A microcontroller-based data acquisition system for the Mac

by Francis J. Deck

In the September/October issue of the *SciTech Journal*, I presented a system for programming single-chip microcontrollers with a Macintosh[1]. A microcontroller (MCU) can be used to simplify the design and construction of sophisticated Mac peripherals. For instance, Mike Cook's recently upgraded serial input/output board (see Serial I/O Interface Kit, page 11) interfaces an MCU to the Mac serial port to control several digital input/output lines[2]. To avoid duplicating his efforts, I thought I would concentrate on analog measurement in this article.

Many kinds of electronic sensors produce an output voltage proportional to some stimulus, such as pressure, temperature, or light intensity. Because of the convenience and wide variety of these sensors, voltage measurement has become the most common way for computers to collect laboratory data. The circuit presented in this article allows the Mac to measure voltages on four input channels, simultaneously, with an input range of ±2.5V, and a resolution of one part in 4096 (12-bits).

Support tools

A high-level text-based command set is supported by the circuit, making Mac programming easy. All functions can be tested with a terminal emulator program. A variety of Mac programming tools support the serial port, including FutureBASIC, HyperCard, and SuperCard (using a public-domain XFCN resource), LabVIEW, and the scripting languages of various software packages. Support can be extended to compiled languages through calls to the Mac Toolbox.

When high sampling rates are not needed, a serial-interfaced data acquisition device offers many advantages over bus-interfaced hardware, including low cost, portability, and platform independence. Entry-level Mac systems or PowerBooks, though they have no expansion slots, can support serial devices. The Mac can be placed in a safe and convenient location at the end of a long serial cable, minimizing the distance over which sensitive analog signals must travel.

To prevent this from turning into an electronics article, I eliminated all bells and whistles from the circuit. Conspicuously absent are any sort of grounding, isolation, or protection components. Therefore, this circuit should not be installed in systems where voltages higher than 30V are present. High voltages could be coupled into the circuit, the cables, or even your Macintosh, leading to possible shock or hardware damage. You must ensure that the system you construct will be safe, even in the event of a hardware or software failure.

Design methodology

Serial-interfaced Mac peripherals are easy to design when a microcontroller (MCU) chip is used because complex logic and timing operations are implemented with software. A general block diagram of a serial-interfaced device is shown in Figure 1. A "driver" is usually required to transmit and receive signals in a way that conforms to the RS-422 standard used by the Mac serial ports. Cabling distances of over a thousand feet are possible with typical RS-422 drivers. Such a driver chip was omitted from the PIC programmer circuit in the September 1996 *SciTech Journal* article because I expected very short cables to be used.

On the other side of the circuit are interface components that reach out into the real world. These components are seldom directly compatible with the Mac serial port, but the MCU chip can be programmed to perform whatever conversions are necessary. In fact, writing appropriate firmware for the MCU is usually the critical step in designing a peripheral. My experience has been that typical firmware programs for Mac interface projects are a few hundred lines long in assembly language. Most of the code deals with the Mac interface and can be carried from one project to the next.

My design method could be referred to as object-oriented hardware. I treat each specialized interface chip, along with its support subroutines, as an "object." When I want to start using a new kind of chip, I build a "demo" circuit, consisting of the chip and an MCU interfaced to the Mac serial port. This provides a clean environment for testing and debugging the support routines. When the time comes to build a new interface device, I piece together several hardware/software objects and add the last few lines of code. A complication with this technique is that I am still using assembly language, so a fair amount of discipline is required to avoid writing buggy or unmaintainable code.

Data acquisition circuit

A schematic diagram of the data acquisition circuit is shown in Figure 2. A Linear Technologies LTC1293 chip, U2, com- ▶

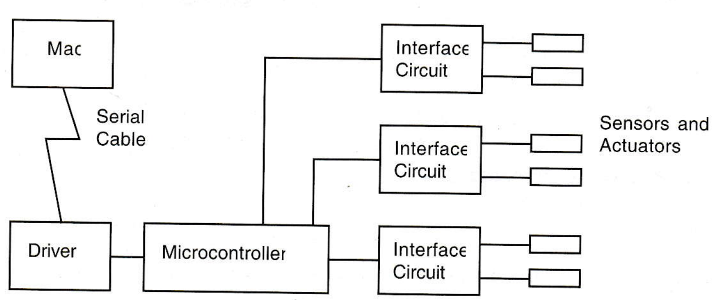


Figure 1. A typical data acquisition circuit uses an MCU forms the bridge between the Mac serial port and specialized interface components.

