Pulse Width Modulation A/D Conversion Techniques with COP800 Family Microcontrollers

National Semiconductor
Application Note 607
Kevin Daugherty
August 1991

1.0 BASIC TECHNIQUE

This application note describes a technique for creating an analog to digital converter using a microcontroller with other low cost components. Many applications do not require the speed associated with a dedicated hardware A/D converter and it is worth evaluating a more cost effective approach.

With a high speed CMOS microcontroller an eight bit A/D can be implemented that converts in approximately 10 ms. This method is based on the fact that if a repetitive waveform is applied to an RC network, the capacitor will charge to the average voltage, provided that the RC time constant is much larger than the pulse widths. The basic equation for computing the analog to digital result is:

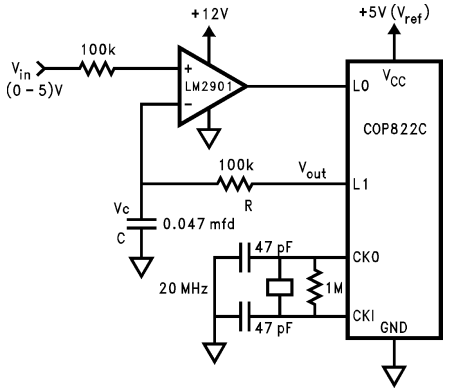
$$V_{in} = V_{ref}[T_{on}/(T_{on} + T_{off})] \quad (1)$$

With this equation it is necessary to precisely measure several time periods within both the T_{on} and T_{off} in order to achieve the desired resolution. Additionally, the waveform would have to be gradually adjusted to allow for the large RC time constant to settle out. This results in a relatively long conversion cycle. Modifying the equation and technique slightly, significantly speeds up the process. This technique works by averaging several pulses over a fixed period of time and is based on the following equation:

$$V_{in} = V_{ref}[\text{Sum of } T_{on}/(\text{Sum of } (T_{on} + T_{off}))] \quad (2)$$

2.0 IMPLEMENTATION

Figure 1 describes the basic circuit schematic that uses a National Semiconductor COP822C microcontroller, a low cost LM2901 comparator, two 100k resistors, and a 0.047 mfd film capacitor. The CMOS COP822C microcontroller provides a squarewave signal with logic levels very close to GND and V_{CC}. This generates a small ramp voltage on the capacitor for the LM2901 quad comparator input.



TL/DD/10407–1

FIGURE 1. Basic Circuit

To minimize error, a tradeoff must be made when selecting the resistor. The microcontroller output (L1) should have a large resistor to minimize the output switching offset (V_{os}), and the comparator should have a small resistor due to error caused by I_{bos} (input bias offset current).

Once the resistor is determined, the capacitor should be chosen so that the RC time constant is large enough to provide a small incremental voltage ramp. This design has a sample time of 20 μs and has a 4.7 ms time constant with a 0.047 mfd film type capacitor which has low leakage current to prevent errors. Since a 100k resistor is used in the RC network for one comparator input, another 100k resistor is required for the V_{in} input to balance the offset voltage caused by the comparator I_b (input bias current).

Figure 2 illustrates the relationship between the microcontroller squarewave output and the capacitor charge and discharge. Every 20 μs the comparator is sampled. If the capacitor voltage (V_c) is below V_{in} the RC network will receive a positive pulse. The inverse is true if V_c is above V_{in} at sample time. Note that with this approach, the PWM waveform is broken up into several small pulses over a fixed period instead of having a single pulse represent the duty cycle; thus a relatively small RC time constant can be used.

Mathematical Analysis:

let n = total number of T_{on} pulses and
m = total number of T_{off} pulses

then $V_c(t) = V_c + n[(V_{out} - V_c)(1 - e - t/RC)] - m[(V_c - V_o)(1 - e - t/RC)]$

let $V_c = V_{in}$ at start of conversion and
$K = (1 - e - t/RC)$

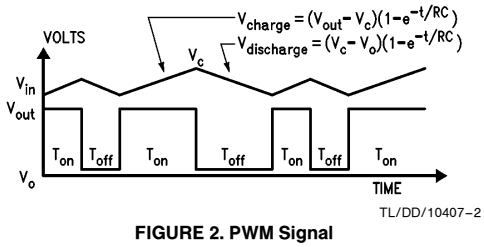
then $V_{in} = V_{in} + K_nV_{out} - K_nV_{in} - K_mV_{in} + K_mV_O$
$0 = K_nV_{out} + K_mV_o - KV_{in}(n + m)$

let $V_{out} = V_{ref} - V_{os}$

solving for V_{in}:

$$V_{in} = nV_{ref}/(n + m) - (nV_{os} - mV_o)(1/(n + m)) \quad (3)$$

Note that the RC value drops out of the equation and therefore is not an error factor.



