Of course, you’re back to programming again and the advantages of modifying the operation of a turnkey package must be weighed against the time and expertise required to do so.

Which software approach is best for you? Given the range of user sophistication and DAS application complexity, any answer to this question must be generalized. I opened this section with a discussion of the key role played by DAS software. Given this premise, I am amazed at the general indifference paid to software by some users when assembling the component parts of a DAS. "I'll buy the hardware, but write the software myself" is the way the thought pattern goes. I'm not saying that users should not write data acquisition software. But in any successful data acquisition project where the user wrote the code himself, either the application was straight forward and perhaps even simple, or the author was very capable. Either way, it's a good bet that a large and perhaps inordinate amount of time was devoted to its development and on-going support.

In my opinion, writing your own code should be an effort of last resort after evaluating all known sources of commercial software. Begin by objectively looking for a turnkey product. Experienced programmers may want to augment this search by looking into one of the several GPL options available to them. Consider dropping a few whistles and bells so you can justify buying that software package that's almost perfect.

Finally, don't fail to purchase a software package because it's too expensive compared to the spreadsheet you bought last month. Remember that the market size for DAS products is small compared to that for business software products. Prices for data acquisition software must be higher to provide the company a reasonable return on their investment in program development.

Those who are able to justify the purchase of commercial software will find that they solve more problems than they create. They're able to put data acquisition hardware to work much more quickly, and support is handled at the developer's time and expense.

Selecting A Computer

This article would not be complete without devoting some discussion to selecting the computer you'll connect to your DAS. In the parlance of the industry, your first decision is to select a platform. You have two fundamental choices: IBM PC and compatible, or a Macintosh. Far and away the greatest number of data acquisition systems are found on the IBM PC and compatible platform, while the Mac never quite garnered widespread support. Unless you have an extreme bias toward the Mac's elegant user interface, use a PC platform. It will dramatically multiply the number of data acquisition hardware and software vendor options available to you.

Next, you'll have to select the CPU that powers your PC. Currently, the dominant CPU is the 80486, although given Intel's announced price reductions on their Pentium, you can expect this engine to start coming on strong within the next year. Both the 286 and 386 processors have faded into obscurity and are no longer offered in new designs. I suggest a new computer for anyone with a 286 machine, but you may still squeeze some life out of 386 designs depending upon the requirements of the data acquisition task and the hardware and software you'll be using. But it is important that you look ahead when choosing other elements of the PC.

For example, you may know that Microsoft Windows 3.1 consumes about 15 Mb of hard disk space. But did you know that the next release, Windows 95, will more than triple in size to 50 Mb? The hard disk space required to hold applications software is exploding as well. A fully installed version of Microsoft Word can consume 25 Mb of disk space. Since this trend shows no sign of abating, you would be wise to purchase as much hard disk capacity as you can reasonably afford. A 500 Mb hard disk in modern computing is not unreasonable.

Finally, since Windows is the dominant user interface on PCs, a graphics coprocessor will dramatically accelerate the speed at which it operates. Windows is a graphics-intensive environment that places a high demand on the computing power of the CPU. DAS applications multiply this demand through their typical use of waveform graphics, much of which is generated in real time.

In large part, the question of how much computing power you'll require can be answered by the developer of the software you've chosen. A call to them or a glance at their data sheet should provide the necessary insight.

Roger W. Lockhart is Vice President of DATAQ Instruments, Inc., a developer of PC-based data acquisition hardware and software products. He may be reached at 241 Springside Drive, Akron, OH 44333. He may be reached by telephone at 330-668-1444, or by E-mail at rlockhart@dataq.com.

Figure 1. Fixed range signal conditioning modules such as the 5B Series from Analog devices and Dataforth are an excellent, low cost choice for a wide range of industrial measurements.

Figure 2. Plug-in amplifiers such as the 6600 Series from Gould Instrument Systems, Inc. offer a wide measurement range per unit, and a broad selection of industrial and medical functions. An optional, built-in ADC converts the unit to a stand-alone DAS capable of connecting to the printer port of any PC.

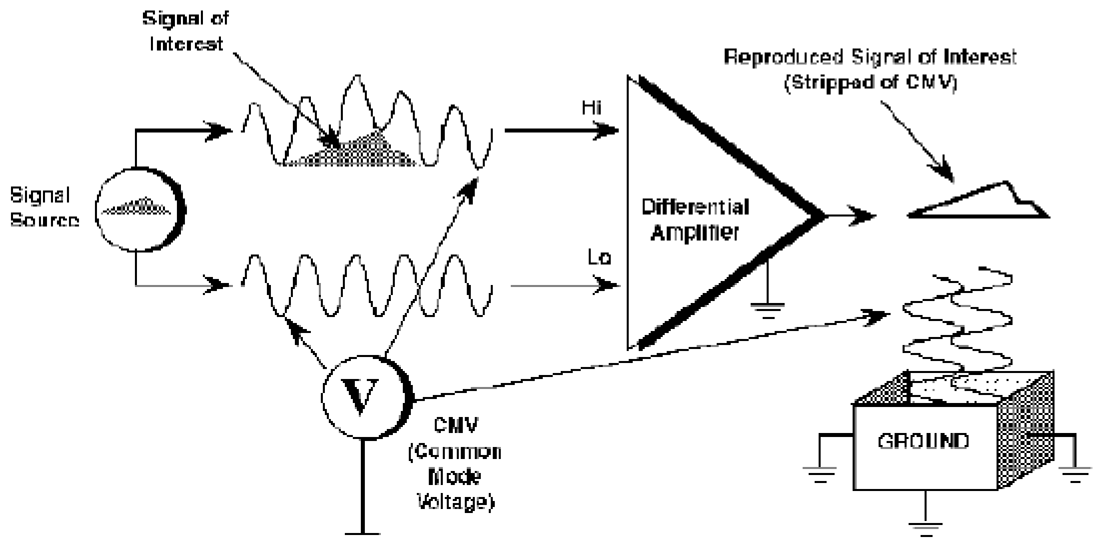

Figure 3. A differential amplifier rejects common mode voltages and allows the measurement of signals which would be impossible with single-ended configurations. Excessive CMV magnitudes may require the addition of an isolation barrier.

