Low Cost A/D Conversion Using COP800

INTRODUCTION

Many microcontroller applications require a low cost analog to digital conversion. In most cases the controller applications do not need high accuracy and short conversion time.

This appnote describes a simple method for performing analog to digital conversion by reducing external elements and costs.

PRINCIPLE OF A/D CONVERSION

The principle of the single slope conversion technique is to measure the time it takes for the RC network to charge up to the threshold level on the port pin, by using Timer T1 in the input capture mode. The cycle count obtained in Timer T1 can be converted into voltage, either by direct calculation or by using a suitable approximation.

Figure 1 shows the block diagram for the simple A/D conversion which measures the temperature.

BASIC CIRCUIT IMPLEMENTATION

Usually most applications use a comparator to measure the time it takes for a RC network to charge up to the voltage level on the comparator input. To reduce cost, it is possible to switch both inputs as shown in *Figure 2*.

Port G3 is the Timer T1 input. Ports G2/G1 are general purpose I/O pins that can be configurated using the I/O configurations (push-pull output/tristate). All Port G pins are Schmitt Trigger inputs. R_{LIM} is required to reduce the discharge current.

GENERAL IMPLEMENTATION

The temperature is measured with a NTC which is linearized with a parallel resistor. Using a parallel resistor, a linearization in the range of 100 Kelvin can be reached. The value of the resistor can be calculated as follow:

$$R_P = R_{tm} * (B - 2T_m)/(B + 2T_m)$$

R_{tm} Value of the NTC at a medium temperature

T_m Medium Temperature

B   NTC−material constant
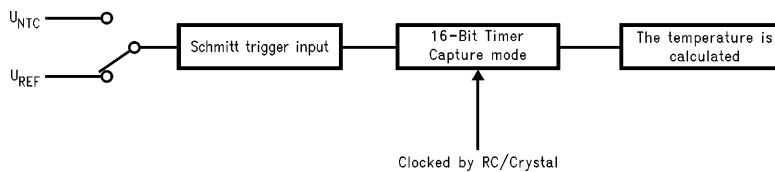
The linearization reduces the code, improves the accuracy and the tolerance of the NTC-R network (e.g. NTC = 100 kΩ $\pm$10%, R = 12 kΩ $\pm$1%, NTC//R $\pm$2%). Using that method the useful range does not cover the whole operating temperature range of the NTC.

GENERAL ACCURACY CONSIDERATIONS

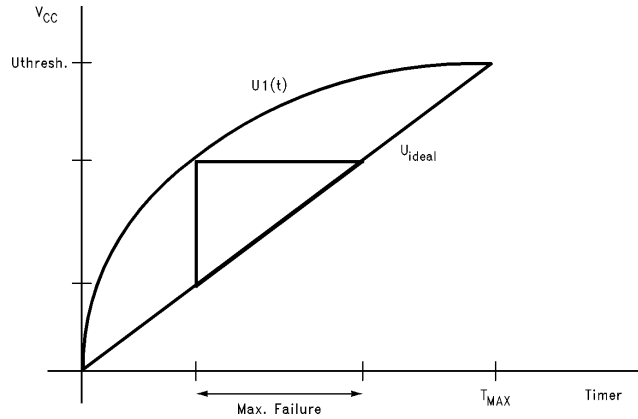Using a single slope A/D conversion the accuracy is dependent on the following parameters:

— Stability of the Clock frequency

— Time constant of the RC network

— Accuracy of the Schmitt Trigger level

— Non-linearity of the RC-network

Figure 3. The maximum failure that appears when a sawtooth is generated without using a current source. In the current application the maximum failure would be more than 15% without using methods for reducing the non-linearities of RC-network/NTC-network.
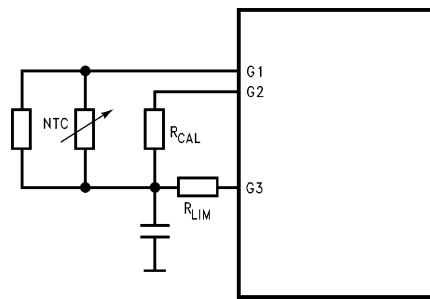
TL/DD/12075–2

FIGURE 2. Basic Circuit Implementation

TL/DD/12075–1

FIGURE 1. Simple A/D Conversion

FIGURE 3. Single Slope A/D Conversion

The maximum error occurs when the gradient of the exponential function (RC) equals the gradient of the straight line (counter).

To reduce the error that is caused by the non-linearity of the RC-network a offset should be added to the calculated value. The offset reduce the failure to the middle.