Francis Deck earned his PhD in Physics at the University of Notre Dame in 1993. By day, he leads product and process development projects for a small optics company in Fort Worth, Texas. He is also actively interested in electronics, data acquisition, process control, and technical management. You can reach Francis via e-mail at <fdeck@aol.com>.

bines the functions of input selector, sample-and-hold, and analog-to-digital converter (ADC) in a single chip. It provides six analog inputs, and can accept negative as well as positive input voltages. Only four of the inputs are actually used because operational amplifiers come in convenient "quad" packages. A preamp or buffer is required to isolate the signal inputs from the ADC.

Each section of U1, a National Semiconductor LMC660 (or Texas Instruments TLC2264) quad CMOS op amp, is wired as a "buffer" with a gain of 1. This circuit exploits specific features of CMOS op amps and may behave erratically if a general-purpose op amp chip is used.

A 100KΩ resistor wired in series with each input protects the op amps from signal overloads. An additional resistor is wired in parallel with each input to avoid the confusion of having an apparent signal on a disconnected input. The ADC can be destroyed by input voltages that exceed its nominal -4 to +5V power supply range. Since the preamp is powered by the same supplies, its output cannot overload the ADC.

The full-scale range of the ADC is determined by a single reference voltage, Vref, generated here by a crude resistive voltage divider. A diode, D1, drains filter capacitor C3 when power is turned off, to avoid damaging the ADC chip. A proper voltage reference chip could be used in place of the divider resistors R9 and R10, but D1 and C3 must remain in place.

A Microchip PIC16C84 microcontroller, U3, is the brains of the circuit. I chose to use a member of the PIC family for compatibility with my Mac-based development tools. The 3.6864MHz clock frequency is a convenient multiple of the 57.6kHz serial port baud rate. The MCU talks to the ADC via a synchronous serial protocol. The chip select line (/CS) initiates each instruction transmitted to the ADC. Successive data bits are placed on data lines (Din and Dout), synchronized by pulses on a clock line (Clk). The Din and Dout lines can be shared because they are never active at the same time. For each data conversion, the MCU transmits an 8-bit command to the ADC, indicating the choice of input channel and a few other particulars, and then receives a 12-bit conversion result word.

The MCU interfaces to the Mac via U4, a National Semiconductor DS8921 RS-422 transceiver. As with the MCU used in Mike Cook's new serial input/output board, the PIC16C84 MCU has no internal UART, so the serial port lines are controlled directly by timing loops in firmware. The pin names and numbers for the serial port refer to a female Mini-DIN8 connector on the data acquisition circuit. A standard "Mac printer" cable, which interchanges the TxD and RxD signals, should be used.

The circuit requires power supply voltages of roughly ±5V, but there is considerable leeway. A standard 9V battery is connected to U5, a standard 79L05 negative voltage regulator IC. By using a negative regulator, the negative terminal of the battery is conveniently available as a -4V supply. The circuit can also be connected to existing ±5V supplies.

Construction of the circuit is straightforward, and nearly any technique will give acceptable results. The performance of the ADC depends primarily on having low-impedance ground connections. I used a cheap solderless breadboard for my prototype, which was good enough to demonstrate the basic operation of the circuit, and to debug my firmware. For my final version, I laid self-adhesive copper tape directly on the breadboard and soldered the grounding wires and 10uF decoupling capacitors in place.

