Using the ADC Functions in the HT66Fx0

D/N：AN0194E

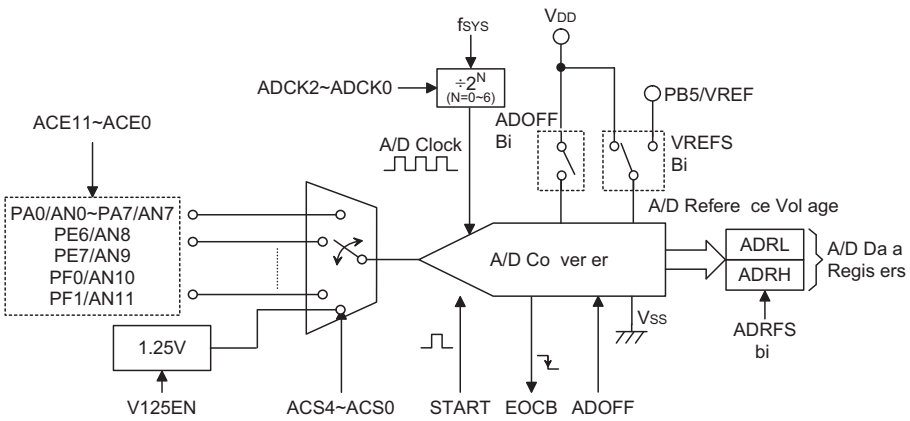
Introduction

The HT66FX0 series MCUs contain an 8-channel 12-bit resolution A/D converter that is able to complete an A/D conversion within 8us. The following provides a description of how to use the A/D functions in the HT66Fx0 series MCUs.

Taking the HT66F40 as an example, the eight A/D channels will be all opened and an A/D conversion initiated after which the conversion values will be saved to illustrate multi channel data collection. It will also be shown how to use the 1.25V reference source to implement accurate measurements.

Operating Principles

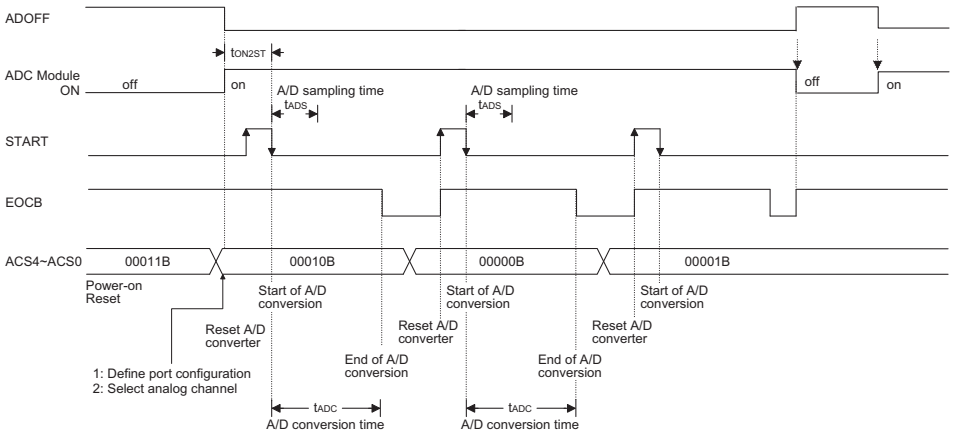
The core of the A/D module in the HT66F40 is an internal12-bit A/D converter. See the diagram below:



The A/D converter power supply is provided by AVDD and AVSS separately connected to VDD and VSS for the general condition. The reference voltage for the A/D conversion is provided by VREF or AVDD. The VREF voltage cannot be larger than AVDD+0.1V. The ADRH and ADRL registers are used to store the 12-bit A/D conversion result. The ADCR0, ADCR1 and ACERL registers control the A/D configuration and operation. The ADCK0~ADCK2 bits in the ADCR1 register are used to setup an A/D clock to ensure accurate A/D conversion. The A/D clock cycle should be no less than 0.5us. The ADOFF bit in the ADCR0 controls the on/off function of the A/D module. The ACE0~ACE7 bits in

the ACERL register sets up the PA port to be either A/D inputs or general I/O pins. The ACS0~ACS2 bits in the ADCR0 register and the ACS4 bit in the ADCR1 register control which channel is connected to the A/D converter, with one of the channels being the internal 1.25V reference voltage. START is the A/D conversion enable bit, changing it as follows: 0→1→0 will start the A/D conversion. EOCB is the end of the conversion flag, when this bit is 0, it means the conversion has completed, otherwise the conversion is still in progress.