TL/DD/10407–2

FIGURE 2. PWM Signal

TRI-STATE® is a registered trademark of National Semiconductor Corporation.

  RRD-B30M75/Printed in U. S. A.

3.0 SOFTWARE DESCRIPTION

Single Channel

Referring to the flow chart in *Figure 3*, and the code listed in *Figure 4*, the software counters T_{on} and TOTAL are first preloaded with the FF. The accumulator and register 0F1 are then loaded with 2 to provide for an initialization and final conversion cycle. Next, the L port is configured to complete the initialization of the microcontroller.

The comparator output is checked with the IFBIT 0,0D2 instruction. This will determine whether the RC network will receive a positive (V_{ref}) or ground pulse. You can think of the microcontroller as part of the feedback path of the comparator. The microcontroller uses the comparator output to decide what level output on L1 is required to keep the capacitor equal to the unknown input voltage. Each time the negative or GND pulse is applied, the T_{on} counter is decremented by DRSZ. Similarly, each time a sample loop is completed the TOTAL counter is decremented by DRSZ. Note that NOP instructions are used in the high and low loops. These are necessary to provide exactly the same cycles for a high or low L1 output pulse.

Once the TOTAL register is decremented to zero, the initialization loop is completed. Immediately afterwards, the L1 output is put in TRI-STATE® mode to minimize capacitor voltage variations while other instructions are completed. After the first conversion, the IFEQ A,0F1 instruction will be true and the T_{on} and TOTAL registers will be reloaded with FF. Following this, the L1 pin is restored as a high output and the 0F1 multiplier is decremented.

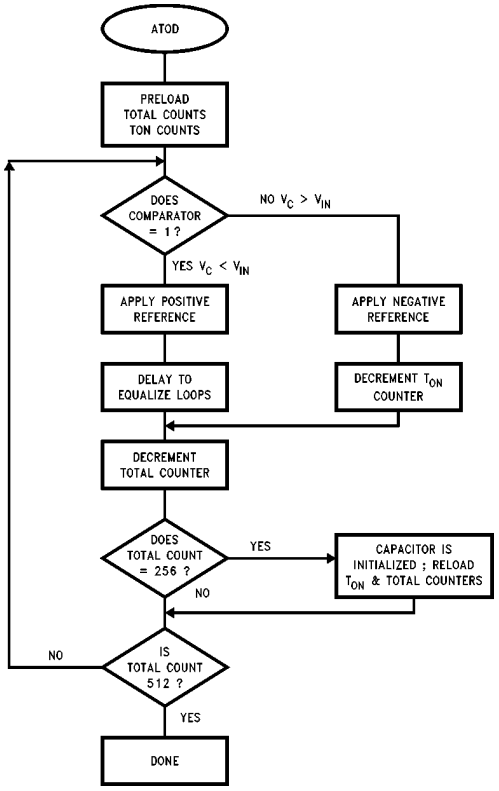
At this point the capacitor is equal to V_{in} and the actual conversion is started. When the TOTAL register is decremented to zero (255 samples later), the conversion is complete. T_{on} will not be reloaded since 0F1 was decremented and IFEQ A,0F1 will no longer be true. The accumulator is then loaded with T_{on} and stored in RAM location 00 with X A,00.

The final two instructions (RBIT 1,LCONF & RBIT 1[B]) are optional depending on the application and the amount of additional code required. This will prevent the capacitor from decaying appreciably between conversions and allow for a much quicker capacitor initialization time. Otherwise more time may be required, or a diode speed-up circuit as shown in *Figure 7d* is required to fully charge the capacitor prior to starting the actual conversion.