Software

The latest version of the PIC development system, described in the September 1996 *SciTech Journal* article, contains the firmware for the data acquisition system as an example project. (The Hypercard stack is downloadable from the MacSciTech FTP site, <ftp://ftp.ari.net/pub/MacSciTech> in the electronics engineering folder). The routines for driving the serial port are similar to those used in the PIC programmer. I only had to write one additional, somewhat difficult, routine for this project—to control the ADC chip.

I designed a simple user interface for the circuit which accepts a single-character command, reads all four ADC inputs, and returns the results as four decimal integers, separated by commas, and terminated by a carriage return. This data format is not the most efficient, but it makes Mac programming easy.

Before writing support software on the Mac, I checked the operation of the circuit, as well as the serial connections, using a general-purpose terminal emulator set to 57.6 kBaud, 8 data bits, no parity, and one stop bit. Then I wrote a data acquisition program using HyperCard and a public-domain XFCN code resource. I have always been pleasantly surprised by how quickly simple data acquisition problems can be solved using HyperCard.

Conclusion

A simple data acquisition project demonstrates the use of single-chip microcontrollers as the basis for custom Macintosh interface devices. The circuit can be extended in several ways: Eight digital input/output lines on the MCU chip are available for general use, as are two analog inputs on the ADC. Software functions such as signal averaging can be added.

As Mac-based hardware and software products gain in sophistication, the importance of low-end data acquisition tools should not be forgotten. The cost to purchase an advanced data acquisition package, possibly including replacement of an otherwise serviceable 68020 or 68030 Mac, is prohibitive for most hobbyists and home users. The circuit presented in this article, along with a copy of the HyperCard Development System, adds data acquisition capabilities to virtually any Mac for under $200.

Lowering the cost of data acquisition broadens the range of potential applications to include seemingly trivial, but nonetheless very useful tasks, such as long-term data logging. This is a great way to "recycle" an older Mac system, since HyperCard requires very little memory or disk space. If the hardware is cheap enough to permit several computers to be outfitted for data acquisition, then the logistical problems involved in sharing a single, expensive system are avoided. Computers can be dedicated to long-term projects, avoiding disruption and the possibility of errors caused by constant reconfiguration.

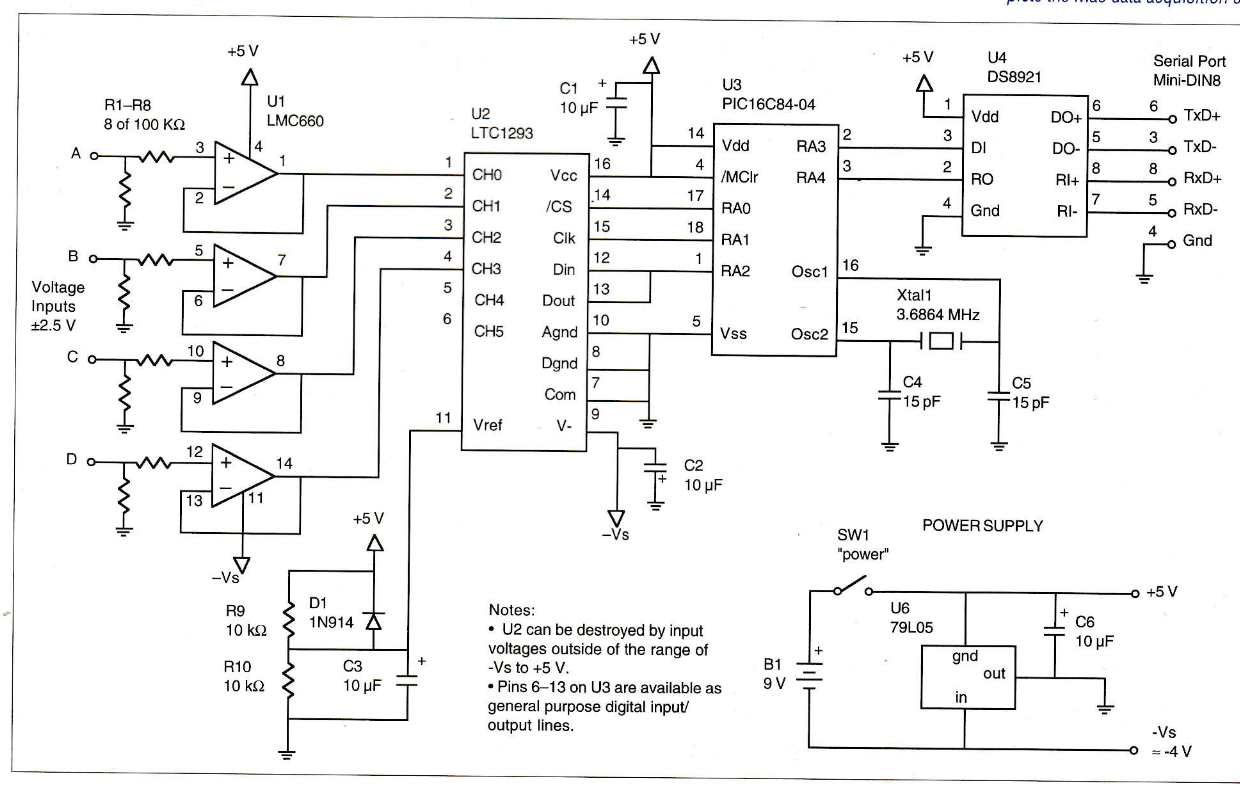
I am maintaining a small but growing Web site devoted to Mac hardware projects at <http://members.aol.com/fdeck/main.html>. Full circuit plans and software can be downloaded from this site, as can the PIC Development System. A profusion of inexpensive hardware gadgets are available for PC-compatibles, but finding comparable Mac-based products remains a challenge. Sometimes, hardware is compatible with the Mac, but no effort is made to provide appropriate documentation or software support. I plan to expand the site with information on commercial products that can be adapted to the Mac. **MST**

