Microcontrollers

ADC

Dr. Jafar Jallad

Palestine Technical University – Kadoorie

Second semester

2019-2020

ADC devices

Analog-to-digital converters are among the most widely used devices for data acquisition. Digital computers use binary (discrete) values, but in the physical world everything is analog (continuous). Temperature, pressure (wind or liquid), humidity, and velocity are a few examples of physical quantities that we deal with every day. A physical quantity is converted to electrical (voltage, current) signals using a device called a *transducer*. Transducers are also referred to as *sensors*. Sensors for temperature, velocity, pressure, light, and many other natural quantities produce an output that is voltage (or current). Therefore, we need an analog-to-digital converter to translate the analog signals to digital numbers so that the microcontroller can read and process them.

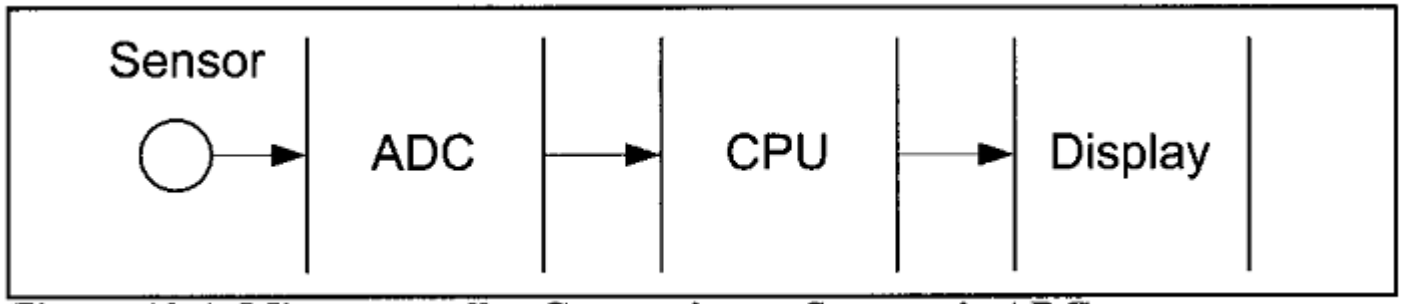


Figure 13-1. Microcontroller Connection to Sensor via ADC

the major characteristics of the ADC are as follows:

Resolution

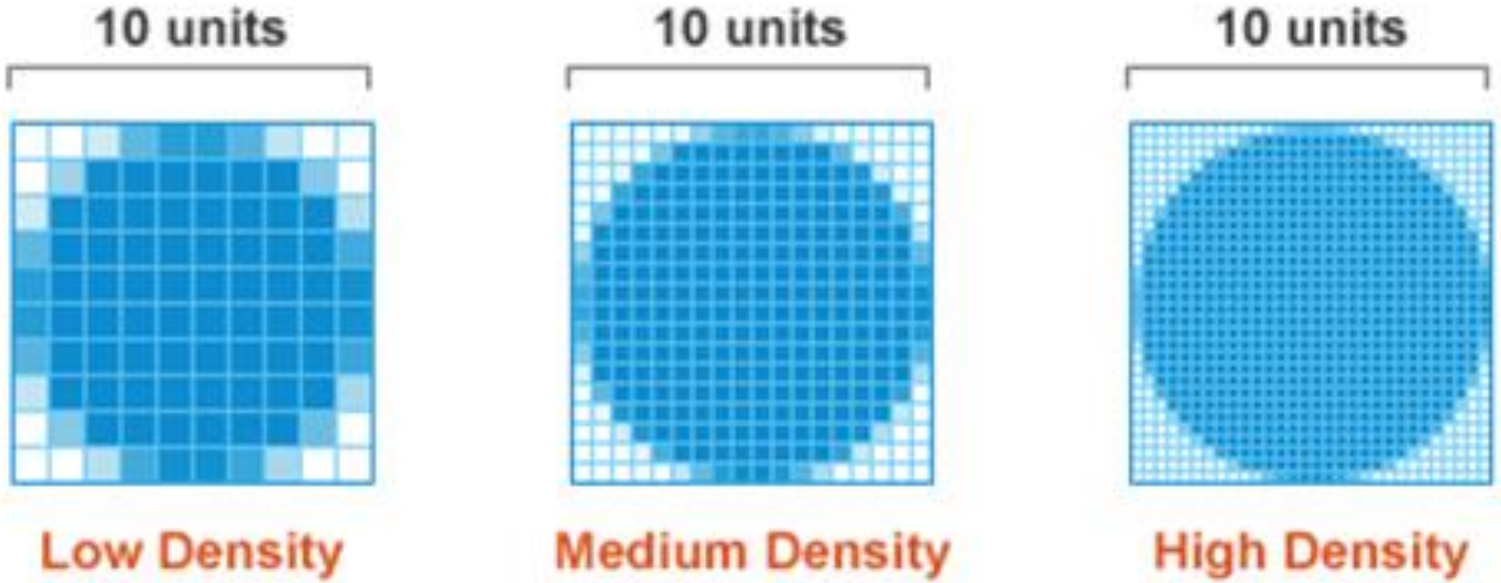
ADC has *n*-bit resolution, where *n* can be 8, 10, 12, 16, or even 24 bits. The higher-resolution ADC provides a smaller step size, where *step size* is the smallest change that can be discerned by an ADC. Some widely used resolutions for ADCs are shown in Table 13-1. Although the resolution of an ADC chip is decided at the time of its design and cannot be changed, we can control the step size with the help of what is called Vref. This is discussed below.

Table 13-1: Resolution versus Step Size for ADC (Vref = 5 V)

n-bit	Number of steps	Step size (mV)
8	256	5/256 = 19.53
10	1,024	5/1,024 = 4.88
12	4,096	5/4,096 = 1.2
16	65,536	5/65,536 = 0.076

Notes: V_{CC} = 5 V

Step size (resolution) is the smallest change that can be discerned by an ADC.



Conversion time

In addition to resolution, conversion time is another major factor in judging an ADC. *Conversion time* is defined as the time it takes the ADC to convert the analog input to a digital (binary) number. The conversion time is dictated by the clock source connected to the ADC in addition to the method used for data conversion and technology used in the fabrication of the ADC chip such as MOS or TTL technology.