Eight Channel

This is bascially the same as that for the single channel. Referring to the flow chart in *Figure 5* and the code in *Figure 6*, the differences are in the front and back ends. Before the conversions are started, the X register is initialized to 00 for RAM location 00. The accumulator is then loaded with the current RAM pointer (LD A,X), OR'ed with the LDATA (OR A,LDATA), and finally the LDATA register is modified to provide for the proper output select (X A,LDTA).

Following the actual conversion cycle, the result is stored at the current RAM pointer (X A,[X+]) which also auto-increments the X register. The next conversion will use this to select the next channel and determine where to store the result. Once the eighth channel is converted, the IFEQ A,X instruction will be true and the RAM pointer will be reset (LD X,#00) before the next conversion is started.

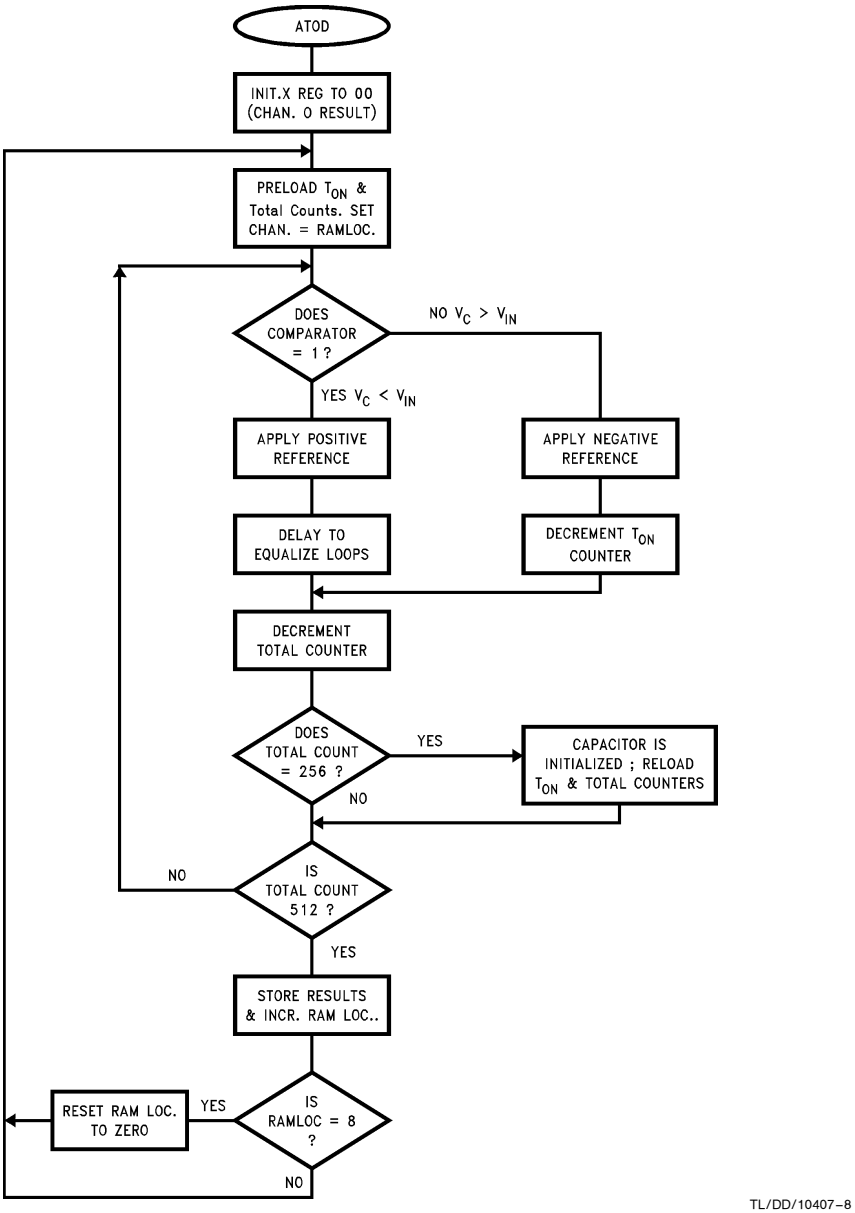


TL/DD/10407–7

FIGURE 3. PWM A/D Flow Chart

```
;The program listed below will work in any COP800 microcontroller
;(i.e. COP820, COP840, COP880, COP888). SET UP FOR .047 mfd CAP.,
;100K RES, @1 MICRO. CYCLE TIME. THE FIRST CONVERSION
;INITIALIZES, AND 2nd IS THE RESULT STORED IN RAM LOCATION 00.
.CHIP 820
LCONF=0D1
LDATA=0D0
TON=0F2
TOTAL=0F0
;
        LD A,#02          ;USED TO DETERMINE WHEN TO RELOAD
        LD TOTAL,#0FF     ;PRELOAD TOTAL COUNTS
        LD 0F1,#2         ;MULTIPLIER (255 TO INIT. PLUS 255 FOR RESULT)
