An Easy-to-Use A/D Converter

Robert Daggit
1648 Hillview Rd
New Brighton MN 55112

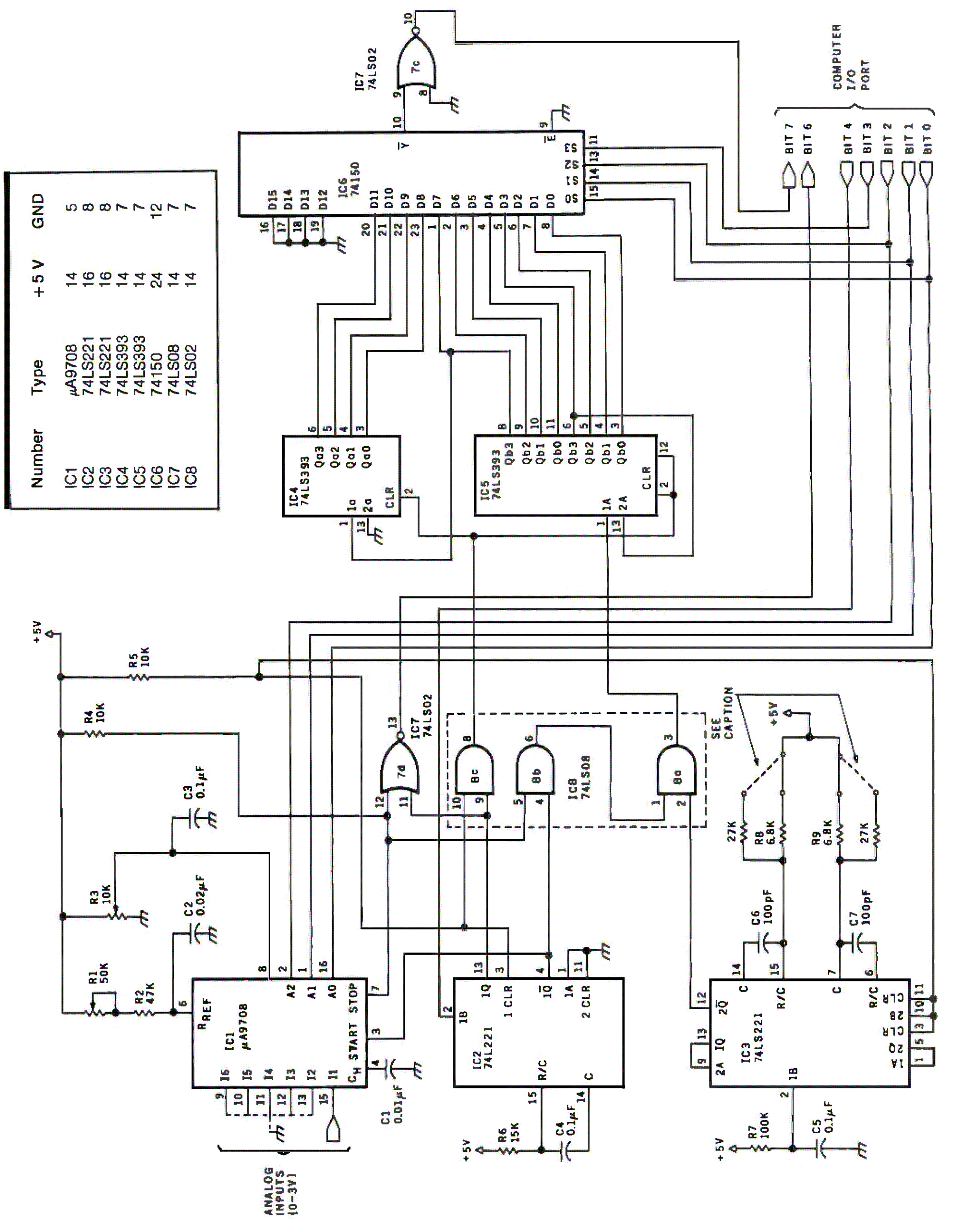
With the addition of an analog-to-digital converter and some simple sensors, a microcomputer can monitor analog voltages, read light levels, sense temperatures, or read the analog output from laboratory instruments. The six-channel A/D (analog-to-digital) converter that I will describe reads positive voltages from 0 to 3 V, with either 8 or 10 bits of accuracy. It interfaces to the computer through an 8-bit bidirectional peripheral port whose I/O (input/output) lines are individually programmable and latched when used as outputs.

