Chapter 12

Data Converters

→Read Section 19 of the **Data Sheet for PIC18F46K20**

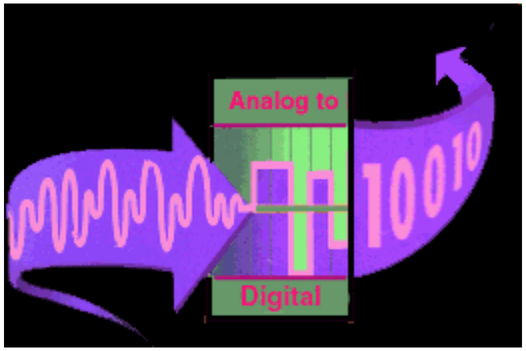
Updated: 4/19/15

Data Converters: Basic Concepts

- Analog signals are continuous, with infinite values in a given range.
- Digital signals have discrete values such as on/off or 0/1.
- Limitations of analog signals
  - Analog signals pick up noise as they are being amplified.
  - Analog signals are difficult to store.
  - Analog systems are more expensive in relation to digital systems.

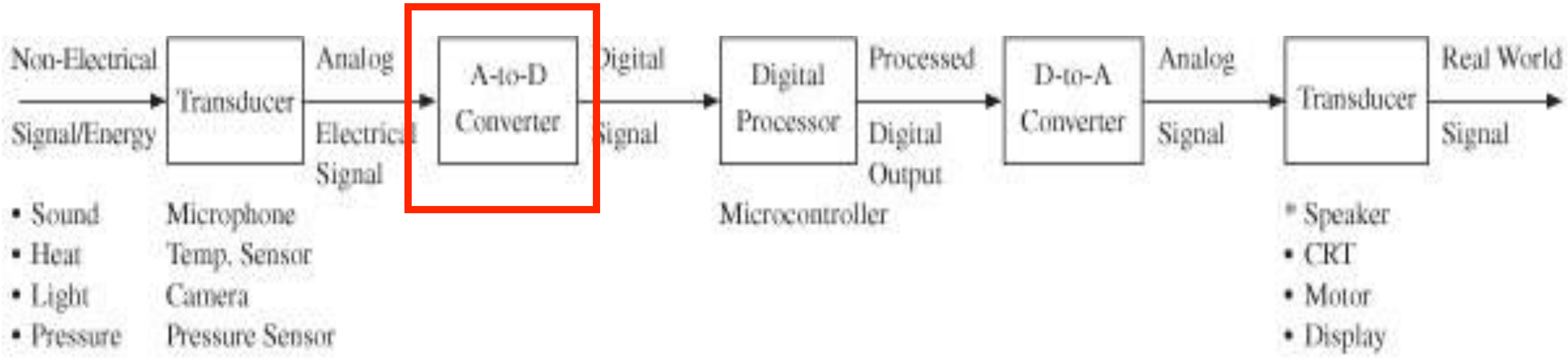
Data Converters: Basic Concepts

- Advantages of digital systems (signals)
  - Noise can be reduced by converting analog signals in 0s and 1s.
  - Binary signals of 0s/1s can be easily stored in memory.
  - Technology for fabricating digital systems has become so advanced that they can be produced at low cost.
- The major limitation of a digital system is how accurately it represents the analog signals after conversion.



Embedded System

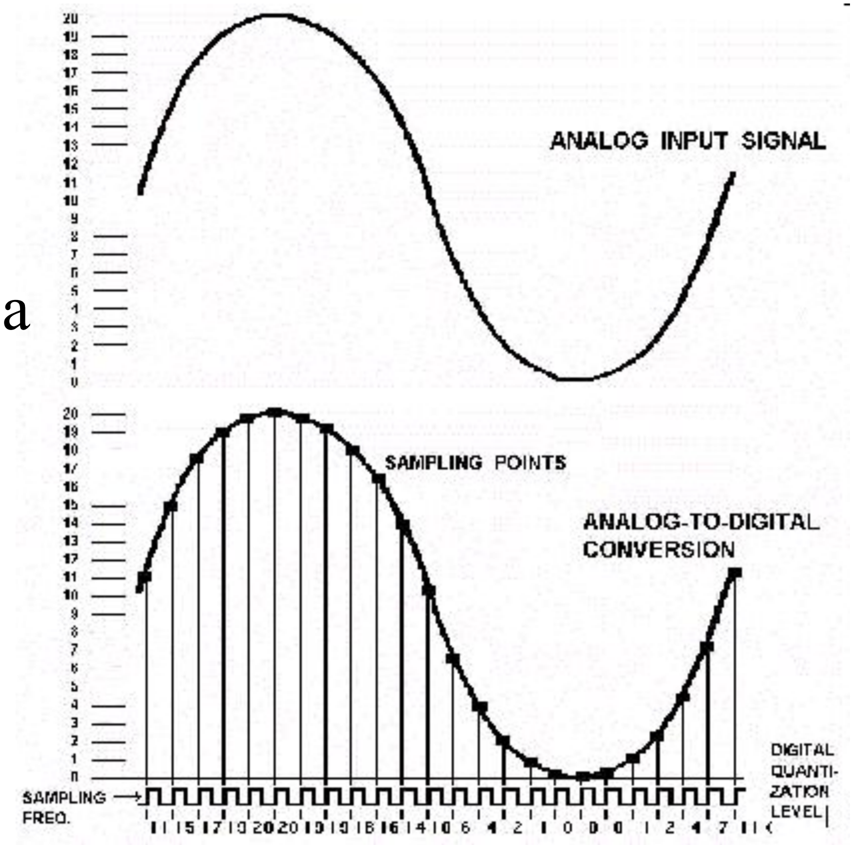
- A typical system that converts signals from analog to digital and back to analog includes:
  - A transducer that converts non-electrical signals into electrical signals
  - An A/D converter that converts analog signals into digital signals
  - A digital processor that processes digital data (signals)
  - A D/A converter that converts digital signals into equivalent analog signals
  - A transducer that converts electrical signals into real life non-electrical signals (sound, pressure, and video)