        LD TON,#0FF       ;PRELOAD Ton
        LD 0FE,#0D0       ;LOAD B REG TO POINT TO LDATA REG.
        LD LDATA,#01      ;L PORT DATA REG, L0=WEAK PULL UP, L1=HIGH
        LD LCONF,#02      ;L PORT CONFIG REG, L0=INPUT, L1=OUTPUT
LOOP:   IFBIT 0,0D2       ;TEST COMPARATOR OUTPUT
        JP HIGH           ;JUMP IF L0=1
        NOP
        NOP               ;EQUALIZE TIME FOR SETTING AND RESETTING
        RBIT 1,[B]        ;DRIVE L1 LOW
        DRSZ Ton          ;DECREMENT Ton WHEN DRIVING LOW
        JMP COUNT
HIGH:   SBIT 1,[B]        ;DRIVE L1 HIGH
        NOP
        NOP
        NOP
        NOP
        NOP
        NOP               ;EQUALIZE HIGH AND LOW LOOPS
COUNT:  DRSZ TOTAL        ;DECREMENT TOTAL COUNTS
        JP LOOP
        RBIT 1,LCONF      ;TRISTATE L1 TO MINIMIZE ERRORS FROM EXTRA
        RBIT 1,[B]        ;CYCLES
        IFEQ A,0F1        ;CHECK INITIALIZATION LOOP COMPLETE
        JP RELOAD         ;JUMP IF TRUE.
        JP DEC            ;JUMP IF NOT END OF 2nd LOOP
RELOAD: LD 0F2,#0FF       ;RELOAD Ton WITH FF
        LD 0F0,#0FF       ;SYNC TOTAL AND Ton COUNTERS
DEC:    SBIT 1,[B]        ;SET L1 HIGH
        SBIT 1,LCONF      ;RESTORE L1 AS OUTPUT.
        DRSZ 0F1          ;DECREMENT MULTIPLIER UNTIL ZERO
        JMP LOOP          ;CONTINUE A/D UNTIL AFTER 2nd CONVERSION
        LD A,TON          ;LOAD A WITH Ton
        X, A,00           ;STORE RESULT IN RAM LOCATION 00
.end
```