About the Author
Robert Daggit is a Senior Research Technician at the Systems and Research Center of Honeywell Inc in Minneapolis. He is interested in the application of microprocessors to small, dedicated systems for laboratory use.

Once started, the converter operates asynchronously with respect to the computer and requires a minimum of code in the user's program. Conversion times are voltage-dependent, with an approximate range of 1 to 2 ms (milliseconds). A sample program segment and subroutine written in 6502 assembly language are included to illustrate the use of the converter.

Major components of the A/D converter unit, shown as a schematic diagram in figure 1, are a Fairchild Semiconductor μA9708 analog-to-digital-converter integrated circuit, a clock, a 12-bit counter, and a 16-bit output multiplexer. The μA9708 features an analog input multiplexer, controlled by address lines A0 thru A2, that selects one of eight input sources. Address 0 selects the internal zero voltage, and address 7 selects the internal reference voltage. Addresses 1 thru 6 select user inputs I1 thru I6, as shown in figure 1. Although the manufacturer rates the μA9708 at 8 bits of accuracy, it performs well at 10 bits of accuracy. A series of voltage readings taken at 0.1 V intervals from 0 to 3 V compared favorably with readings taken with a Fluke Model 8000A Digital Multimeter. Voltage differences ranged from 2 to 11 mV (millivolts). The greatest relative error, defined as the absolute value of the voltage difference divided by the multimeter reading, was less than 2%.

In order to read one of the analog channels, the channel address is placed on the address lines, and the ramp-start input (pin 3) is set low. The ramp-stop output (pin 7) goes high at this time. With the address lines stable for a signal-acquisition time of about 1 ms, the ramp capacitor, C1, charges to the voltage



Number	Type	+5 V	GND
IC1	µA9708	14	5
IC2	74LS221	16	8
IC3	74LS221	16	8
IC4	74LS393	14	7
IC5	74LS393	14	7
IC6	74150	24	12
IC7	74LS08	14	7
IC8	74LS02	14	7

Figure 1: *Schematic diagram of the A/D converter. Inputs I1 thru I6 of IC1 are the user's analog-input channels. The input voltage is converted to a binary number in the counter (IC4 and IC5), where it is retained until needed. The binary output is read in bit-serial fashion by the output multiplexer, IC6. Interface to the computer is through an 8-bit I/O port.*

Easy selection of 8 or 10 bits of accuracy is accomplished by installing the clock timing components (C6, C7, R8, and R9) on a DIP header (see figure 2).

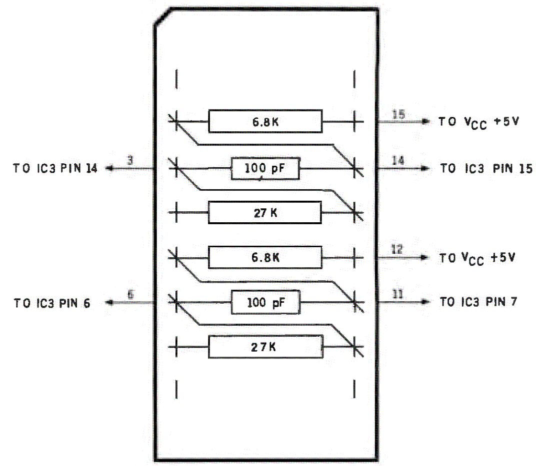


Figure 2: *Wiring of the DIP header (top view). This optional feature may be installed for easy selection of 8 or 10 bits of accuracy. The clock timing components are mounted on the header in such a way that when it is reversed in its socket, the time constants of IC3 (a 74LS221 monostable multivibrator) are appropriately changed.*

Listing 1: *A program segment, written for the 6502 microprocessor, that illustrates use of the A/D converter. Hexadecimal 10 is added to the channel address, and this value is then written to the interfacing I/O port to start the conversion. Data from the counter is read when needed.*