So, how does A/D Converter works?

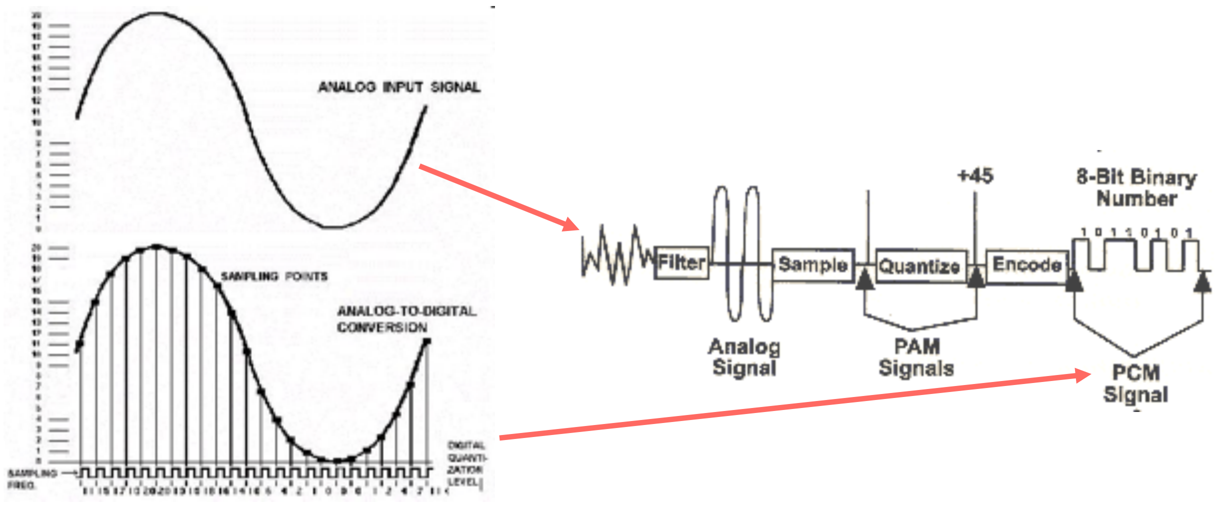
A/D Converter

- In order to change an analog signal to digital, the input analog signal is sampled at a high rate of speed.
- The amplitude at each of those sampled moments is converted into a number equivalent – this is called quantization.
- These numbers are simply the combinations of the 0s and 1s used in computer language – this called encoding.



http://www.cybercollege.com/tvp008.htm

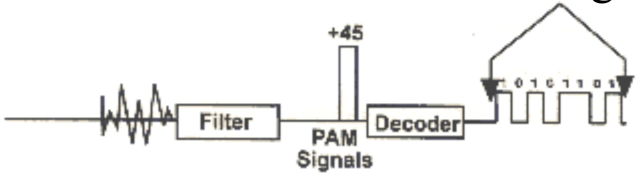
A/D Conversion – Pulse Code Modulation/Demodulation