A/D Conversion Timing Diagram:

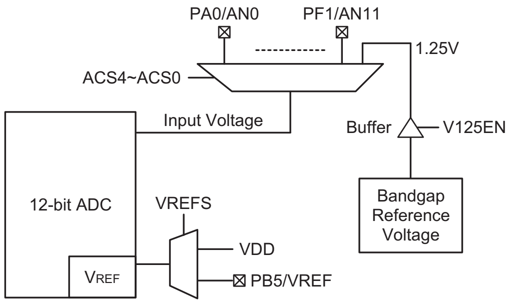


The ADC sampling time takes 4 t_{AD} and one conversion requires 16 t_{AD}. The parameter t_{AD} has a minimum value of 0.5us and maximum value of 10us, so it takes only 16 x 0.5us = 8us to complete an A/D conversion. For different system frequencies, it should be noted that t_{AD} should be set between 0.5us ~ 10us. Refer to the following t_{AD} selection table when programming:

f_{SYS}	A/D Clock Period(t_{AD})							
	ADCK2~0 =000(f_{SYS})	ADCK2~0 =001(f_{SYS}/2)	ADCK2~0 =010(f_{SYS}/4)	ADCK2~0 =011(f_{SYS}/8)	ADCK2~0 =100(f_{SYS}/16)	ADCK2~0 =101(f_{SYS}/32)	ADCK2~0 =110(f_{SYS}/64)	ADCK2~0 =111
1MHz	1us	2us	4us	8us	x	x	x	Undefined
2MHz	500ns	1us	2us	4us	8us	x	x	Undefined
4MHz	x	500ns	1us	2us	4us	8us	x	Undefined
8MHz	x	x	500ns	1us	2us	4us	8us	Undefined
12MHz	x	x	x	667ns	1.33us	2.67us	5.33us	Undefined

"x" means not recommended as a t_{AD} of less than 0.5us or larger than 10us may lead to inaccurate conversion results.

The A/D module in the HT66F40 includes an internal 1.25V reference voltage which is provided as an accurate reference voltage and which is able to implement an accurate measurement of the power supply voltage and A/D sampling so as to prevent the power voltage variations from creating erroneous A/D conversion results.



A/D Input Structure

Accurate Sampling

Power Supply Voltage Measurement

Enable the internal 1.25V reference voltage according to the A/D conversion process mentioned above and set the channel as the 1.25V reference voltage. After the conversion, mark the sampling A/D value as ADR1.

By $\frac{V_{REF}}{2^{12}} = \frac{1.25V}{ADR1}$

$$\Rightarrow \quad V_{REF} = \frac{1.25V * 2^{12}}{ADR1} \cdots\cdots\cdots ①$$

As the internal 1.25V reference voltage is not sensitive to the power supply voltage and temperature, it is very stable. According to VREF which may be AVDD or input from PB5, an accurate voltage value little influenced by temperature can be acquired. If VREFS is LOW, the VREF will be sourced from AVDD.

Accurate A/D Sampling

From the A/D conversion procedure depicted above, select the corresponding channel to be measured. Mark the obtained A/D sampling value as ADR2 after the A/D conversion is completed. The voltage of the pin to be measured is V.

By $\frac{V_{REF}}{2^{12}} = \frac{V}{ADR2}$

and from above,

$$\frac{V}{ADR2} = \frac{1.25V}{ADR1}$$

$$\Rightarrow V = \frac{1.25V * ADR2}{ADR1} \cdots\cdots\cdots ②$$

From the description mentioned above, the internal 1.25V reference voltage is very stable with little influence of temperature and power supply voltage. Therefore the acquired V is precise and will also be little affected by the temperature and power supply voltage.

From the formula ①, the result of ② can be managed by users according to actual requirement.

Special Notes for Data Programming

To increase the accuracy of the voltage measurement, refer to the following steps:

- Scan the channel 2n times and add all the sampled A/D values. After summing and then shifting by n bits will bring about a more precise and stable A/D value and thus decrease the error due to the A/D conversion.
- According to the formula above, it is recommended to multiply and then to divide the values, and increase the 1.25V for 1000 times or more before executing the multiply operation. This can prevent measurement errors from ignoring A/D sampling values in the process of division operations.
- With regard to the internal 1.25V reference voltage characteristic for each specific IC, it is recommended to use a precise power supply to measure the actual voltage of the internal 1.25V reference voltage and then use it as a parameter for the program to raise the accuracy of the measurements.