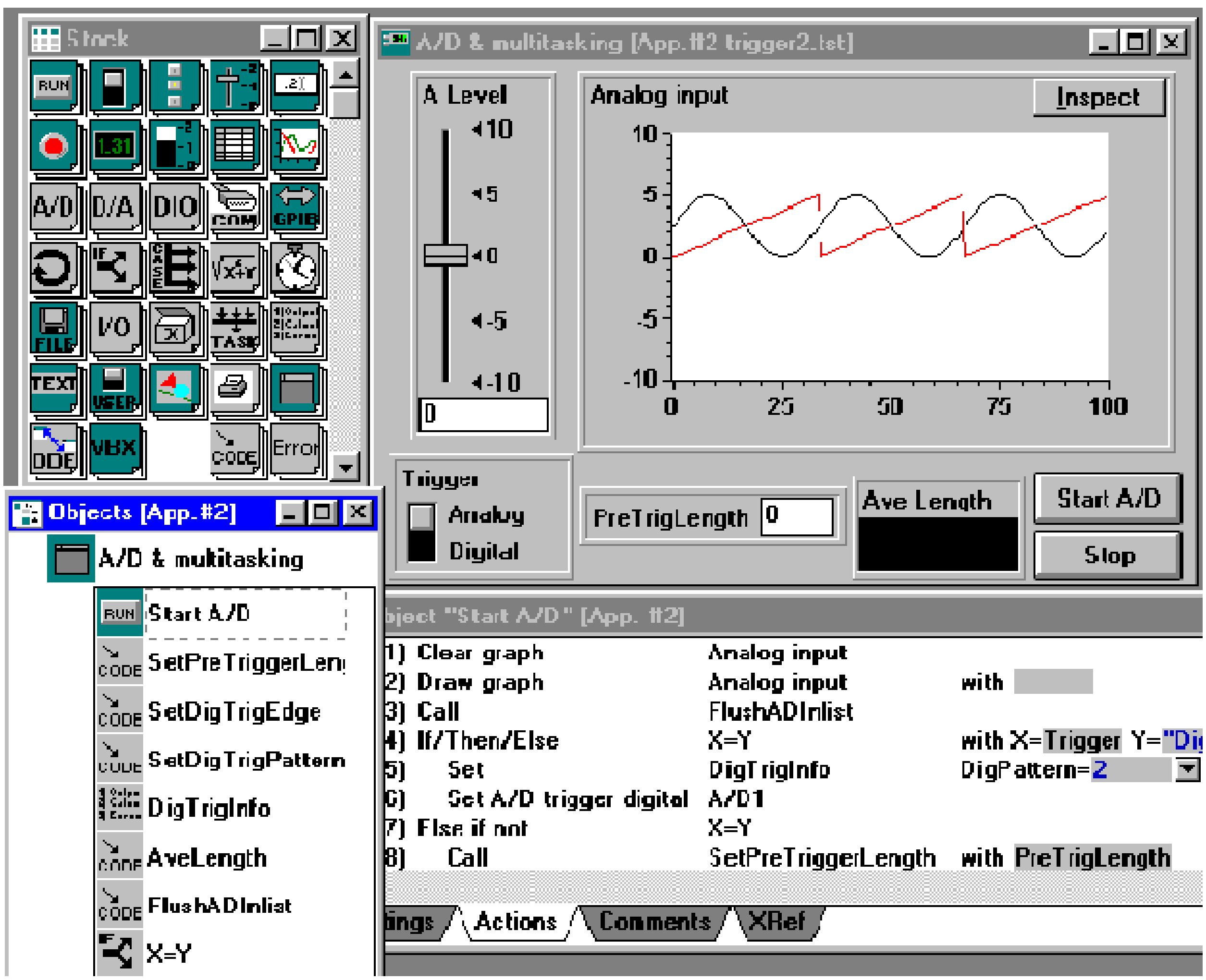

Figure 4. An example of a graphical programming language (GPL) is TestPoint by Capital Equipment Corporation.

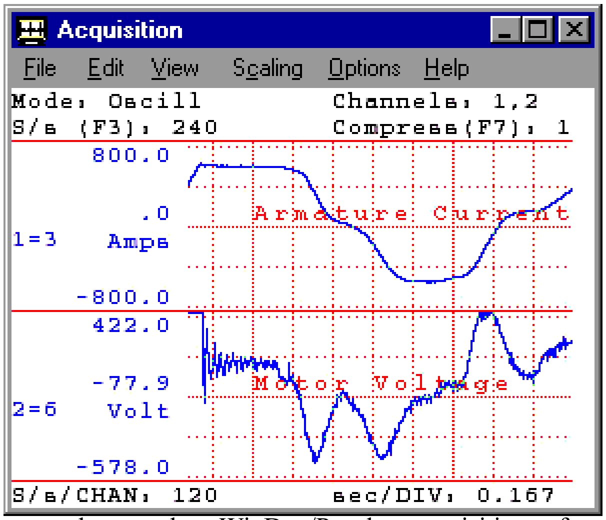

Figure 5. Turnkey DAS software products such as WinDaq/Pro data acquisition software and WinDaq Waveform Browser playback, and analysis software by DATAQ Instruments require no programming and allow immediate, productive use of data acquisition hardware.