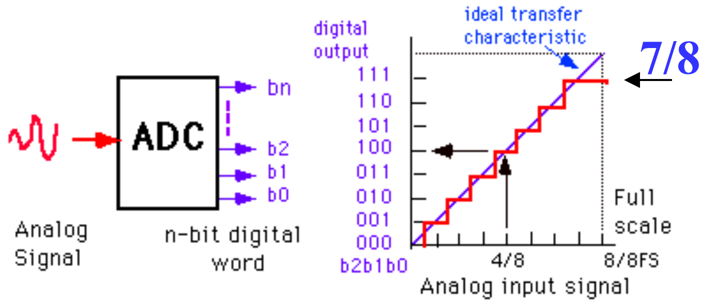
Modulation

Demodulation

PCM Signal



Analog-to-Digital



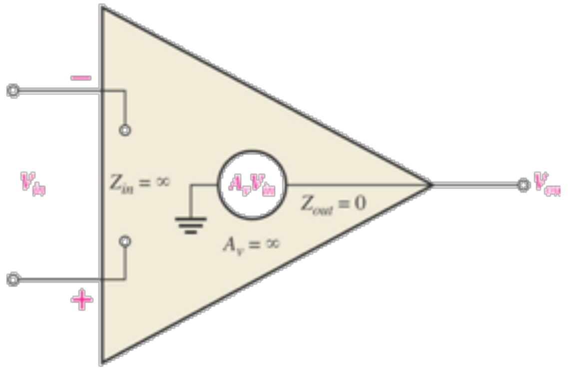
- A simple hypothetical A/D converter circuit with one analog input signal and three digital output lines with eight possible binary combinations: 000 to 111
  - Shows the graph of digital output for FS V analog input
- The following points can be summarized in the above process:
  - Maximum value this quantization process reaches is **7/8** V for a 1 V analog signal; includes 1/8 V an inherent error
  - 1/8 V (an inherent error) is also equal to the value of the Least Significant Bit (LSB) = 001.
  - Resolution of a converter is defined in terms of the number of discrete values it can produce; also expressed in the number of bits used for conversion or as **$1/2^n$** where n =number of bits
  - The value of the most significant bit (MSB) -100- is equal to ½ the voltage of the full-scale value of 1 V.
  - The value of the largest digital number 111 is equal to full-scale value minus the value of the LSB.
  - The quantization error can be reduced or the resolution can be improved by increasing the number of bits used for the conversion

A little Detour: Opamp Review

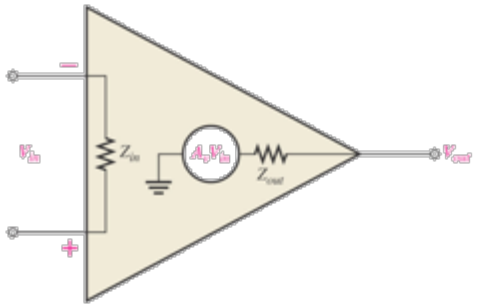
- http://www.engin.brown.edu/courses/en123/Lectures/DAconv.htm
- Some basics: http://www.seas.upenn.edu/~ese206/labs/adc206/adc206.html
- Check out the Applet on my web

Opamps

- Ideal opamps
  - Infinite BW
  - Infinite voltage gain
  - Infinite input impedance
  - Zero output impedance
- Practical opamps
  - Wide BW
  - Very high voltage gain
  - Very high input impedance
  - Very low output impedance



(a) Ideal op-amp representation

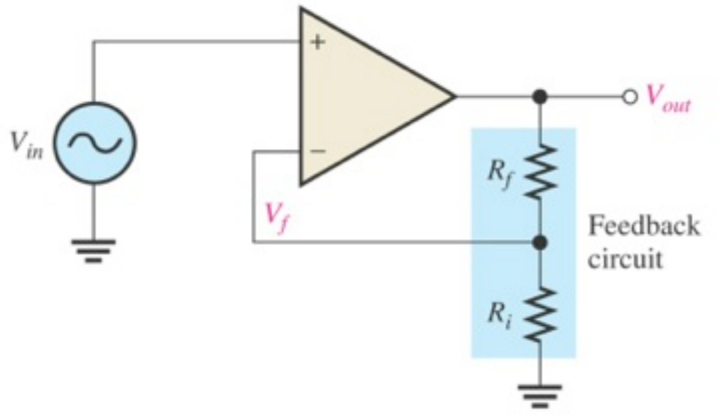


(b) Practical op-amp representation

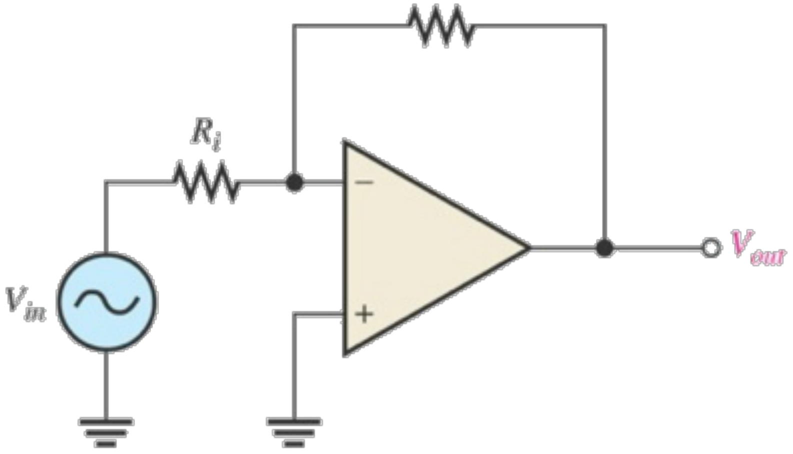
http://www.chem.uoa.gr/Applets/AppletOpAmps/Appl_OpAmps2.html

Closed Loop Frequency Response

- Non-inverting
  - Source is connected to the non-inverting input
  - Feedback is connected to the inverting input
  - If Rf and Ri are zero, then unity feedback used for buffering
  - Av = 1+Rf/Ri