A/D Sampling Steps

The A/D module in the HT66F40 MCU contains an internal 1.25V voltage circuit which allows more precise A/D sampling values without being affected by temperature and power supply voltage. A complete A/D sampling process consists of two parts. One is to execute an A/D conversion on the internal 1.25V reference voltage channel and to save the A/D values as an A/D calibration value for the other channels. The VREF value can also be obtained through formula ①. The other is to execute A/D sampling on the AN0~AN7 pins and calibrate the A/D sampling values with formula ②.

The following provides the accurate A/D sampling steps for multiple channels. Designers may have to modify the content according to actual requirements. The internal 1.25V reference voltage sampling can be ignored if the power voltage and temperature of the application has little importance or accurate A/D sampling is not necessary.

Step 1 : Set the VREFS in the ADCR1 as the reference voltage of the A/D conversion. 0=VDD, 1=VREF input from PB5. Select the ADCK0~ADCK2 in the ADCR1 register as the A/D conversion clock. Detailed parameters are shown in the table below:

ADCK2	ADCK1	ADCK0	A/D Clock
0	0	0	f_{SYS}
0	0	1	$f_{SYS}/2$
0	1	0	$f_{SYS}/4$
0	1	1	$f_{SYS}/8$
1	0	0	$f_{SYS}/16$
1	0	1	$f_{SYS}/32$
1	1	0	$f_{SYS}/64$
1	1	1	Undefined - not used

Step 2 : Clear the ADOFF bit in the ADCR0 register to zero to enable the A/D conversion.

Step 3 : Set the ACS4 and V125EN in the ADCR1 register to 1, select the internal 1.25V reference voltage channel and enable the 1.25V reference voltage. Delay t_{BG}.

Step 4 : Setup the ACE0~ACE7 bits in the ACERL register and select the pins to be A/D input pins or common I/O pins. For detailed parameters see the table below.

Bit	7	6	5	4	3	2	1	0
Name	ACE7	ACE6	ACE5	ACE4	ACE3	ACE2	ACE1	ACE0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	1	1	1	1	1	1	1	1

Step 5 : The interrupt control register should be setup correctly when using the A/D interrupt so as to enable the A/D interrupt. The master interrupt control bit EMI and the A/D interrupt enable bit ADE should be both set to 1.

Step 6 : Set the START bit in the ADCR0 register from Low to High and then change to Low to initiate an A/D conversion. Note that the initial START bit should be Low.

Step 7 : Check the EOCB bit in the ADCR0 register to determine if the A/D conversion has completed. The EOCB bit will be high during the conversion process and low when the A/D conversion is completed. The other way to determine the completion of an A/D conversion is to use the A/D interrupt. If the A/D interrupt is enabled and the stack is not full, when the A/D conversion is completed, an A/D interrupt will be generated. After this ADRL and ADRH can be read.

Step 8 : Setup the ACS0~ACS2 and ACS4 in the ADCR0 register and select a channel to implement an A/D conversion. Detailed parameters for the ACS0~ACS2 and ACS4 bits are shown in the table below:

ACS4	ACS2	ACS1	ACS0	Analog Channel
0	0	0	0	AN0
0	0	0	1	AN1
0	0	1	0	AN2
0	0	1	1	AN3
0	1	0	0	AN4
0	1	0	1	AN5
0	1	1	0	AN6
0	1	1	1	AN7
1	x	x	x	1.25V

Step 9 : same as Step 6 and Step 7.