V_{ref}

V_{ref} is an input voltage used for the reference voltage. The voltage connected to this pin, along with the resolution of the ADC chip, dictate the step size. For an 8-bit ADC, the step size is $V_{ref}/256$ because it is an 8-bit ADC, and 2 to the power of 8 gives us 256 steps. See Table 13-1. For example, if the analog input range needs to be 0 to 4 volts, V_{ref} is connected to 4 volts. That gives 4 V/256 = 15.62 mV for the step size of an 8-bit ADC. In another case, if we need a step size of 10 mV for an 8-bit ADC, then V_{ref} = 2.56 V, because 2.56 V/256 = 10 mV. For

Table 13-2: V_{ref} Relation to V_{in} Range for an 8-bit ADC

V_{ref} (V)	V_{in} (V)	Step Size (mV)
5.00	0 to 5	5/256 = 19.53
4.0	0 to 4	4/256 = 15.62
3.0	0 to 3	3/256 = 11.71
2.56	0 to 2.56	2.56/256 = 10
2.0	0 to 2	2/256 = 7.81
1.28	0 to 1.28	1.28/256 = 5
1	0 to 1	1/256 = 3.90

Table 13-3: V_{ref} Relation to V_{in} Range for an 10-bit ADC

V_{ref} (V)	V_{in} (V)	Step Size (mV)
5.00	0 to 5	5/1,024 = 4.88
4.096	0 to 4.096	4.096/1,024 = 4
3.0	0 to 3	3/1,024 = 2.93
2.56	0 to 2.56	2.56/1,024 = 2.5
2.048	0 to 2.048	2.048/1,024 = 2
1.28	0 to 1.28	1/1,024 = 1.25
1.024	0 to 1.024	1.024/1,024 = 1

Digital data output

In an 8-bit ADC we have an 8-bit digital data output of D0–D7 while in the 10-bit ADC the data output is D0–D9. To calculate the output voltage, we use the following formula:

$$D_{out} = \frac{V_{in}}{step\ size}$$

where Dout = digital data output (in decimal), Vin = analog input voltage, and step size (resolution) is the smallest change, which is Vref/256 for an 8-bit

Example 13-1

For an 8-bit ADC, we have $V_{ref} = 2.56$ V. Calculate the D0–D7 output if the analog input is: (a) 1.7 V, and (b) 2.1 V.

Solution:

Because the step size is 2.56/256 = 10 mV, we have the following:

(a) D_{out} = 1.7 V/10 mV = 170 in decimal, which gives us 10101011 in binary for D7–D0.

(b) D_{out} = 2.1 V/10 mV = 210 in decimal, which gives us 11010010 in binary for D7–D0.