Further, the accuracy can be improved by using a relative measurement method. The following diagram shows the method.
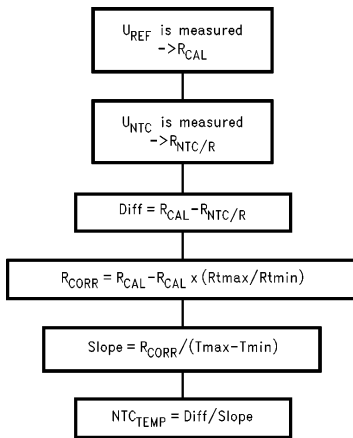
FIGURE 4. Accuracy Improvement

Measurement:

— Timer Capture mode: R_{CAL} * C is measured

— Timer Capture mode: $R_{NTC//R}$ * C is measured

Calculation:

— Build the vertical-component ($R_{TMIN} - R_{TMAX}$) of the triangle

— Calculate the slope

— Calculate the actual temperature

Using this method the accuracy is primarily dependent on the accuracy of R_{TMIN} and R_{TMAX} and independent of the stability of the system clock, the capacitor and the threshold of the Schmitt Trigger level. The variation of the capacitor only leads to variation of the resolution.

The following diagram shows the ideal resistance/temperature characteristic of a NTC which is linearized with a parallel resistor.
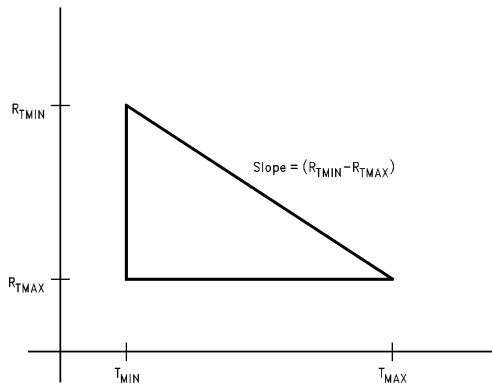
FIGURE 5. Resistance vs Temperature Characteristics

APPLICATION EXAMPLE

The following application example for temperature measurement demonstrates the procedure. The temperature is measured from 20° to 100° and is displayed on a Triplex LCD display.

NTC_{20} $= 100\ k\Omega\ \pm10\%$

R_P $= 12\ k\Omega\ \pm1\%$

T_m $= 333\ Kelvin \rightarrow 60\ Degrees$

B $= 4800\ Kelvin$

$NTC_{20}//R_P = 10.7\ k\Omega\ \pm2\%$

R_{CAL} $= 10.7\ k\Omega\ \pm1\%$

T_{MIN} $= 20\ Degree$

R_{TMIN} $= 10.7\ k\Omega$

T_{MAX} $= 100\ Degree$

R_{TMAX} $= 2.8\ k\Omega$

C $= 1\ \mu F$

RC-Clock $= 2\ MHz \rightarrow 200\ kHz$ instruction cycle, $5\ \mu s$

Timeconst. $= R_{CAL} * C \rightarrow 0.0107s$

Resolution $= 2140 \rightarrow 11$ byte, depends which Cap. value is used

Accuracy $= \pm2\ Degree$

This temperature measurment example shows a low cost technique ideally suited for cost sensitive applications which do not need high accuracy.

Figure 6 shows the complete circuit of the demoboard using the Triplex LCD method and the low cost A/D conversion technique.