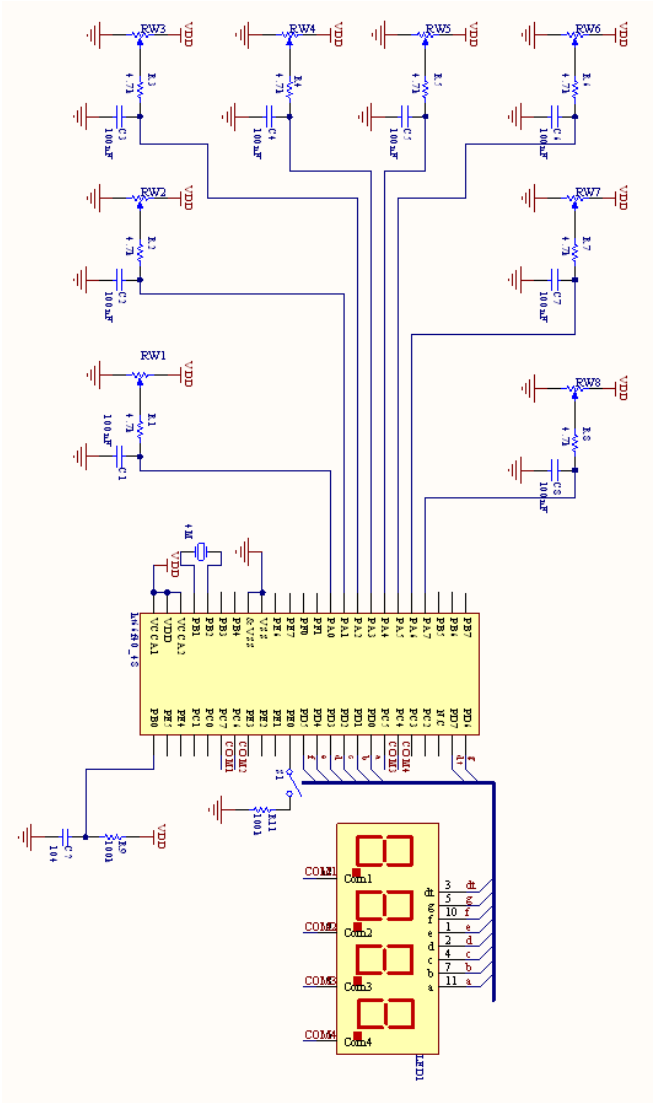
Step 10 : If multi-channel sampling is required, modify the channel as in Step 8 and then execute the Step 9 operation.

Take different steps according to the procedures above and different requirements. A summary is listed as the table below:

Occasion	VREF Measurement	Single Channel (No Calibration)	Multi-channel (No Calibration)	Single Channel (With Calibration)	Multi-channel (With Calibration)
Step Order	Step 1	Step 1	Step 1	Step 1	Step 1
	Step 2	Step 2	Step 2	Step 2	Step 2
	Step 3	Step 8	Step 8	Step 3	Step 3
	Step 5	Step 4	Step 4	Step 4	Step 4
	Step 6	Step 5	Step 5	Step 5	Step 5
	Step 7	Step 6	Step 6	Step 6	Step 6
		Step 7	Step 7	Step 7	Step 7
			Step 10	Step 8	Step 8
				Step 9	Step 9
					Step 10

Circuits

The following diagram shows an application example using multi-channel voltage (with calibration) measurement. Through this example, users should have a clearer concept for the A/D conversion of the HT66Fx0 series MCU.



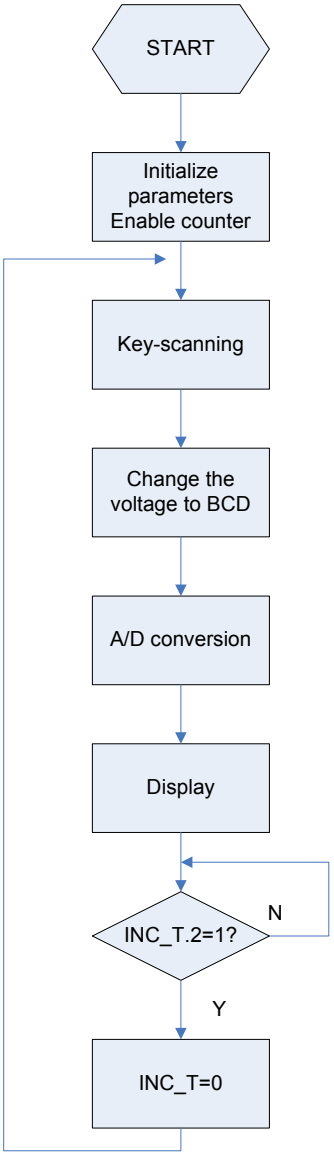
Circuit Description

Two example programs are provided one for the polling method and one for the interrupt method. These simple applications will introduce how to use the A/D function in the HT66F40 to implement multi-channel sampling using the polling flag and the interrupt.