Parallel versus serial ADC

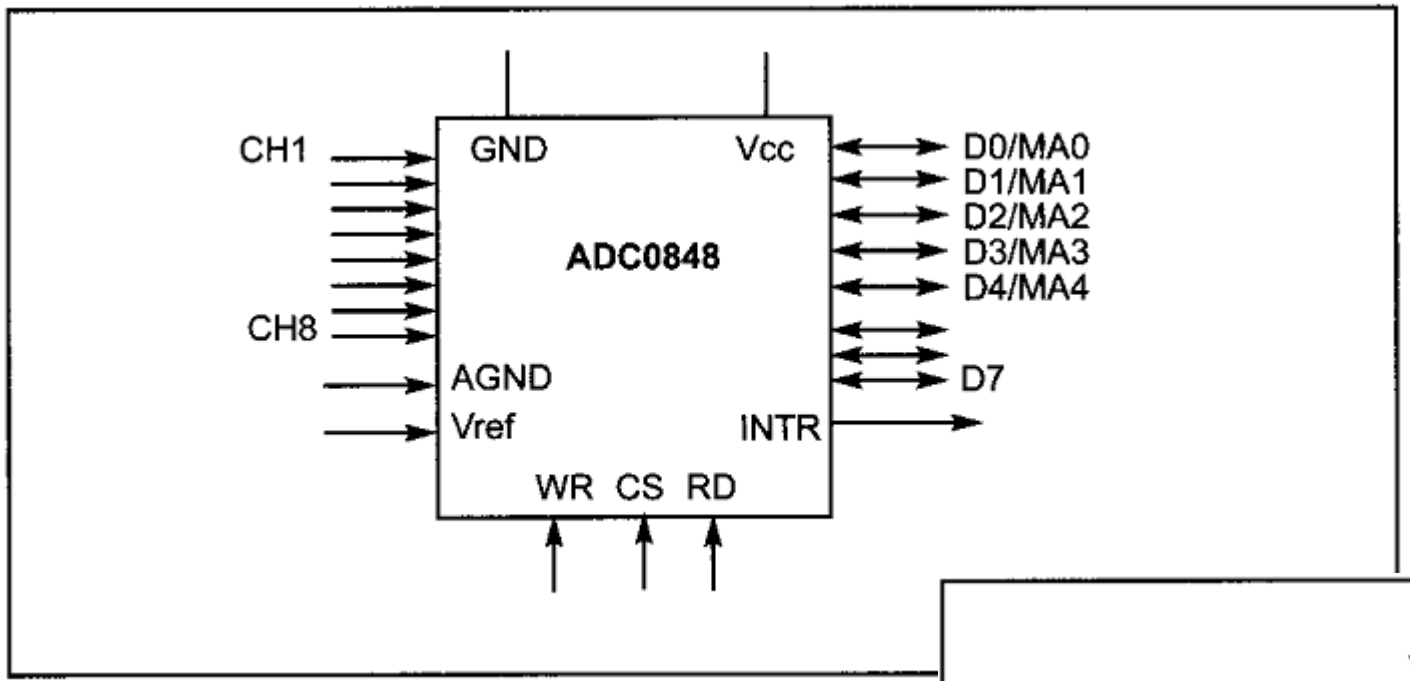


Figure 13-3. ADC0848 Parallel ADC Block Diagram

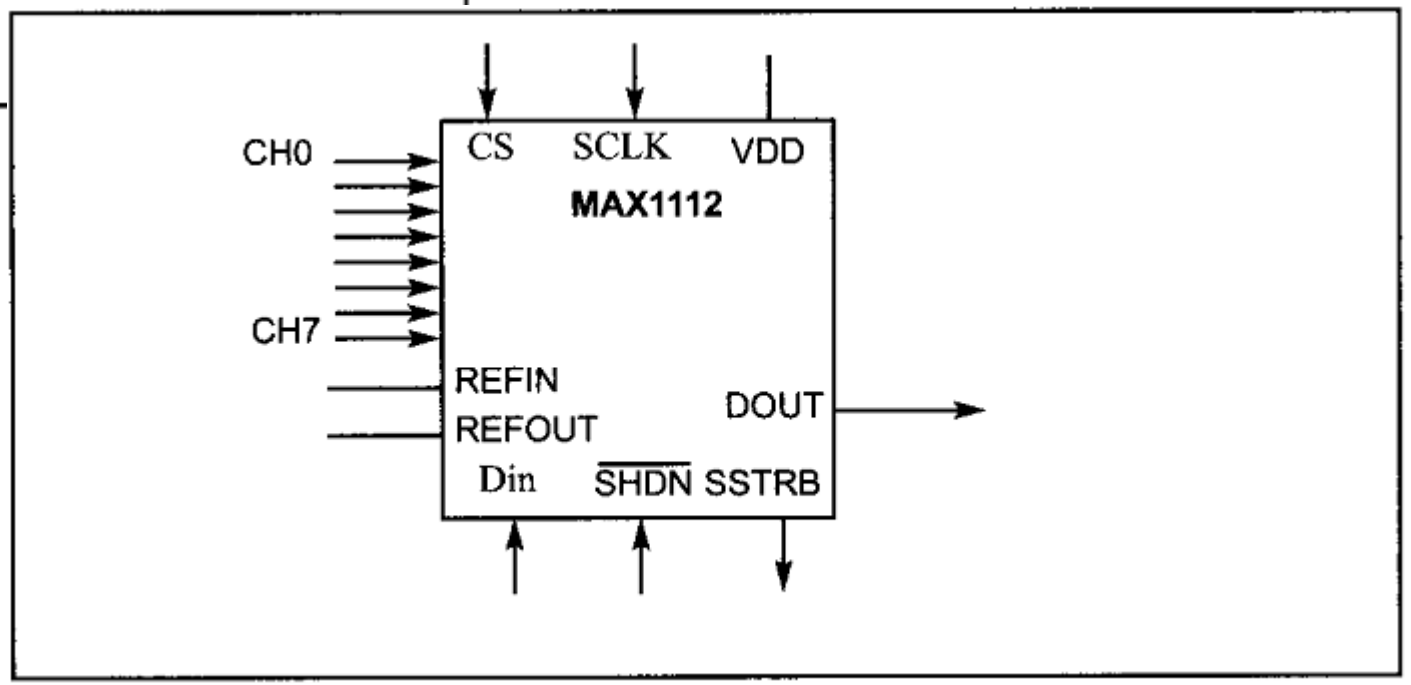


Figure 13-4. MAX1112 Serial ADC Block Diagram

Start conversion and end-of-conversion signals

The fact that we have multiple analog input channels and a single digital output register makes it necessary for start conversion (SC) and end-of-conversion (EOC) signals. When SC is activated, the ADC starts converting the analog input value of Vin to an *n*-bit digital number. The amount of time it takes to convert varies depending on the conversion method as was explained earlier. When the data conversion is complete, the end-of-conversion signal notifies the CPU that the converted data is ready to be picked up.

From the discussion we conclude that the following steps must be followed for data conversion by an ADC chip:

1. Select a channel.
2. Activate the start conversion (SC) signal to start the conversion of analog input.
3. Keep monitoring the end-of-conversion (EOC) signal.
4. After the EOC has been activated, we read data out of the ADC chip.

Introduction

Microcontrollers are very useful especially when it comes to communicate with other devices, such as sensors, motors, switches, memory and even other microcontroller. As we all know many interface methods have been developed over years to solve complex problem of balancing need of features, cost, size, power consumption, reliability etc. but ADC Analog-to-Digital converter remains famous among all. Using this ADC we can connect any type of Analog sensor.

PIC16F877A ADC Module

The Analog-to-Digital (A/D) Converter module has eight for the 40/44-pin devices.

The conversion of an analog input signal results in a corresponding 10-bit digital number. The A/D module has high and low-voltage reference input that is software selectable to some combination of VDD, VSS, RA2 or RA3.

The A/D converter has a unique feature of being able to operate while the device is in Sleep mode. To operate in Sleep, the A/D clock must be derived from the A/D's internal RC oscillator.

PIC16F877A ADC Pins

ADC CHANNEL	PIN
Channel 0	RA0/AN0 (Port A)
Channel 1	RA1/AN1 (Port A)
Channel 2	RA2/AN2/VRef- (Port A)
Channel 3	RA3/AN3/VRef+ (Port A)
Channel 4	RA5/AN4 (Port A)
Channel 5	RE0/AN5 (Port E)
Channel 6	RE1/AN6 (Port E)
Channel 7	RE2/AN7 (Port E)

Registers used for ADC

- A/D Control Register 0 (ADCON0)
- A/D Control Register 1 (ADCON1)
- A/D Result High Register (ADRESH)
- A/D Result Low Register (ADRESL)

A/D Control Register 0 (ADCON0)

The ADCON0 register, shown in below image, controls the operation of the A/D module i.e. Used to Turn ON the ADC, Select the Sampling Freq and also Start the conversion.

ADCON0 REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	—	ADON
bit 7							bit 0

ADCON0 REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	—	ADON
bit 7							bit 0

ADCON1 REGISTER

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	ADCS2	—	—	PCFG3	PCFG2	PCFG1	PCFG0
bit 7							bit 0

ADCS1-ADCS0: A/D Conversion Clock Select bits This bits are based on ADCON1 Reister's ADCS2 bit.

ADCON1 <ADCS2>	ADCON0 <ADCS1:ADCS0>	Clock Conversion
0	00	Fosc/2
0	01	Fosc/8
0	10	Fosc/32
0	11	FRC (clock derived from the internal A/D RC oscillator)
1	00	Fosc/4
1	01	Fosc/16
1	10	Fosc/64
1	11	FRC (clock derived from the internal A/D RC oscillator)

ADCON0 REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	—	ADON
bit 7							bit 0

CHS2-CHS0: Analog Channel Select bits

000 = Channel 0 (AN0)
001 = Channel 1 (AN1)
010 = Channel 2 (AN2)
011 = Channel 3 (AN3)
100 = Channel 4 (AN4)
101 = Channel 5 (AN5)
110 = Channel 6 (AN6)
111 = Channel 7 (AN7)

ADCON0 REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	—	ADON
bit 7							bit 0

GO/DONE: A/D Conversion Status bit

When ADON = 1:

1 = A/D conversion in progress (setting this bit starts the A/D conversion which is automatically cleared by hardware when the A/D conversion is complete)

0 = A/D conversion not in progress

ADON: A/D On bit

1 = A/D converter module is powered up

0 = A/D converter module is shut-off and consumes no operating current

A/D Control Register 1 (ADCON1)

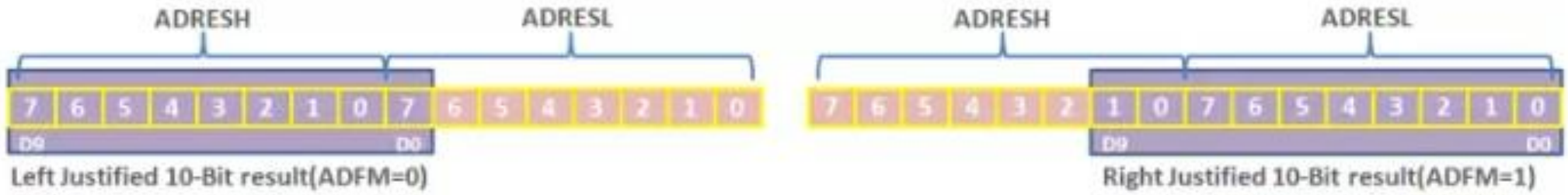
The ADCON1 register, shown below, configures the functions of the port pins i.e Used to configure the GPIO pins for ADC. The port pins can be configured as analog inputs (RA3 can also be the voltage reference) or as digital I/O.