FIGURE 4. Single Channel PWM A/D Listing



TL/DD/10407–8

FIGURE 5. 8 Channel PWM A/D Flow Chart

```
;L0,1,2 SELECTS CHANNEL OF CD4051 8:1 MUX, L3 IS THE COMP.
;OUTPUT, AND L4 DRIVES THE RC. RESULTS STORED IN RAM 00-07.
.CHIP 820
LDATA=0D0
LCONF=0D1
TON=0F2
TOTAL=0F0
        LD X,#00          ;INITIALIZE X REG FOR 1st RAM LOC.
CONVER: LD TOTAL,#0FF     ;PRELOAD TOTAL COUNTS
        LD 0F1,#02        ;TOTAL LOOP COUNTER
        LD TON,#0FF       ;PRELOAD Ton
        LD 0FE,#0D0       ;INIT. B REG TO POINT TO LDATA REG
        LD LDATA,#018     ;LDATA, L0-2=LOW, L3=PULLUP, L4=HIGH
        LD A,X            ;USED CURRENT RAM POINTER TO SELECT-
        OR A,LDATA        ;PROPER A/D CHANNEL.
        X A,LDATA         ;MODIFY LDATA FOR CHANNEL SELECTION.
        LD LCONF,#017     ;LCONF REG. L0-L2, L4=OUTPUT, L3=IN
LOOP:   IFBIT 3,0D2       ;TEST COMPARATOR OUTPUT AT L3 INPUT
        JMP HIGH          ;JUMP IF L3=HIGH
        NOP
        NOP               ;EQUALIZE TIME FOR SET AND RESET
        RBIT 4,[B]        ;DRIVE L4 LOW WHEN COMPARATOR IS LOW.
        DRSZ TON          ;DECREMENT Ton WHEN APPLYING NEG. REF.
        JMP COUNT         ;JUMP TO COUNT UNLESS Ton REACHES ZERO
HIGH:   SBIT 4,[B]        ;DRIVE L4 HIGH WHEN COMPARATOR IS HIGH
        NOP
        NOP
        NOP
        NOP
        NOP
        NOP               ;EQUALIZE HIGH AND LOW LOOP TIMES
COUNT:  DRSZ TOTAL        ;DEC. TOTAL COUNTS EACH LOOP
        JMP LOOP          ;JUMP UNLESS TOTAL CNTS.=0
        RBIT 4,LCONF      ;TRISTATE L4 TO MINIMIZE ERROR
        RBIT 4,[B]        ; "
        LD A,#02          ;USE TO DETERMINE WHEN TO RELOAD
        IFEQ A,0F1        ;CHECK FOR 2nd CONVERSION COMPLETE
        JP RELOAD         ;IF TRUE.
        JP DEC            ;OTHERWISE JUMP TO DEC
RELOAD: LD TON,#0FF       ;RELOAD Ton FOR START OF NEXT CONV.
        LD TOTAL,#0FF     ;SYNC Ton AND TOTAL COUNTERS
DEC:    SBIT 4,[B]        ;SET L4 HIGH
        SBIT 4,LCONF      ;RESTORE L4 AS OUTPUT.
        DRSZ 0F1          ;DECREMENT TOTAL LOOP UNTIL ZERO
        JMP LOOP          ;DONE WHEN 0F1 IS ZERO.
        LD A,TON          ;LOAD A WITH Ton RESULT
        X A,[X+]          ;STORE RESULT AT CURRENT RAM POINTER
                          ;AND AUTO INCREMENT POINTER
        LD A,#08          ;CHECK [X] RAM POINTER FOR
        IFEQ A,X          ;EIGHTH CHANNEL CONVERTER
        LD X,#00          ;RESET RAM POINTER IF [X]=8
        JMP CONVER
.END
```

FIGURE 6. 8-Channel PWM A/D Listing

4.0 ACCURACY AND CIRCUIT CONSIDERATIONS

The basic circuit will provide 8 bits ±1 LSB accuracy depending on the choice of comparator, and passive components. With this type of design several tradeoffs and error sources should be considered. First of all, conversion equation 2 assumes that the microcontroller output switches exactly to GND and V_{CC} (or V_{ref}). The COP822C will typically switch between 10 mV and 20 mV from GND and V_{CC} with a light load. This will cause an error equal to the offset voltage times the duty cycle (equ. 3). Fortunately, the offsets tend to cancel each other at mid range voltages. At near GND and V_{CC} input voltages the offsets are minimal due to the very small voltage drop across the resistor. If the error is undesirable, the offset voltage can be reduced by paralleling outputs with the same levels together, or by using a CMOS buffer such as a 74HC04 to drive the RC network (see *Figure 7* for suggested circuits).

Another possible source of error is with the LM2901 worst case input bias offset current of 200 nA over temperature. This will cause an error equal to R_{in} X I_{bos}, which equals 20 mV with a 100k resistor. Either the resistor or the I_{bos} can be reduced to improve the error. If the resistor is reduced then the L port offset voltages will increase so the preferred approach is to select a comparator with lower I_{bos} such as the LP339 which has an I_{bos} of only ±15 nA. The comparator V_{os} may also introduce error. The LM2901 V_{os} is ±9 mV, the LP339 V_{os} is only ±5 mV. An added benefit of using the LP339 is that since the I_{bos} is so small, the resistor for the RC network can be larger. In addition, one RC network could be used for several comparator input channels (refer to *Figure 7A*).