The Triplex LCD drive technique is documented in a separate application note.
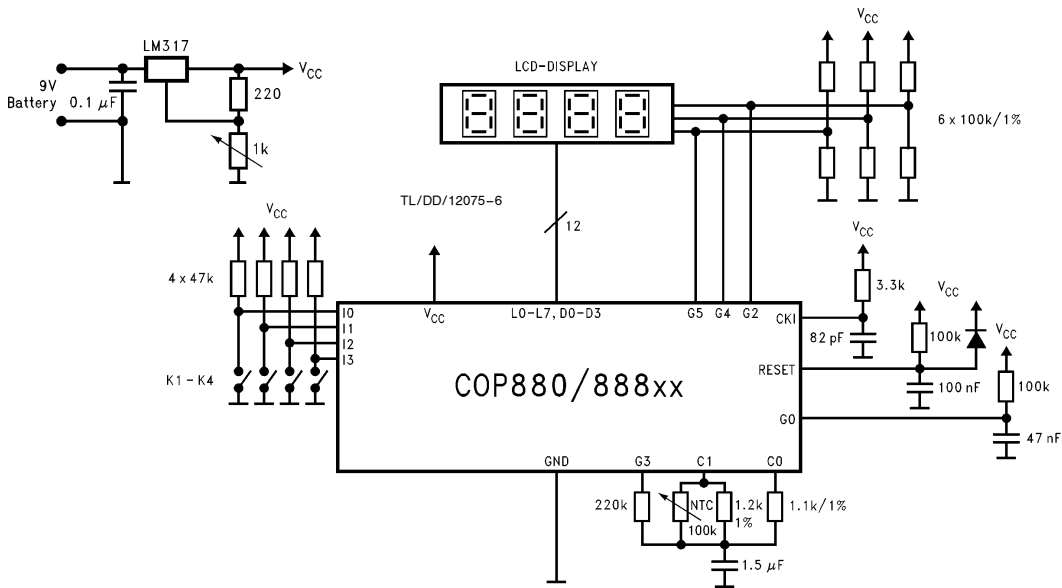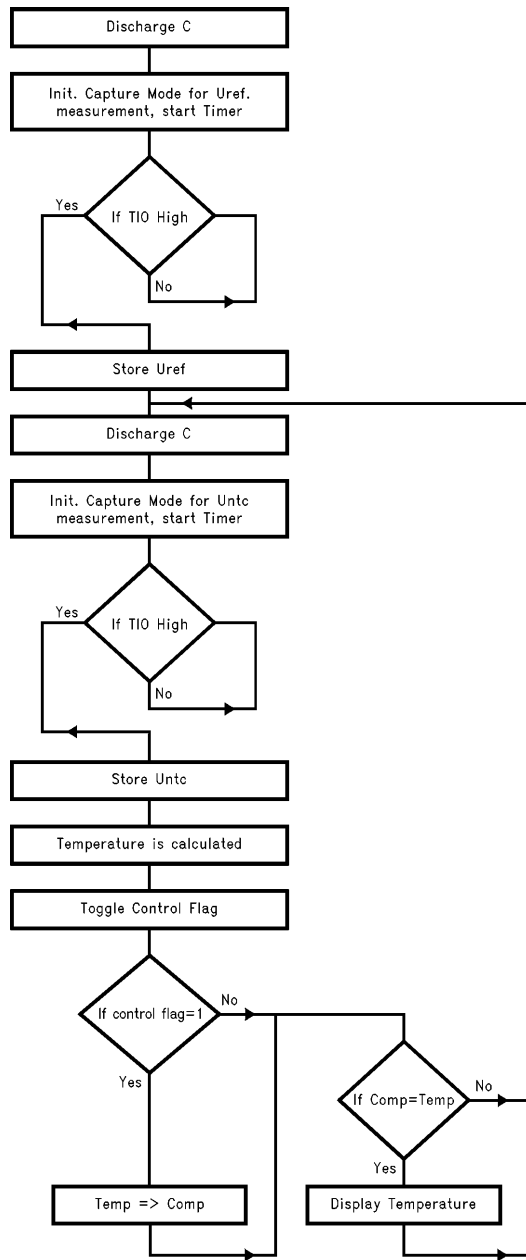

TL/DD/12075-6

TL/DD/12075-7

FIGURE 6. Circuit Diagram

Pressing key 1, key 2 the temperature is displayed in Degree/Fahrenheit.

Pressing key 3, key 4 Up/Down counter is displayed.

3

SOURCE CODE

Figure 7 shows the flow chart of the program.