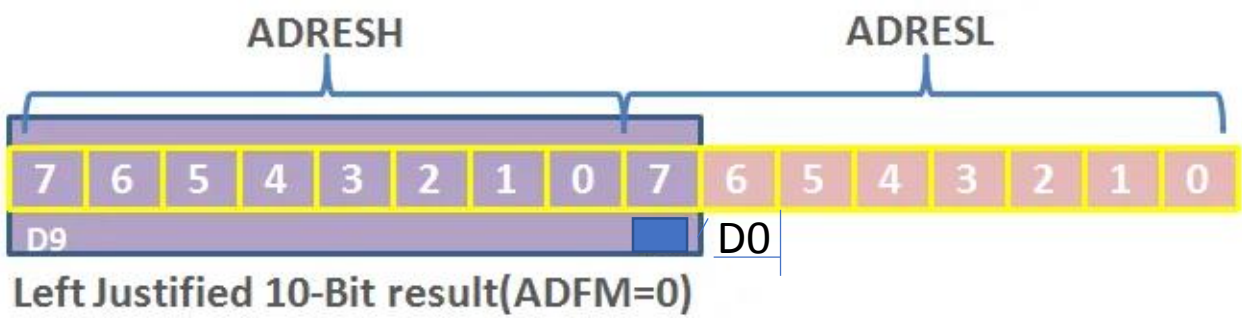
ADCON1 REGISTER

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	ADCS2	—	—	PCFG3	PCFG2	PCFG1	PCFG0
bit 7							bit 0

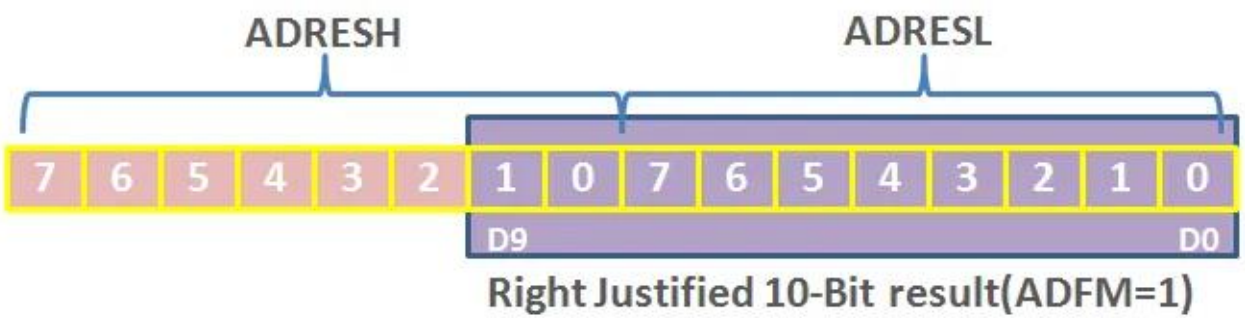
ADFM: A/D Result Format Select bit

1 = Right justified. Six (6) Most Significant bits of ADRESH are read as '0'.
0 = Left justified. Six (6) Least Significant bits of ADRESL are read as '0'.





Left Justified 10-Bit result(ADFM=0)



Right Justified 10-Bit result(ADFM=1)

ADCS2: A/D Conversion Clock Select bit

ADCON1 <ADCS2>	ADCON0 <ADCS1:ADCS0>	Clock Conversion
0	00	Fosc/2
0	01	Fosc/8
0	10	Fosc/32
0	11	FRC (clock derived from the internal A/D RC oscillator)
1	00	Fosc/4
1	01	Fosc/16
1	10	Fosc/64
1	11	FRC (clock derived from the internal A/D RC oscillator)

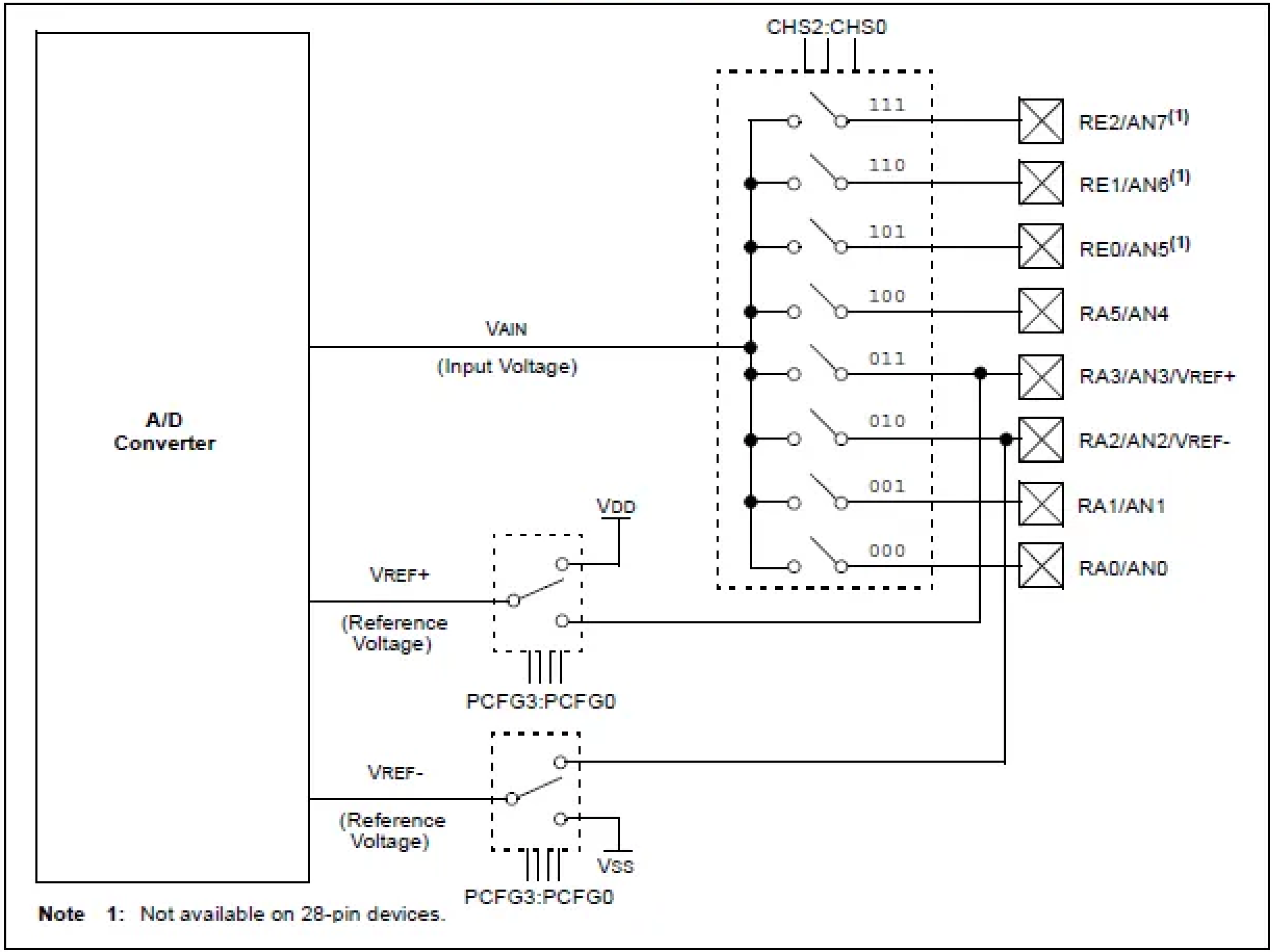
PCFG3-PCFG0: A/D Port Configuration Control bits