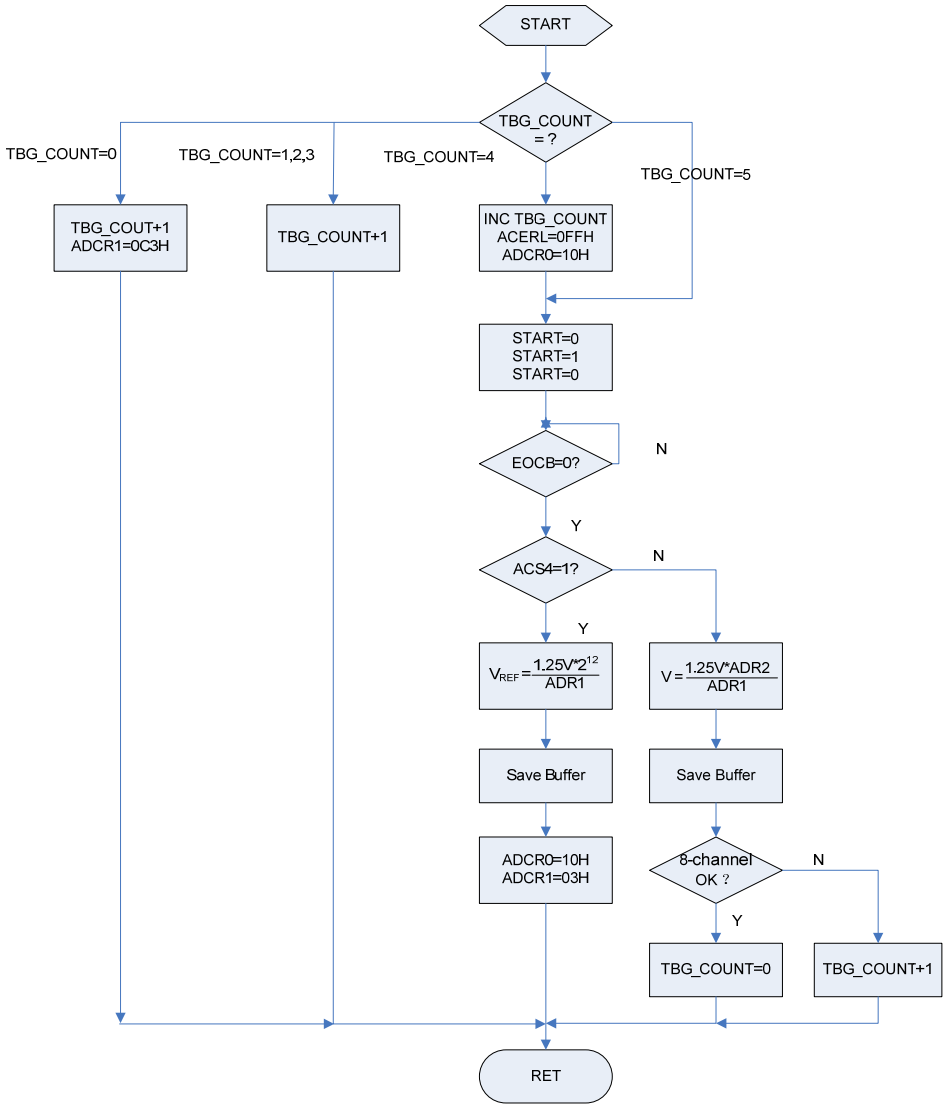
Press S1 to change an A/D sampling channel. Show the current sampled value on display 0, and the channel voltages on display 1 and 2.

Connect the AVDD and AVSS of the A/D module to VDD and VSS. Select the A/D conversion reference voltage VREF as VDD.

Software Flowchart (Polling Method)



Master flowchart of the multi-channel temperature sampling



Multi-channel A/D calibration (polling method) flowchart

Program Description (Polling Method)

Main flowchart of the multi-channel voltage sampling

Continuously execute the A/D sampling and calibration on the internal 1.25V reference voltage channel and the 8 AN channels. Output the corresponding channel voltage using the key scanning value. Show the current channel on display 3 and the voltage of the current channel on display 2 and 1.

A/D conversion flow by Polling Method

- A/D Setup

  Setup ADCR1: ADCK0~ADCK2, these three bits are used to select the clock frequency of the A/D conversion. The VREFS bit is the A/D reference voltage selection bit, with 0 as ADD and 1 as the VREF pin. The V125EN bit is the internal 1.25V reference voltage control bit. The ACS4 bit is the internal 1.25V reference voltage channel control bit.

  Setup ACERL: ACE0~ACE7, these eight bits control if the PA port is used as an I/O or A/D input.