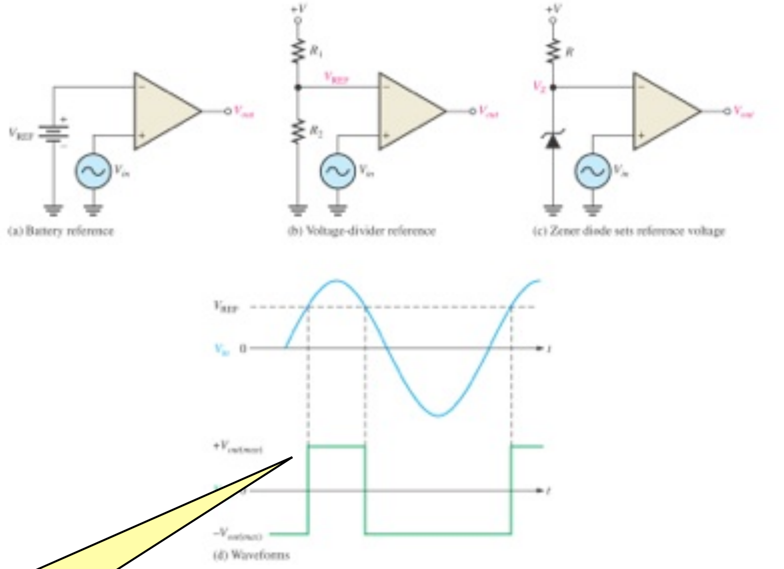


- Inverting
  - Feedback and source are connected to the inverting input
  - Av = - Rf/Ri



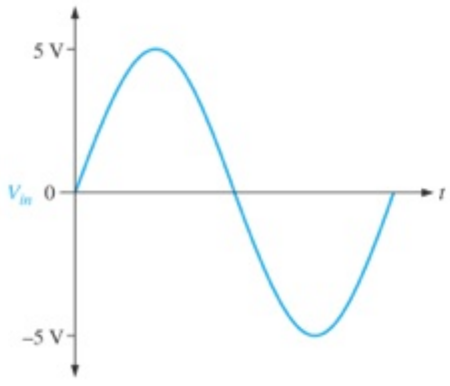
Comparators

- Determines which input is larger
- A small difference between inputs results maximum output voltage (high gain)
- Zero-level detection
- Non-zero-level detection

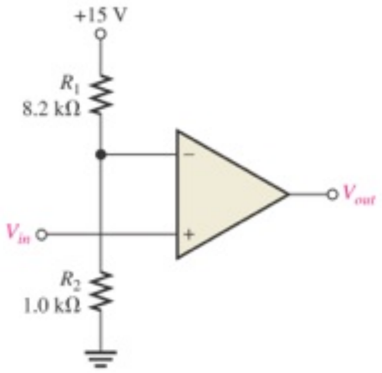


Max and minimum

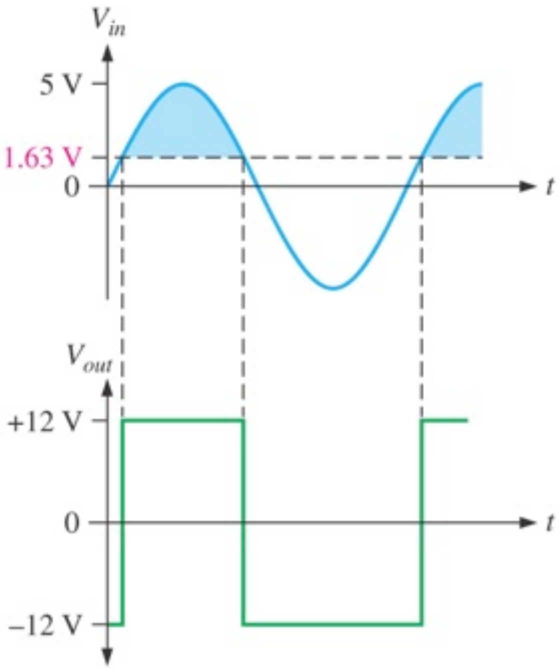
Example



(a)



(b)



Vref = Vin(max).R2/(R1+R2)=1.63 V

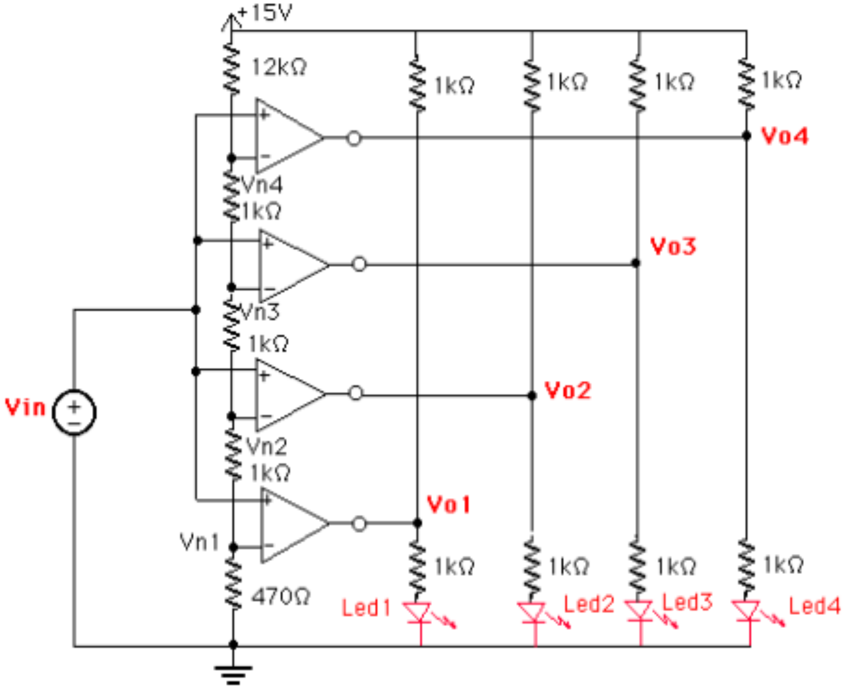
Back to A/D Converters….