PCFG <3:0>	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	VREF+	VREF-	C/R
0000	A	A	A	A	A	A	A	A	VDD	VSS	8/0
0001	A	A	A	A	VREF+	A	A	A	AN3	VSS	7/1
0010	D	D	D	A	A	A	A	A	VDD	VSS	5/0
0011	D	D	D	A	VREF+	A	A	A	AN3	VSS	4/1
0100	D	D	D	D	A	D	A	A	VDD	VSS	3/0
0101	D	D	D	D	VREF+	D	A	A	AN3	VSS	2/1
011x	D	D	D	D	D	D	D	D	—	—	0/0
1000	A	A	A	A	VREF+	VREF-	A	A	AN3	AN2	6/2
1001	D	D	A	A	A	A	A	A	VDD	VSS	6/0
1010	D	D	A	A	VREF+	A	A	A	AN3	VSS	5/1
1011	D	D	A	A	VREF+	VREF-	A	A	AN3	AN2	4/2
1100	D	D	D	A	VREF+	VREF-	A	A	AN3	AN2	3/2
1101	D	D	D	D	VREF+	VREF-	A	A	AN3	AN2	2/2
1110	D	D	D	D	D	D	D	A	VDD	VSS	1/0
1111	D	D	D	D	VREF+	VREF-	D	A	AN3	AN2	1/2

A = Analog input D = Digital I/O

C/R = # of analog input channels/# of A/D voltage references

A/D BLOCK DIAGRAM



Note 1: Not available on 28-pin devices.

Example 13-2

For a PIC -based system, we have V_{ref} = Vdd = 5 V. Find (a) the step size, and (b) the ADCON1 value if we need 3 channels. Assume that the ADRESH:ADRESL registers are right justified.

Solution:

(a) The step size is 5/1,024 = 4.8 mV.
(b) ADCON1 = 1x000100 because option 100 gives us 3 analog input channels. The x = ADCS2 is decided by the conversion speed.

ADCON1 REGISTER

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	ADCS2	—	—	PCFG3	PCFG2	PCFG1	PCFG0

bit 7

PCFG <3:0>	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	VREF+	VREF-	C/R
0100	D	D	D	D	A	D	A	A	VDD	VSS	3/0

Example 13-3

For a PIC18-based system, we have V_{ref} = 2.56 V. Find (a) the step size, and (b) the ADCON1 value if we need 3 channels. Assume that the ADRESH:ADRESL registers are right justified.

Solution:

(a) The step size is 2.56/1,024 = 2.5 mV.
(b) ADCON1 = 1x000011 because option 0011 gives us 3 analog input channels where x = ADCS2 is decided by the conversion speed.

ADCON1 REGISTER

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	ADCS2	—	—	PCFG3	PCFG2	PCFG1	PCFG0

bit 7

PCFG <3:0>	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	VREF+	VREF-	C/R
0011	D	D	D	A	VREF+	A	A	A	AN3	VSS	4/1

Calculating A/D conversion time

ADCON1 registers we can set the A/D conversion time. The conversion time is defined in terms of Tad, where Tad is the conversion time per bit. To calculate the Tad, we can select a conversion clock source of Fosc/2, Fosc/4, Fosc/8, Fosc/16, Fosc/32, or Fosc/64, where Fosc is the speed of the crystal frequency connected to the PIC chip. For the PIC18, the conversion time is 12 times the Tad. Notice that the Tad cannot be faster than 1.6 ms.

We can also use the the internal RC oscillator for the conversion clock source, instead of the Fosc of the external crystal oscillator. In that case the Tad is typically 4–6 μs and conversion time is 12 × 6 μs = 72 μs.

Another timing factor that we must pay attention to is the acquisition time (Tacq). After an A/D channel is selected, we must allow some time for the sample-and-hold capacitor (C hold) to charge fully to the input voltage level present at the channel. It is only after the elapsing of this acquisition time that the A/D conversion can be started. Although many factors (e.g., Vdd and temperature) affect the duration of Tacq, we can use a typical value of 15 μs.

Example 13-4

A PIC is connected to the 10 MHz crystal oscillator. Calculate the conversion time for all options of ADCS bits in both the ADCON0 and ADCON1 registers.

Solution:

The options for the conversion clock source for both ADCON0 and ADCON1 are as follows:

(a) For Fosc/2, we have 10 MHz / 2 = 5 MHz.
Tad = 1 / 5 MHz = 200 ns. Invalid because it is faster than 1.6 μs.
(b) For Fosc/4, we have 10 MHz / 4 = 2.5 MHz.
Tad = 1 / 2.5 MHz = 400 ns. Invalid because it is faster than 1.6 μs.
(c) For Fosc/8, we have 10 MHz / 8 = 1.25 MHz.
Tad = 1 / 2.5 MHz = 800 ns. Invalid because it is faster than 1.6 μs.
(d) For Fosc/16, we have 10 MHz / 16 = 625 kHz.
Tad =1 / 625 kHz = 1.6 μs. The conversion time = 12 × 1.6 μs = 19.2 μs
(e) For Fosc/32, we have 10 MHz / 32 = 312.5 kHz.
Tad = 1 / 312.5 kHz = 3.2 μs. The conversion time = 12 × 3.2 μs = 38.4 μs
(f) For Fosc/64, we have 10 MHz / 64 = 156.25 kHz.
Tad = 1 / 156.25 kHz = 6.4 μs. The conversion time = 12 × 6.4 μs = 76.8 μs
Notice that for the Fosc/4, Fosc/16, and Fosc/64 selections, we must use the ADSC2 bit in the ADCON1 register, in addition to the ADCS bits in the ADCON0 register.