Address	Object Code	Label	Mnemonics	Comments
0250	A9 10		LDA H#10	;CHANNEL 0 ADDRESS
0252	8D 01 A8		STA DRA	;INITIATE A/D CONVERSION
0255	20 30 03		JSR RDADC	;READ CHANNEL 0 COUNT
0258	85 D0		STA D0	
025A	86 D1		STX D1	
025C	A9 17		LDA H#17	;CHANNEL 7 ADDRESS
025E	8D 01 A8		STA DRA	;INITIATE A/D CONVERSION
0261	20 30 03		JSR RDADC	;READ CHANNEL 7 COUNT
0264	85 C0		STA C0	
0266	86 C1		STX C1	
0268	A9 11		LDA H#11	;CHANNEL 1 ADDRESS
026A	8D 01 A8		STA DRA	;INITIATE A/D CONVERSION
026D	A9 02		LDA H#02	
026F	20 7C 05		JSR SUBM	;COUNT(REF) − COUNT(0)
0272	A5 C0		LDA C0	
0274	A6 C1		LDX C1	
0276	85 A0		STA A0	;SAVE CORRECTED REF COUNT
0278	86 A1		STX A1	
027A	20 30 03		JSR RDADC	;READ CHANNEL 1 COUNT
027D	85 C0		STA C0	
027F	86 C1		STX C1	
0281	A9 02		LDA H#02	
0283	20 89 05		JSR CMPM	;IS COUNT(1) < COUNT(0)?
0286	10 08		BPL SKIP	
0288	A5 D0		LDA D0	;SET COUNT(1)
028A	85 C0		STA C0	; TO
028C	A5 D1		LDA D1	; COUNT(0).
028E	85 C1		STA C1	;
0290	A9 02	SKIP:	LDA H#02	
0292	20 7C 05		JSR SUBM	;COUNT(1) − COUNT(0)

at the selected input. The ramp-start input is then set high. This disconnects the input voltage from the ramp capacitor, which now discharges linearly at a controlled rate through resistors R1 and R2. When the ramp capacitor is discharged, the ramp-stop output goes low. Since the capacitor's discharge time is directly proportional to the input voltage, a counter running during the interval from the conditions ramp-start-high to ramp-stop-low will, at the end, contain a count that is proportional to input voltage.

In this circuit, a low-to-high transition of peripheral-port bit 4 triggers IC2, a 74LS221 monostable multivibrator. Its Q output goes high to clear the counter, while the $\overline{Q}$ output holds the ramp-start line low, allowing the μA9708 (IC1) to acquire the voltage from the selected channel. Upon timing out, IC2's outputs change states, raising the ramp-start line to a high logic level and turning on the counter. When the ramp-stop line goes low, the counting stops, and peripheral-port bit 6 goes high to signal the computer that the conversion is complete. The counter value is the useful output of the converter, and is retained until it has been read and the next conversion cycle has begun.

The clock, IC3, is a multivibrator whose frequency is set to about 1 MHz by the 100 pF capacitors, C6 and C7, and 6.8 k-ohm resistors, R8 and R9, for a 10-bit count. An 8-bit count is selected by replacing R8 and R9 with 27 k-ohm resistors. If the frequency-determining components are installed symmetrically on a header, as shown in figure 2, the 8- or 10-bit counts can be selected by simply unplugging the header and reversing it.

A ripple counter and a 16-bit output multiplexer, controlled by address lines A0 thru A3, complete the circuit.

Before the circuit is used, all unused analog inputs should be grounded and the reference voltage and ramp slope should be set. The 10 k-ohm potentiometer, R3, is first adjusted until the reference voltage at pin 8 of IC1 is exactly 3 V, as indicated by an accurate voltmeter. Then the converter connected to the computer is run in a loop, repeatedly addressing and reading the reference voltage at address 7. The 50 k-ohm potentiometer, R1, is adjusted until the count is just under hexadecimal FF for an 8-bit count, or hexadecimal 3FF for a 10-bit count.

In normal use, the program must first configure the peripheral-port bits 0 thru 4 as outputs and bits 5 thru 7 as inputs, and it must clear bit 4. Voltage readings are taken by writing the value of the channel address plus hexadecimal 10 to the peripheral port and then waiting until bit 6 goes high. The channel address should not be changed during this time. Reading of the counter data automatically clears peripheral port bit 4, enabling its low-to-high transition when the next address is written to the port. The counter is read a bit at a time by writing the address of the desired bit into the peripheral port, reading the port, and then left-shifting bit 7 (the counter data bit) into a register pair