A/D Conversion - Types

- Can be classified in four groups:
  - Integrator:
    - Charges a capacitor for a given amount of time using the analog signal.
    - It discharges back to zero with a known voltage and the counter provides the value of the unknown signal.
    - Provides slow conversion but low noise.
    - Often used in monitoring devices (e.g., voltmeters)
  - Flash: uses multiple comparators in parallel.
    - The known signal is connected to one side of the comparator and the analog signal to be converted to the other side of the comparator.
    - The output of the comparators provides the digital value.
    - This is a high-speed, high cost converter.

A/D Conversion

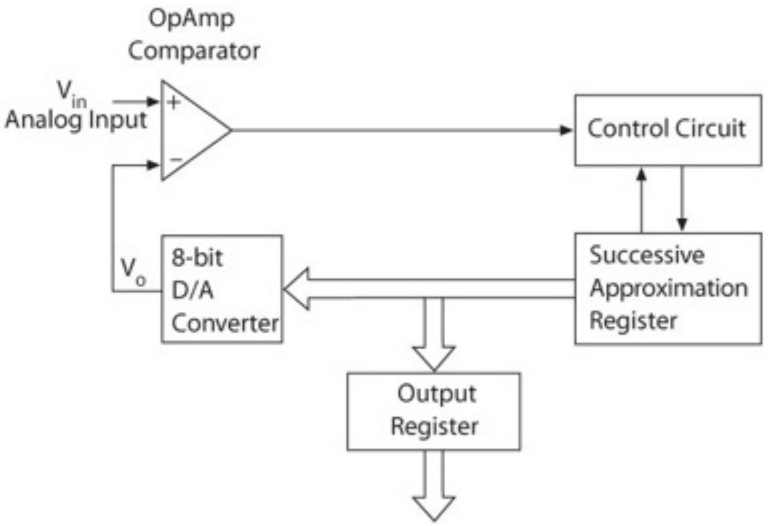
- Flash Converter
  - The circuit consists of 4 comparators whose inverting inputs are connected to a voltage divider.
  - A comparator is basically an operational amplifier used without feedback.
  - The outputs of the comparators correspond to a digital word.
  - When the input rises above Vn1 , the first comparator will switch to a high output voltage causing the LED to light up, indicating a (0001).
  - For larger input voltages the output of other comparators will switch high as well. For large input voltages (above Vn3) all comparators will be high corresponding to (1111) digital output.



A/D Conversion

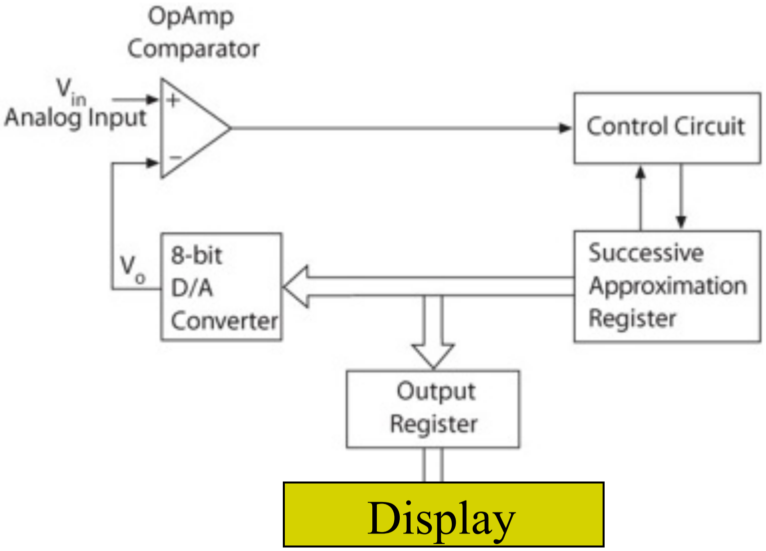
- Successive approximation: Includes a D/A (digital to analog) converter and a comparator. An internal analog signal is generated by turning on successive bits in the D/A converter.

- Counter: Similar to a successive approximation converter except that the internal analog signal is generated by a counter starting at zero and feeding it to the D/A converter.



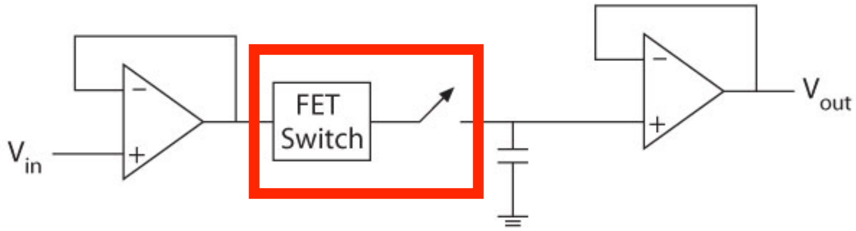
Successive Approximation A/D Converter Circuit