  Setup ADCR0: ACS0~ACS2, these three bits are used to switch the conversion channel. The ADRFS bit is to control the stored data format. The ADOFF bit controls the on/off function of the A/D. The EOCB is the end of conversion flag. The START bit is the A/D conversion start bit.

- Enable A/D
  Set the START bit Low and then change it from High to Low to initiate an A/D conversion.

- EOCB
  EOCB is High during an A/D conversion process and will switch to Low when the conversion is complete.

A/D sampling operation flow by polling method

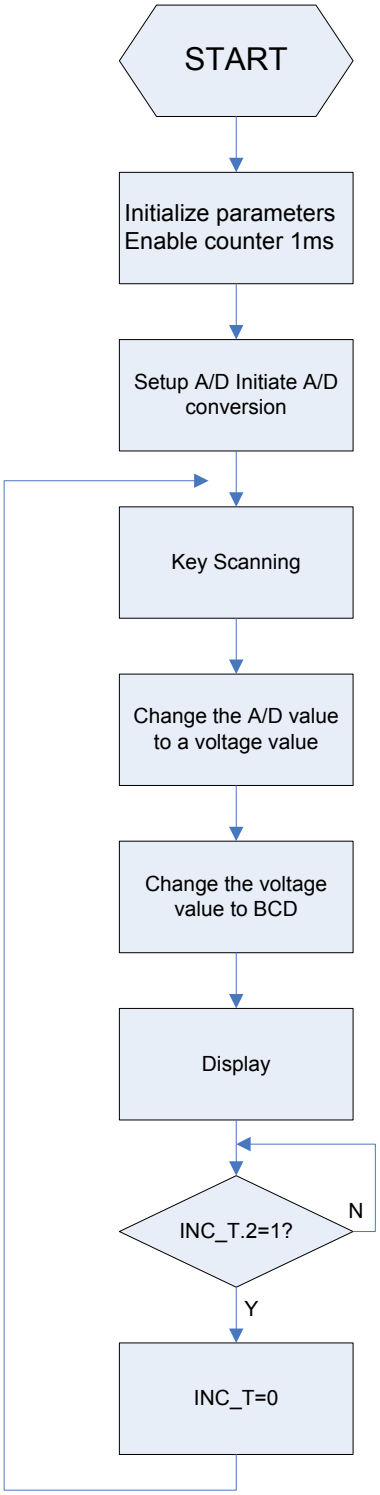
First take the A/D value of the internal 1.25V channel and save it with formula① to get a VREF value. Follow up with a scanned value of the other eight AN pins and calibrate the sampled A/D values with formula ② and convert to corresponding voltage values.

Program Example (the Polling Method)

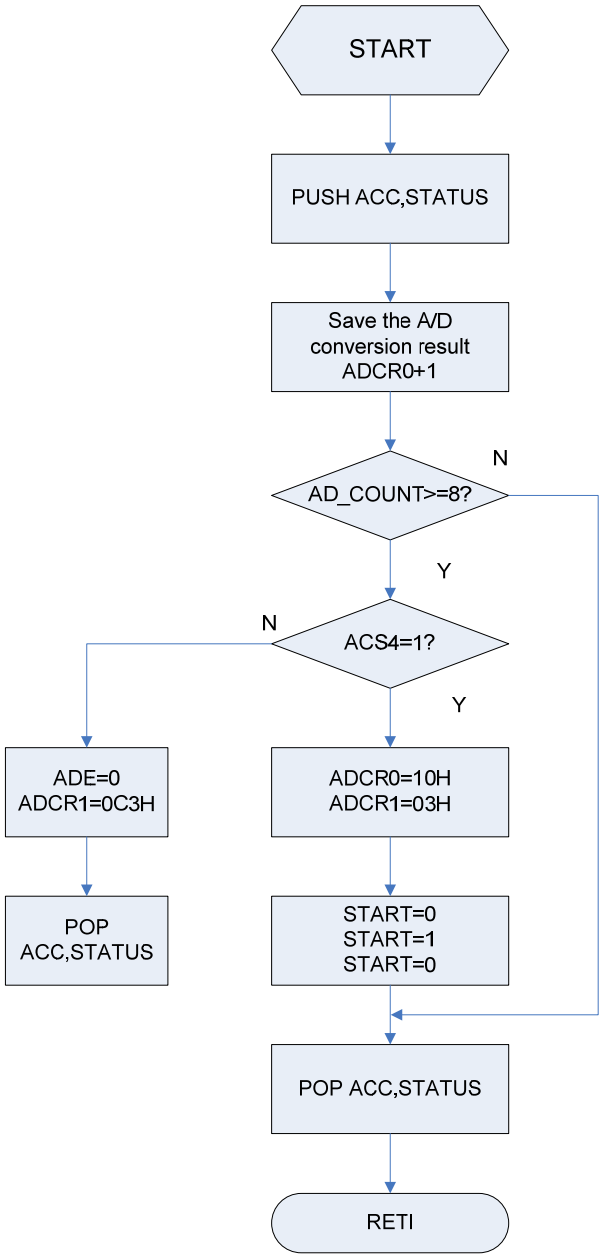
```
configuration option：
High Freq.OSC:       XTAL
Low Freq.OSC:        LIRC
Fsub clock source:   LIRC
WDT:                 Disable
;other option select by user.
```