Listing 2: *RDADC, a 6502 subroutine to read data from the counter in the converter. The 16-bit counter value is returned in the accumulator and X register. Status bits reflect the condition of the high-order byte.*

```
;                  ***** READ A/D CONVERTER *****
;        THIS SUBROUTINE READS THE COUNTER OF THE A/D CONVERTER.
;        IT RETURNS THE HIGH-ORDER BYTE IN THE ACCUMULATOR
;        AND THE LOW-ORDER BYTE IN THE X REGISTER.
;
;        SCRATCH LOCATIONS USED: F0,F1
;
0330  A9 40       RDADC: LDA   H#40   ;LOAD MASK TO TEST BIT 6
0332  2C 01 A8     LP1:  BIT   DRA    ;IS A/D CONVERSION COMPLETED?
0335  50 FB              BVC   LP1    ;IF NOT, LOOP UNTIL DONE
0337  A2 0F              LDX   H#0F   ;LOAD INDEX REGISTER/COUNTER
0339  8E 01 A8     LP2:  STX   DRA    ;BIT ADDRESS
033C  AD 01 A8           LDA   DRA    ;READ BIT
033F  2A                 ROL   A      ;ROTATE ACCUMULATOR
0340  26 F1              ROL   F1     ;ROTATE MEMORY LOCATION F1
0342  26 F0              ROL   F0     ;ROTATE MEMORY LOCATION F0
0344  CA                 DEX
0345  10 F2              BPL   LP2    ;BRANCH IF POSITIVE
0347  A6 F1              LDX   F1     ;LOAD LOW-ORDER BYTE
0349  A5 F0              LDA   F0     ;LOAD HIGH-ORDER BYTE
034B  60                 RTS
```

Reference Designation	Part
IC1	μA9708, A/D converter
IC2,IC3	74LS221, monostable multivibrator
IC4,IC5	74LS393, dual 4-bit binary counter
IC6	74150, 1 of 16 data selectors
IC7	74LS02, quad 2-input NOR gate
IC8	74LS08, quad 2-input AND gate
C1	0.01 μF, polyester
C2	0.02 μF, ceramic
C3,C4,C5	0.1 μF, ceramic
C6,C7	100 pF, ceramic
R1	50 k-ohm, 10-turn potentiometer
R2	47 k-ohm, ¼ W, 5% tolerance
R3	10 k-ohm, 10-turn potentiometer
R4,R5	10 k-ohm, ¼ W, 10%
R6	15 k-ohm, ¼ W, 5%
R7	100 k-ohm, ¼ W, 10%
R8,R9	6.8 k-ohm or 2.7 k-ohm, ⅛ W, 5%

Table 1: *Parts list for circuit of figure 1. Capacitor C1 should be a low-leakage type. No precision tolerances are required.*

or 2 bytes of memory that will contain the 16-bit count. The sequence is repeated for each bit, starting with the most-significant bit at hexadecimal address 0F and ending with the least-significant bit at address 00.

The most efficient operation will result when the analog-to-digital conversion is initiated at a point in the program that occurs a number of instructions before the voltage reading is required. The computer is then free to execute the intervening instructions before having to wait for completion of the conversion. The hand-assembled program segment, shown in listing 1, illustrates the use of the converter and the RDADC subroutine (see listing 2). Note the instructions inserted between the initiation of the conversion at hexadecimal address 026A and the reading of the output at address 027A.

A nonzero count is always obtained, even when reading 0 V. This count must be subtracted from the reference voltage and channel counts. Thus, the computation for a linearized and scaled voltage reading becomes:

$$V(i) = \frac{\text{Count(Channel } i) - \text{Count (0)}}{\text{Count (7)} - \text{Count (0)}} \times V_{REF}$$

where V_{REF} is the reference voltage.

Long-term drift effects are minimized by reading the zero and reference voltages each time a channel is sampled. When reading very small input voltages, the possibility exists that a channel count may be smaller than the zero count. The apparent instability resulting from this condition is avoided by simply setting the channel count equal to the zero count.

The uses for such a converter are many and diverse. For example, if you are an energy-conscious homeowner, you may wish to monitor temperatures throughout your home. Or, if you are an amateur horticulturist, you may wish to monitor light intensity and temperatures of air and soil to optimize growing conditions for plants or cuttings. Whatever the application, I hope that this converter, with its 8 bits of accuracy for table subscripts or 10 bits of accuracy for better resolution, will serve you well. ■