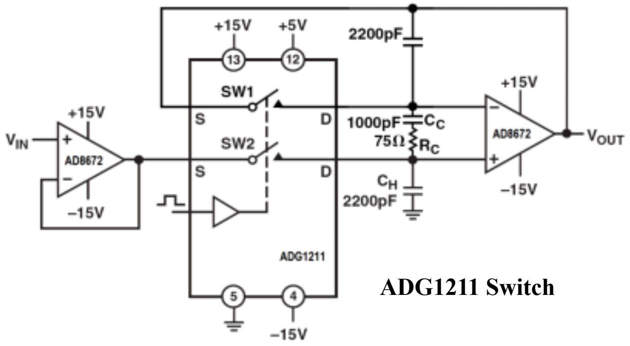
- The SAR (successive approximation register) begins by turning on the MSB Bit7.
- V_o of the D/A converter is compared with the analog input voltage V_{in} in the comparator.
- If analog voltage is less than the digital voltage, Bit7 is turned off and Bit6 is turned on.
- If analog voltage is greater than the digital voltage, Bit7 is kept on and Bit6 is turned on.
- The process of turning bit on/off is continued until Bit0.
- Now the 8-bit input to the D/A converter represents the digital equivalent of the analog signal V_{in}.



Bit 7 is set: b7=1
If Va < Vd → b7=0; b6=1
If Va > Vd → b7=1; b6=1
…..
If Va < Vd → b7=0; …b0=1
If Va > Vd → b7=1; … b0=1
Done

Sample and Hold Circuit





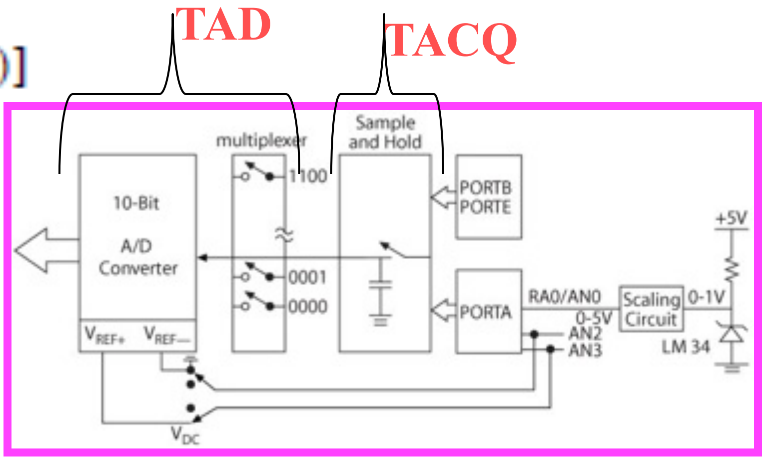
ADG1211 Switch

- If the input voltage to an A/D converter is variable, the digital output is likely to be unreliable and unstable. Therefore, the varying voltage source is connected to the ADC through a sample and hold circuit.
- Basic Operation:
  - When the switch is connected, it samples the input voltage.
  - When the switch is open, it holds the sampled voltage by charging the capacitor.
  - Acquisition time: time to charge the capacitor after the switch is open and settle the output.
  - Conversion time: total time needed from the start of a conversion (turning on the MSB in the SAR) until the end of the conversion (turning on/off Bit0 in the SAR) - TAD: conversion time per bit.

Calculating Required Acquisition Time (TACQ)

Tc=1.2usec; for temperature = 0-100°C (for simplicity)
and external impedance of 10kohm 3.0V VDD - Tamp = 5usec.
TAD = Acquisition time – we set this time on the device (depends on which clock we have Selected - Total conversion time (this is what we need)
TACQ = minimum required total acquisition time
(calculated based on temperature & device properties)

$$\begin{aligned} T_{ACQ} &= \text{Amplifier Settling Time} + \text{Hold Capacitor Charging Time} + \text{Temperature Coefficient} \\ &= T_{AMP} + T_C + T_{COFF} \\ &= 5\mu s + T_C + [(\text{Temperature} - 25°C)(0.05\mu s/°C)] \end{aligned}$$


TAD
TACQ

Note: TAD < TACQ
e.g., TACQ=12TAD

Refer to Datasheet

Example

- Assume clock is 32MHz; at 60 deg. C. What should be the total conversion time?
  - Required Acquisition Time

$$\begin{aligned} T_{ACQ} &= \text{Amplifier Settling Time} + \text{Hold Capacitor Charging Time} + \text{Temperature Coefficient} \\ &= T_{AMP} + T_{C} + T_{COFF} \\ &= 5\mu s + T_{C} + [(\text{Temperature} - 25^\circ C)(0.05\mu s/^\circ C)] \end{aligned}$$

More Later Slides….

A/D Examples

- Example 1
  - Assumes the input analog voltage is changing between 0-5 V.
  - Using a 3-bit A/D converter draw the output as the input signal ramps from 0 to 5V.
  - Calculate the resolution.
  - What is the maximum possible voltage out? (this is called the full-scale output)
  - If the output is 011, what is the input?
- Example 2
  - Assumes the input analog voltage is changing between -5 to 5 V; using a 10-bit A/D converter.
  - Calculate the number of quantization levels.
  - Calculate the voltage resolution.