TL/DD/12075-8

FIGURE 7. Flow Chart

The following code is required to implement the function. It does not include the code for the Triplex LCD drive.

```
RAM = 17 Byte;
ROM = 450 Byte; Optimization is possible about 50 byte if the B - pointer consistent is used!
;**************************A/D-CONVERSION*********************************
;
;
;
;*************************VAR. DECLARATION******************************
.SECT REGPAGE,REG
COUNT1:      .DSB 1
COUNT2:      .DSB 1
;
.SECT BASEPAGE,BASE
ZL:          .DSB 1          ;TEMPORARY
YL:          .DSB 1          ;TEMPORARY
;
.SECT RAMPAGE,RAM
CALIBLO:     .DSB 1          ;CALIBRATION-VALUE
CALIBHI:     .DSB 1
NTCLO:       .DSB 1          ;NTC-VALUE
NTCHI:       .DSB 1
TEMP:        .DSB 2          ;TEMP.-VALUE
KORRL:       .DSB 2
COMPL:       .DSB 1
COMPH:       .DSB 1
CONTROL:     .DSB 1          ;STATUS REGISTER
;********************START MAIN PROGRAM**************************
MAIN: LD     SP,#06F         ;INIT SPACKPOINTER
      JSR    DISCH           ;DISCHARGE C (A/D-CONVERSION)
      JSR    CALB            ;INIT CAPTURE MODE FOR UREF. MEASURMENT
POLL: IFBIT  3,PORTGP        ;POLL - MODE (TIO - PORT)
      JP     CAL
      JP     POLL
CAL:  LD     B,#CALIBLO
      JSR    CAPTH           ;STOP TIMER, STORE CAPTURE VALUE
      JSR    CALCR           ;SLOPE IS CALCULATED
NEW:  JSR    DISCH           ;DISCHARGE C (A/D-CONVERSION)
      JSR    NTC             ;INIT CAPTURE MODEFOR UNTC MEASURMENT
POLL1: IFBIT 3,PORTGP        ;POLL-MODE
      JP     CAL1
      JP     POLL1
CAL1: LD     B,#NTCLO
      JSR    CAPTH           ;STOP TIMER, STORE CAPTURE VALUE
      JSR    CALCN           ;TEMPERATURE IS CALCULATED
      JSR    DISCH           ;DISCHARGE C (A/D-CONVERSION)
      JSR    DCHECK          ;REDUCE THE DISPLAY FLICKERING
      JMP    NEW
.ENDSECT
;*****************************************************************************
;
```

TL/DD/12075–9

```
;*************************************************************
.SECT CODE1,ROM
;THIS ROUTINE IS REQUIRED TO REDUCE THE NOICE ON THE LINE AND THE
; DISPLAY FLICKERING.

.SECT CODE1,ROM
DCHECK:                   ;COMPARE TWO VALUES, IF EQUAL THEN
      LD   A,CONTROL      ;DISPLAY IT, OTHERWISE THE OLD VALUE
      XOR  A,#080         ;IS DISPLAYED
      X    A,CONTROL
      IFBIT 7,CONTROL
      JSR  SAVE           ;TEMP. SAVE
      JSR  COMP           ;COMPARE
      RET
;*************************************************************
; HANDLER FOR CAPTURE MODE
CAPTH: RBIT TPND,PSW      ;RESET TIMER PENDING
      RBIT TRUN,PSW       ;STOP TIMER
      LD   A,#0FF
      SC
      SUBC A,TAULO
      X    A,[B+]         ;STORE THE CAPTURED VALUE
      LD   A,#0FF
      SUBC A,TAUHI
      X    A,[B+]         ;STORE THE CAPTURED VALUE
      RET
;*************************************************************
; CALIBRATION SUBROUTINE, UREF IS MEASURED
CALB:
      RBIT 3,PORTGD
      RBIT 3,PORTGC       ;TRISTATE TIO
      LD PORTCD,#00
      LD PORTCC,#00       ;TRISTATE PORT C
      T1CAP HIGH          ;INIT CAPTURE MODE, HIGH SENSITIVE (MACRO)
      LD B,#CALIBLO
      SBIT 0,PORTCD       ;CONFIGURE C0 TO OUTPUT HIGH
      SBIT 0,PORTCC       ;CHARGE CAP.
      SBIT TRUN,CNTRL     ;START TIMER CAPTURE MODE
      RET
;*************************************************************
; NTC SUBROUTINE, UNTC IS MEASURED
NTC:
      RBIT 3,PORTGD
      RBIT 3,PORTGC       ;TRISTAT TIO
      LD PORTCD,#00
      LD PORTCC,#00       ;TRISTATE PORT C
      T1CAP HIGH          ;INIT CAPTURE MODE, HIGH SENSITIVE (MACRO)
      LD B,#NTCLO
      SBIT 1,PORTCD       ;CONFIGURE C1 TO OUTPUT HIGH
      SBIT 1,PORTCC       ;CHARGE CAP.
      SBIT TRUN,CNTRL     ;START TIMER CAPTURE MODE
      RET
;*************************************************************
```