Example 13-5

A PIC is connected to the 4 MHz crystal oscillator. Calculate the conversion time if we want to use only the ADCS bits of the ADCON0 register.

Solution:

The options for the conversion clock source available in the ADCON0 register are as follows:

(a) For Fosc/2, we have 4 MHz / 2 = 2 MHz.

Tad = 1 / 2 MHz = 400 ns. Invalid because it is faster than 1.6 µs.

(b) For Fosc/8, we have 4 MHz / 8 = 500 kHz.

Tad = 1 / 500 kHz = 2 µs. The conversion time = 12 × 2 µs = 24 µs

(c) For Fosc/32, we have 4 MHz / 32 = 125 kHz.

Tad = 1 / 125 kHz = 8 µs. The conversion time = 12 × 8 µs = 96 µs

Example 13-6

Find the values for the ADCON0 and ADCON1 registers for the following options: (a) channel AN0 as analog input, (b) Vref+ = Vdd, Vref– = Vss, (c) Fosc/64, (d) A/D result is right justified, and (e) A/D module is on.

Solution:

From Figure 13-6, we have ADCON0 = 10000x1. With x = 0 we have 10000001.
From Figure 13-7, we have ADCON1 = 11xx1110. With x = 0 we have 11001110.

ADCON1 REGISTER

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	ADCS2	—	—	PCFG3	PCFG2	PCFG1	PCFG0
bit 7							bit 0

PCFG <3:0>	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	VREF+	VREF-	C/R
0000	A	A	A	A	A	A	A	A	VDD	VSS	8/0
0001	A	A	A	A	VREF+	A	A	A	AN3	VSS	7/1
0010	D	D	D	A	A	A	A	A	VDD	VSS	5/0
0011	D	D	D	A	VREF+	A	A	A	AN3	VSS	4/1
0100	D	D	D	D	A	D	A	A	VDD	VSS	3/0
0101	D	D	D	D	VREF+	D	A	A	AN3	VSS	2/1
011x	D	D	D	D	D	D	D	D	—	—	0/0
1000	A	A	A	A	VREF+	VREF-	A	A	AN3	AN2	6/2
1001	D	D	A	A	A	A	A	A	VDD	VSS	6/0
1010	D	D	A	A	VREF+	A	A	A	AN3	VSS	5/1
1011	D	D	A	A	VREF+	VREF-	A	A	AN3	AN2	4/2
1100	D	D	D	A	VREF+	VREF-	A	A	AN3	AN2	3/2
1101	D	D	D	D	VREF+	VREF-	A	A	AN3	AN2	2/2
1110	D	D	D	D	D	D	D	A	VDD	VSS	1/0
1111	D	D	D	D	VREF+	VREF-	D	A	AN3	AN2	1/2

ADCON0 REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	—	ADON
bit 7							bit 0

ADCS2: A/D Conversion Clock Select bit

ADCON1 <ADCS2>	ADCON0 <ADCS1:ADCS0>	Clock Conversion
0	00	Fosc/2
0	01	Fosc/8
0	10	Fosc/32
0	11	FRC (clock derived from the internal A/D RC oscillator)
1	00	Fosc/4
1	01	Fosc/16
1	10	Fosc/64
1	11	FRC (clock derived from the internal A/D RC oscillator)

Steps to follow

To do an A/D Conversion, follow these steps:

1. Configure the A/D module:

   - Configure analog pins/voltage reference and digital I/O (ADCON1)
   - Select A/D input channel (ADCON0)
   - Select A/D conversion clock (ADCON0)
   - Turn on A/D module (ADCON0)

2. Wait the required acquisition time.

3. Start conversion:

   - Set GO/DONE bit (ADCON0)

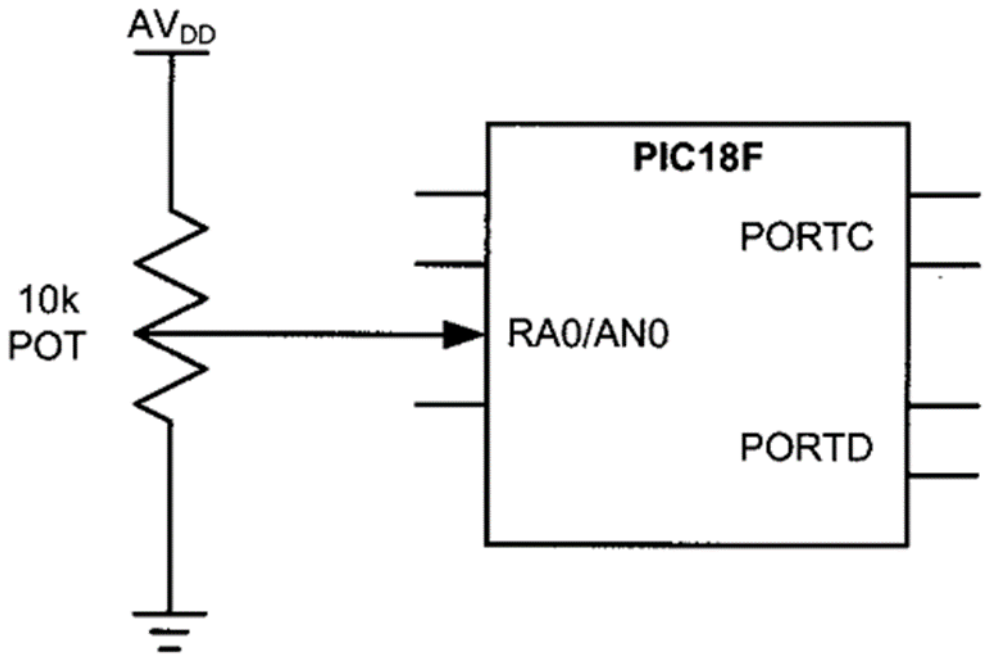
4. Wait for A/D conversion to complete by either:

- Polling for the GO/DONE bit to be cleared (interrupts disabled); OR
- Waiting for the A/D interrupt