A/D Examples

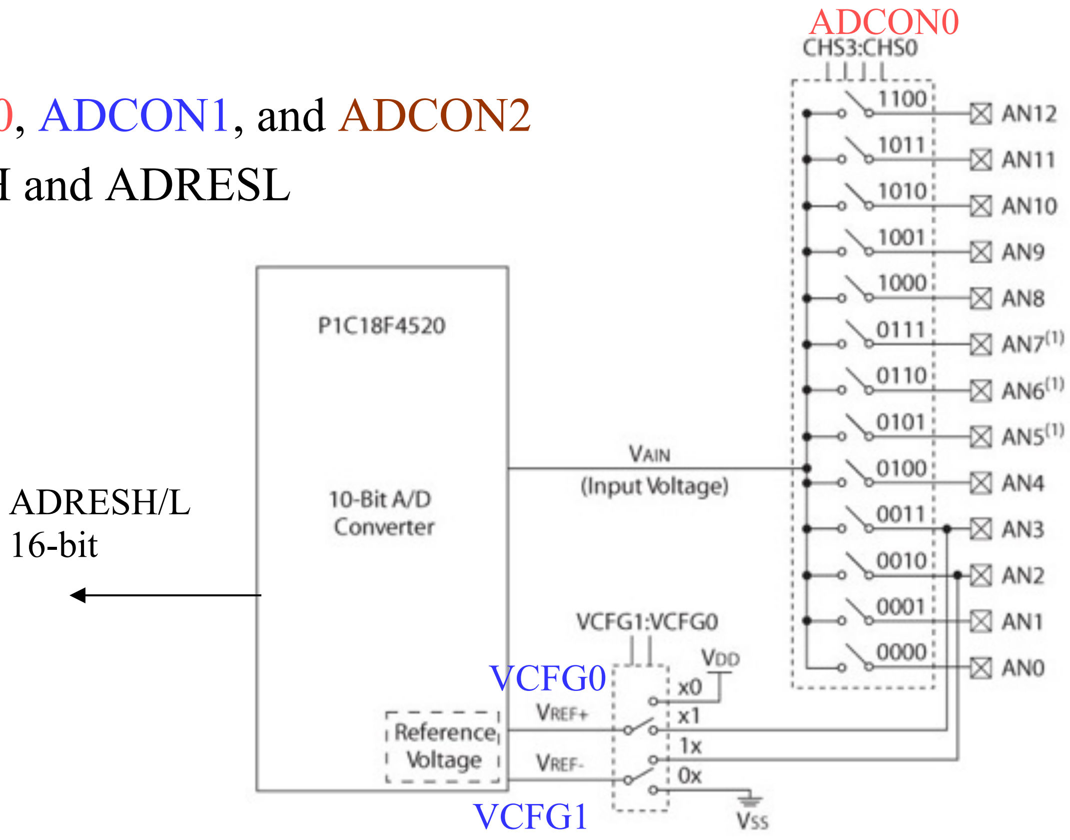
- Example 1
  - Assumes the input analog voltage is changing between 0-5 V.
  - Using a 3-bit A/D converter draw the output as the input signal ramps from 0 to 5V.
  - Calculate the resolution in volts. 5/ 2^3V
  - What is the maximum possible voltage out? (this is called the full-scale output) (5 – Resolution)
  - If the output is 011, what is the input? 3x5/8Volt
- Example 2
  - Assumes the input analog voltage is changing between -5 to 5 V; using a 10-bit A/D converter.
  - Calculate the number of quantization levels. 2^10
  - Calculate the voltage resolution. 5-(-5)/1024=9.76 mV

PIC18F4520 Analog-to-Digital (A/D) Converter Module (1 of 3)

- The PIC184520 microcontroller includes:
  - 10-bit A/D converter
  - 13 channels AN0 – AN12
  - Three control registers
    - ADCON0, ADCON1, and ADCON2

PIC18F4520 Analog-to-Digital (A/D) Converter Module (2 of 3)

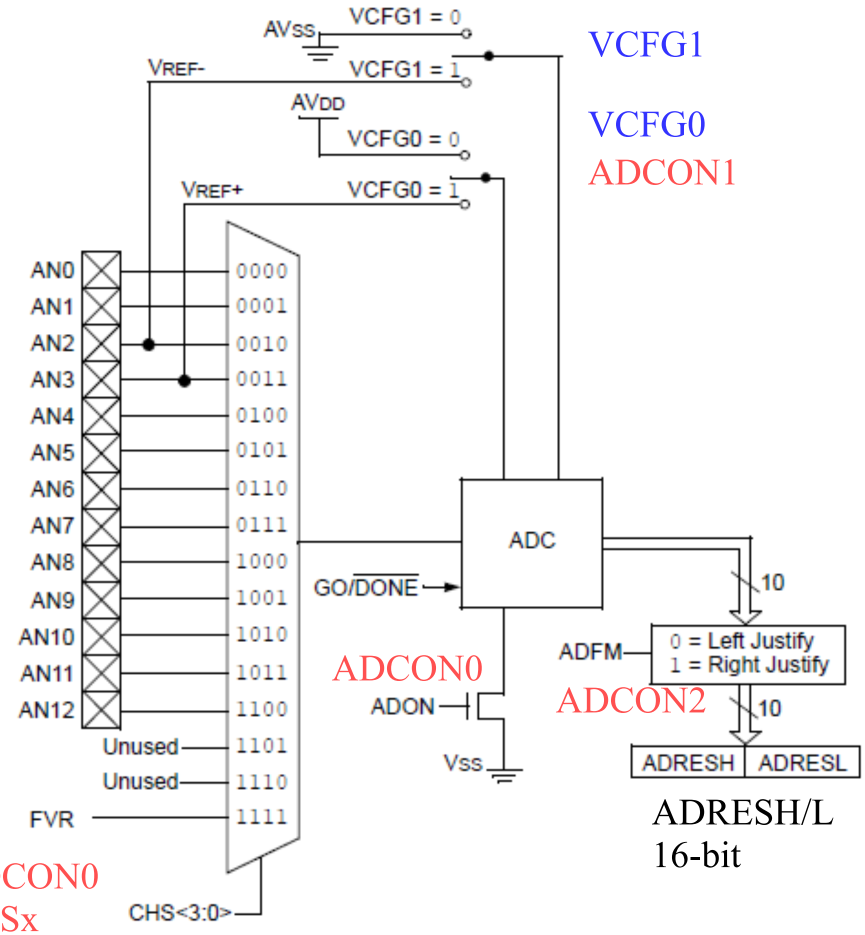
- ADCON0, ADCON1, and ADCON2
- ADRESH and ADRESL