See the attachment for program codes and description.

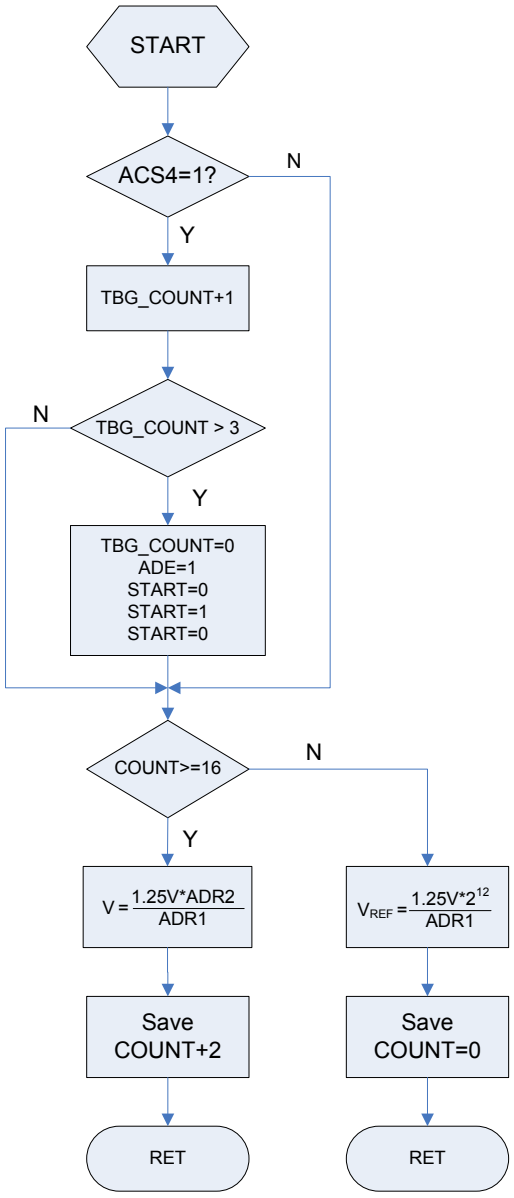
Software Flowchart (the Interrupt Method)



Master flowchart of Multi-channel temperature sampling



A/D sampling flowchart by A/D interrupt method



A/D values handling flow with A/D interrupt method

Program Description (Interrupt Method)

Initialize the A/D after the program starts by setting the ADCR1, ADCR0, ACERL bits and then initiate an A/D conversion, after first enabling the A/D interrupt and master interrupt control bits. Repeat the key scanning after initialization and display the A/D sampled results. Use formula ① and ② to handle the A/D values and obtain the voltage values of the eight channels.

As long as an A/D interrupt occurs, enter the A/D interrupt subroutine to back up the ACC and STATUS registers and save the results of the conversion. Decide which channel has interrupted, and set the corresponding channel to enable an A/D conversion, return ACC and STATUS, and exit the interrupt service subroutine. If scanning is completed, re-select a channel, initiate an A/D conversion, return ACC and STATUS, and exit the interrupt service subroutine.

Program Example (the Interrupt Method)

```
configuration option：
High Freq.OSC:                  XTAL
Low Freq.OSC:                   LIRC
Fsub clock source:              LIRC
WDT:                            Disable
;other option select by user
```

See the attachment for program codes and description.

Conclusion

This application note has shown two ways of implementing A/D sampling with the HT66F40 using the voltage sampling methods above.