Francis' *MCU Development Kit* is available for purchase through MacSciTech. See page 10.

References

1. F. J. Deck, "A PIC microcontroller development system for the Mac," *SciTech Journal*, Vol. 6, No. 9, p. 23 (September/October 1996)
2. M. Cook, "A new low-cost serial board—S I/O III," *SciTech Journal*, Vol. 6, No. 5, p. 27 (July/August 1996)

Figure 2. Only a handful of chips are needed to complete the Mac data acquisition circuit.



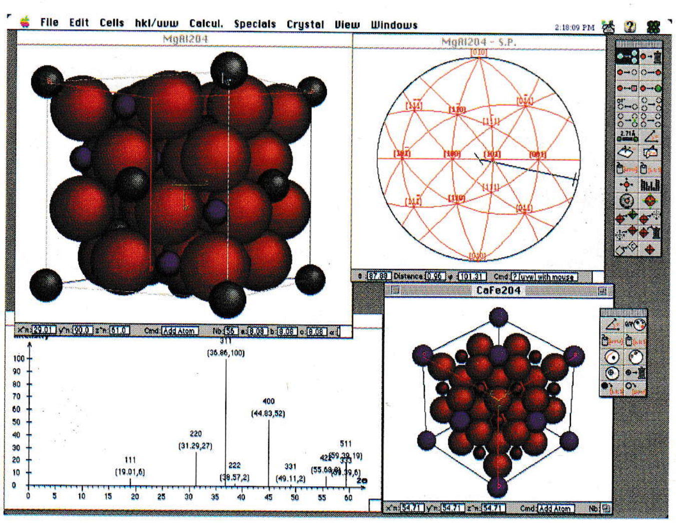
SciTech Journal™

January 1997
Volume 7, Number 1 $5.95 US, $6.95 Canada

Official Journal of the MacSciTech Association
http://www.macscitech.org/

FEATURES

DEPARTMENTS



Ca.R.Ine Crystallography 3.0
Crystallographic software
ESM Software
see page 38

49 Midgley Lane
Worcester, MA 01604-3564 USA

***************************MIXED ADC
Francis Deck
Physicist
3500 Willowbrook Drive
Ft Worth, TX 76133
S15 P4

Bulk Rate
U.S. Postage
PAID
Fulton, MO
Permit No.38