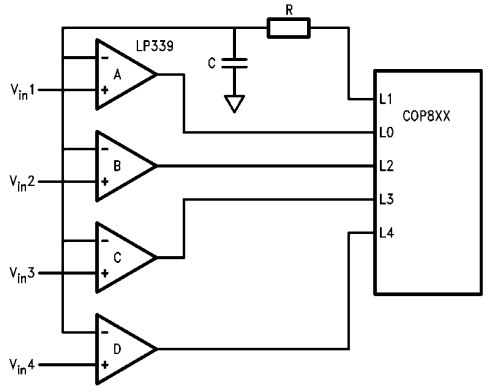
By using the LM604 *(Figure 7B)* the basic software can be easily extended for converting several channels. This will only require a control line to be selected before a conversion is started. Since the LM604 needs to be powered from a higher voltage than the input voltage range, the output voltage will also be higher than the microcontroller supply. This requires a current limiting resistor to be used in series between the LM604 output and the COP8XX. Note that two or more LM604's can be paralleled for providing several more A/D channels by utilizing the EN control input that can TRI-STATE the LM604 output when high.

When more than 4 channels of analog signals are required to be measured, the circuit in *Figure 7(d)* is recommended. This circuit utilizes an inexpensive CD4051 8:1 multiplexer with a single comparator (which could be on-board the micro). When measuring several input voltages that can vary, TRI-STATING the output driving the RC between conversions is not possible. It is necessary to provide 6x RC time constants to charge the capacitor to within 0.25%. Note that there are two 1N4148's across the comparator inputs. The diodes provide a quick capacitor charge path providing that the total input resistance is much smaller than the resistor used in the RC network (a 2k resistor will meet the requirements within 255 sample times). Once the capacitor is charged to within about 0.6V, the diodes will start turning off. At this point the microcontroller will start dominating the charge/discharge of the capacitor. After the initialization cycle is complete, the capacitor is very close to the unknown V_{in} and the diodes are effectively out of the circuit.

Depending on the speed and accuracy requirements, the total number of counts used in the conversion can be changed. Increasing the counts will give more accuracy with the practical limit of about 9–10 bits. With increased resolution, the capacitor ramp voltage per sample time should be decreased so that the capacitor can be initialized to within 1 LSB prior to conversion. This can be done by either increasing the RC time constant, or by using an initialization routine with a shorter sample time. The conversion time will depend on the total counts and the microcontroller oscillator frequency as described below:

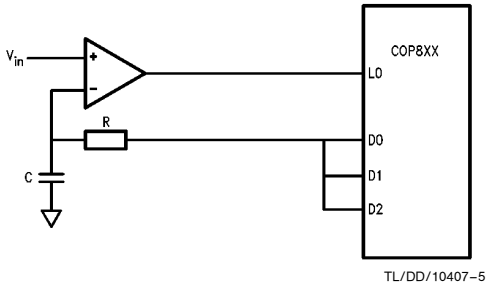
$$T_{con} = \text{Total counts} \times (20 \text{ cycles}) \times (\text{instruction cycle time})$$

Another factor to consider is when a non-ratiometric conversion is required, the reference voltage must have the tolerance to match the desired accuracy.



TL/DD/10407–4

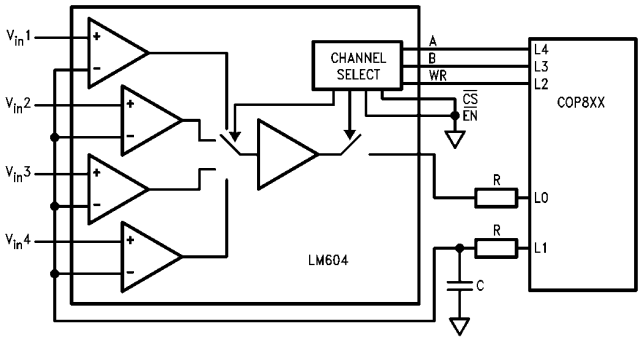
A. Multiple Channels with LP339 Low I_{bos} Comparator