TL/DD/12075–10

6

```
;***********************************************************
;DISCHARGE - ROUTINE
DISCH:
        LD PORTCD,#000
        LD PORTCC,#000
        RBIT TIO,PORTGD     ;DISCHARGE CAP.
        SBIT TIO,PORTGC
        LD COUNT1,#H(500)  ;DISCHARGE TIME
        LD COUNT2,#L(500)
        JSR C1              ;DELAY ROUTINE FOR DISCHARGE TIME
        RET
;***********************************************************
;THIS SUBROUTINE CALCULATES THE SLOPE
;THE FOLLOWING CALCULATIONS ARE DONE
;KORR=CALIB/11KOHM (RCALIB.=11KOHM)
;KORR=KORR*2,8KOHM (T=100 DEGREE, RNTC=2,8KOHM)
;CALIB=CALIB-KORR
;DIV=CALIB\80    (TEMPRANGE=80 DEGREE,100-20), SLOPE IS CALCULATED
CALCR:
;KORR=CALIB/11KOHM
        LD ZL,#L(110)
        LD ZL+1,#H(110)
        LD A,CALIBLO
        X  A,YL
        LD A,CALIBHI
        X  A,YL+1
        JSR DIVBIN16        ;SUBROUTINE BINARY DIVIDE 16 BIT BY 16 BIT
        LD A,YL
        X  A,KORRL
;***********************************************************
;KORR=KORR*28
        LD A,KORRL
        X  A,ZL
        LD A,#28
        X  A,YL
        JSR MULBIN8         ;SUBROUTINE MULTIPLY TWO 8 BIT VALUES
        LD A,YL
        X  A,KORRL
        LD A,YL+1
        X  A,KORRL+1
;***********************************************************
;KORR=CALIB-KORR
        LD B,#CALIBLO
        LD A,[B+]
        SC
        SUBC A,KORRL
        X A,KORRL
        LD A,[B]
```

7

```
        SUBC A,KORRL+1
        X A,KORRL+1
;*****************************************************************
;DIV=KORR/80
        LD ZL,#L(80)
        LD ZL+1,#H(80)
        LD A,KORRL
        X  A,YL
        LD A,KORRL+1
        X  A,YL+1
        JSR DIVBIN16          ;SUBROUTINE BINARY DIVIDE 16 BIT BY 16 BIT
        LD A,YL
        X  A,DIV
        RET
;*****************************************************************
;THIS SUBROUTINE CALCULATES THE TEMPERATURE
;THE FOLLOWING CALCULATIONS ARE DONE
;TEMP=CALIB-NTC
;TEMP=TEMP/DIV
;ADD OFFSET 20 DEGREE
;CONVERSION FROM HEX TO BCD
;*****************************************************************
;TEMP=CALIB-NTC
CALCN: LD B,#CALIBLO
        LD A,[B+]
        SC
        SUBC A,NTCLO
        X   A,TEMP
        LD A,[B]
        SUBC A,NTCHI
        IFNC
        JMP ERR
        X   A,TEMP+1
;*****************************************************************
;TEMP=TEMP/DIV
        LD A,TEMP
        X  A,YL
        LD A,TEMP+1
        X  A,YL+1
        LD A,DIV
        X  A,ZL
        CLRA
        X  A,ZL+1
        JSR DIVBIN16          ;SUBROUTINE BINARY DIVIDE 16 BIT BY 16 BIT
        LD A,YL
        ADD A,#20             ;ADD TEMPERATURE OFFSET
        IFGT A,#56            ;IF TEMPERATURE IS HIGER THAN 56 DEGREE THEN
        JSR CORR              ;ADD CORRECTION. OFFSET
;*****************************************************************
```

```
;****************************************************************
;HEX TO BCD CONVERSION
        X   A,ZL
        LD  A,ZL
        IFGT A,#100         ;IF TEMPERATURE IS MORE THAN 100 DEGREE THEN
        JP  ERR             ;ERROR
        JSR BINBCD          ;SUBROUTINE BINARY TO BCD CONVERSION;
        LD A,BCDLO
        X  A,TEMP
        LD A,BCDLO+1
        X  A,TEMP+1
        RET
ERR:   LD  A,#00E           ;ERROR MESSAGE IS DISPLAYED
        X  A,TEMP
        CLR A
        X  A,TEMP+1
        RET
;****************************************************************
COMP:LD     A,COMPL         ;IF THE LAST BOTH MEASURMENTS ARE EQUAL
        SC                  ;THEN DISPLAY
        SUBC  A,TEMP
        IFEQ  A,#0
        JP    DISPLAY
        RET                 ;OTHERWISE DISPLAY THE OLD VALUE
DISPLAY:LD  A,TEMP
        X     A,PB+2
        LD    A,TEMP+1
M1:     X     A,PB+3
        JSR   LCDDR         ;UPDATE THE DISPLAY
        JSR   DEL           ;DELAY TIME
        RET
;****************************************************************
SAVE: LD    A,TEMP          ;TEMPORARY SAVE
        X     A,COMPL
        LD    A,TEMP+1
        X     A,COMPH
        RET
;****************************************************************
```

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.