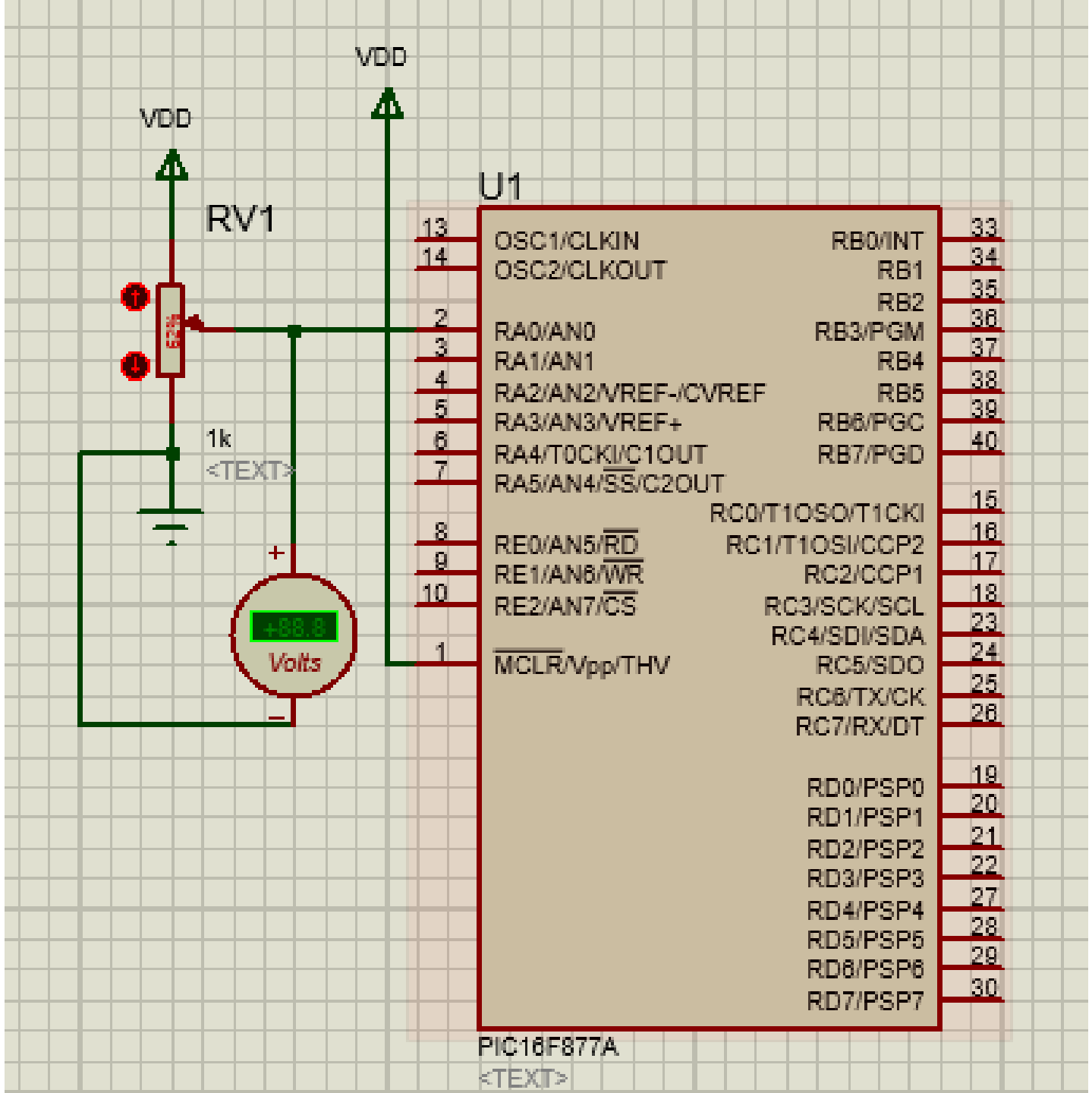
5. Read A/D Result register pair (ADRESH:ADRESL), clear bit ADIF if required.

6. For the next conversion, go to step 1 or step 2 as required.

Program 13-1: This program gets data from channel 0 (RA0) of ADC and displays the result on PORTC and PORTD. This is done every quarter of second.

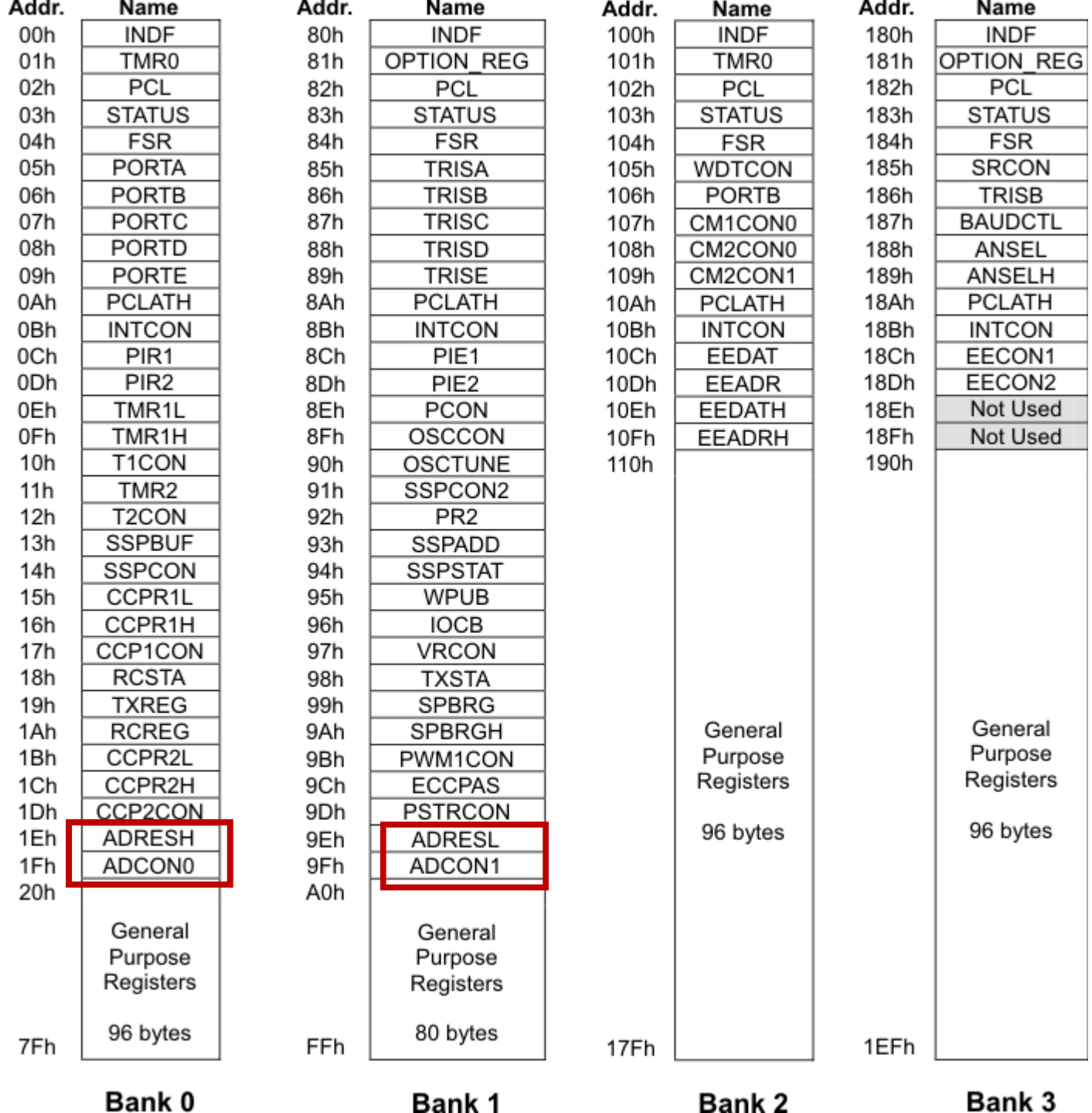


```
        CLRF   TRISC              ;make PORTC an output
        CLRF   TRISD              ;make PORTD an output
        BSF    TRISA,0 ;make RA0 an input for analog input
        MOVLW  0x81               ;Fosc/64, channel 0, A/D is on
        MOVWF  ADCON0
        MOVLW  0xCE  ;right justified, Fosc/64, AN0 = analog
        MOVWF  ADCON1
OVER    CALL   DELAY ;wait for Tacq (sample and hold time)
        BSF   ADCON0,GO           ;start conversion
BACK    BTFSC ADCON0,DONE         ;keep polling end-of-conversion
        GOTO BACK
        MOVF ADRESL, W
        MOVWF PORTC
        MOVF ADRESH, W
        MOVWF PORTD
        CALL   QSEC DELAY
        GOTO   OVER                 ;keep repeating it
        END
```