TL/DD/10407–5

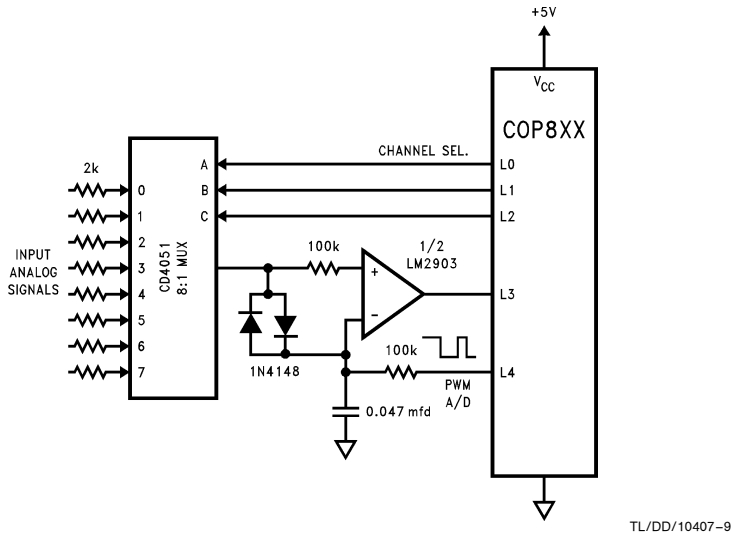
B. High Drive with Multiple Outputs

FIGURE 7. Suggested Circuits



TL/DD/10407–6

C. Four Channel A/D with LM604 MUX-Amplifier



TL/DD/10407–9

D. Eight Channel PWM A/D Circuit

FIGURE 7. Suggested Circuits (Continued)

5.0 CONCLUSION

The PWM A/D technique described in this application note provides a relatively fast discrete implementation with substantial cost savings compared to a dedicated hardware A/D. Minimal microcontroller I/O and software is required to interface with a comparator and RC network. Depending on the application requirements, the designer can tailor the basic 8-bit A/D a number of ways. By varying the total software counts, the desired speed and resolution can be adjusted. The number of A/D channels will determine the number of comparators used. In chosing the comparator, it is recommended that the designer refer to the data sheets and match the I_{bos} and V_{os} to the desired accuracy.

When other than a 1 μs instruction cycle is used, the RC time constant of 4.7 ms should be scaled to provide for a maximum peak-peak ramp voltage of <1 LSB of the desired accuracy. For example, if 8-bit accuracy is desired and the instruction cycle time is now 4 μs instead of 1 μs, multiply 4.7 ms by 4 to calculate the new RC.

Keep in mind that the comparator input voltage is limited so that you do not get erroneous/nonlinear results. Another possible problem is during development. When doing in-circuit emulation with the development equipment, note that there will be ground loops in the cable thus causing errors in your measurements. You can reduce this by connecting an extra GND and V_{CC} wire between your prototype and development system power and GND. It is still possible to see offsets in the sockets holding the COP8XX in the development board, however this should be relatively small. The best test is to take accurate measurements with an emulator in the actual prototype circuit.

Lit. # 100607

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

National Semiconductor Corporation
2900 Semiconductor Drive
P.O. Box 58090
Santa Clara, CA 95052-8090
Tel: 1(800) 272-9959
TWX: (910) 339-9240

National Semiconductor GmbH
Livry-Gargan-Str. 10
D-82256 Fürstenfeldbruck
Germany
Tel: (81-41) 35-0
Telex: 527649
Fax: (81-41) 35-1

National Semiconductor Japan Ltd.
Sumitomo Chemical
Engineering Center
Bldg. 7F
1-7-1, Nakase, Mihama-Ku
Chiba-City,
Ciba Prefecture 261
Tel: (043) 299-2300
Fax: (043) 299-2500

National Semiconductor Hong Kong Ltd.
13th Floor, Straight Block,
Ocean Centre, 5 Canton Rd.
Tsimshatsui, Kowloon
Hong Kong
Tel: (852) 2737-1600
Fax: (852) 2736-9960

National Semiconductores Do Brazil Ltda.
Rue Deputado Lacorda Franco
120-3A
Sao Paulo-SP
Brazil 05418-000
Tel: (55-11) 212-5066
Telex: 391-1131931 NSBR BR
Fax: (55-11) 212-1181

National Semiconductor (Australia) Pty, Ltd.
Building 16
Business Park Drive
Monash Business Park
Nottinghill, Melbourne
Victoria 3168 Australia
Tel: (3) 558-9999
Fax: (3) 558-9998