VDD
VDD
RV1
62%
1k
<TEXT>
+88.8
Volts
U1
13
14
2
3
4
5
6
7
8
9
10
1
OSC1/CLKIN
OSC2/CLKOUT
RA0/AN0
RA1/AN1
RA2/AN2/VREF-/CVREF
RA3/AN3/VREF+
RA4/T0CKI/C1OUT
RA5/AN4/SS/C2OUT
RE0/AN5/RD
RE1/AN6/WR
RE2/AN7/CS
MCLR/Vpp/THV
33
34
35
36
37
38
39
40
15
16
17
18
23
24
25
26
19
20
21
22
27
28
29
30
RB0/INT
RB1
RB2
RB3/PGM
RB4
RB5
RB6/PGC
RB7/PGD
RC0/T1OSO/T1CKI
RC1/T1OSI/CCP2
RC2/CCP1
RC3/SCK/SCL
RC4/SDI/SDA
RC5/SDO
RC6/TX/CK
RC7/RX/DT
RD0/PSP0
RD1/PSP1
RD2/PSP2
RD3/PSP3
RD4/PSP4
RD5/PSP5
RD6/PSP6
RD7/PSP7
PIC16F877A
<TEXT>

```
#include p16f877a.inc          ; Include register definition file

;=================================================================
; VARIABLES
;=================================================================
    counter EQU 20H


    Bank0 macro
    BCF STATUS,RP0
    BCF STATUS,RP1
    ENDM

     Bank1 macro
    BSF STATUS,RP0
    BCF STATUS,RP1
    ENDM

;=================================================================
; RESET and INTERRUPT VECTORS
;=================================================================

    ; Reset Vector
RST   code  0x0
    goto  Start
```

```
PGM   code
Start
    Bank1
    MOVLW b'11111111'
    MOVWF TRISA ; Configuration of PORT A as input
    CLRF TRISC ; Configuration of PORT B as output (To show ADC convertion with LEDs)
    CLRF TRISD ; Configuration of PORT D as output (To show ADC convertion with LEDs)
    MOVLW b'10000000' ; A/D Result Format (Right justified)
    MOVWF ADCON1
    Bank0
    CLRF PORTB ; Setting PORTB to "0000000"
    MOVLW b'01000001' ; Selection of conversion clock (FOSC/8), analog channel (AN0), A/D conversion not in progress
    MOVWF ADCON0
DELAY_INIT
    MOVLW d'19'
    MOVWF counter
DELAY_LOOP
    DECFSZ counter, f
    GOTO DELAY_LOOP
    BSF ADCON0,2 ; Set A/D conversion in progress
ADC_CONVERT
    BTFSC ADCON0,2
    GOTO ADC_CONVERT
    Bank1
    MOVF ADRESL, 0 ; Move the result of ADRESL to working register
    Bank0
    MOVWF PORTC ; Move to PORTB what it is in the working register
    MOVF ADRESH, 0 ; Move the result of ADRESH to working register
    MOVWF PORTD ; Move to PORTD what it is in the working register
    GOTO DELAY_INIT
    END
```


Addr. Name
00h INDF
01h TMR0
02h PCL
03h STATUS
04h FSR
05h PORTA
06h PORTB
07h PORTC
08h PORTD
09h PORTE
0Ah PCLATH
0Bh INTCON
0Ch PIR1
0Dh PIR2
0Eh TMR1L
0Fh TMR1H
10h T1CON
11h TMR2
12h T2CON
13h SSPBUF
14h SSPCON
15h CCPR1L
16h CCPR1H
17h CCP1CON
18h RCSTA
19h TXREG
1Ah RCREG
1Bh CCPR2L
1Ch CCPR2H
1Dh CCP2CON
1Eh ADRESH
1Fh ADCON0
20h
General Purpose Registers
96 bytes
7Fh
Bank 0
Addr. Name
80h INDF
81h OPTION_REG
82h PCL
83h STATUS
84h FSR
85h TRISA
86h TRISB
87h TRISC
88h TRISD
89h TRISE
8Ah PCLATH
8Bh INTCON
8Ch PIE1
8Dh PIE2
8Eh PCON
8Fh OSCCON
90h OSCTUNE
91h SSPCON2
92h PR2
93h SSPADD
94h SSPSTAT
95h WPUB
96h IOCB
97h VRCON
98h TXSTA
99h SPBRG
9Ah SPBRGH
9Bh PWM1CON
9Ch ECCPAS
9Dh PSTRCON
9Eh ADRESL
9Fh ADCON1
A0h
General Purpose Registers
80 bytes
FFh
Bank 1
Addr. Name
100h INDF
101h TMR0
102h PCL
103h STATUS
104h FSR
105h WDTCON
106h PORTB
107h CM1CON0
108h CM2CON0
109h CM2CON1
10Ah PCLATH
10Bh INTCON
10Ch EEDAT
10Dh EEADR
10Eh EEDATH
10Fh EEADRH
110h
General Purpose Registers
96 bytes
17Fh
Bank 2
Addr. Name
180h INDF
181h OPTION_REG
182h PCL
183h STATUS
184h FSR
185h SRCON
186h TRISB
187h BAUDCTL
188h ANSEL
189h ANSELH
18Ah PCLATH
18Bh INTCON
18Ch EECON1
18Dh EECON2
18Eh Not Used
18Fh Not Used
190h
General Purpose Registers
96 bytes
1EFh
Bank 3


Thank
you!