Note 1: Channels AN5 through AN7 are not available on 28-pin devices.
2: I/O pins have diode protection to VDD and VSS.

PIC18F2XK20/4XK20 Analog-to-Digital (A/D) Converter Module (2 of 3)

- ADCON0, ADCON1, and ADCON2
- ADRESH and ADRESL



VCFG1

VCFG0

ADCON1

ADCON0

ADCON2

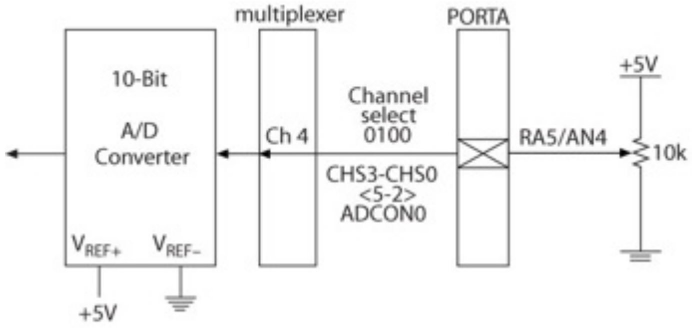
ADRESH/L
16-bit

1.2V Fixed Voltage Reference (FVR)
Register: **CM2CON1**

ADCON0
CHSx

PIC18F4520 Analog-to-Digital (A/D) Converter Module (3 of 3)

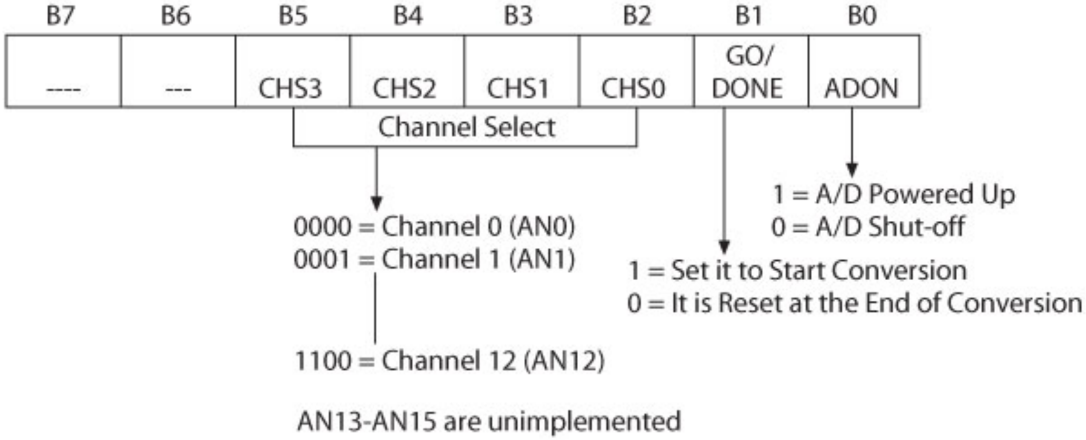
- Three control registers are used to:
  - Set up the I/O pins for analog signals from ports A, B, and E that are used as inputs for A/D conversion. RA5
  - Select a channel: AN4
  - Set up pins RA2 and RA3 to connect external V_{REF} + and V_{REF} - if specified in the control register ADCON1.
  - Select an oscillator frequency divider through the control register ADCON2.
  - Select an acquisition time through the control register ADCON2.



If the input is 0-1V→ Vin=[0-1]:
Option1: Vref+ & Vref- → 1V & GND
Option 2: Shift Vin to Vin' = Vin=[0-Vcc] and then Vref+ & Vref- → Vcc & GND

A/D Control Register0 (ADCON0)

- Primary function of the ADCON0 register:
    - Select a channel for input analog signal
    - Start a conversion
    - Indicate the end of the conversion
- Bit1 is set to start the conversion, and at the end of the conversion this bit is reset.



B7	B6	B5	B4	B3	B2	B1	B0
----	---	CHS3	CHS2	CHS1	CHS0	GO/ DONE	ADON

Channel Select

0000 = Channel 0 (AN0)
0001 = Channel 1 (AN1)
1100 = Channel 12 (AN12)

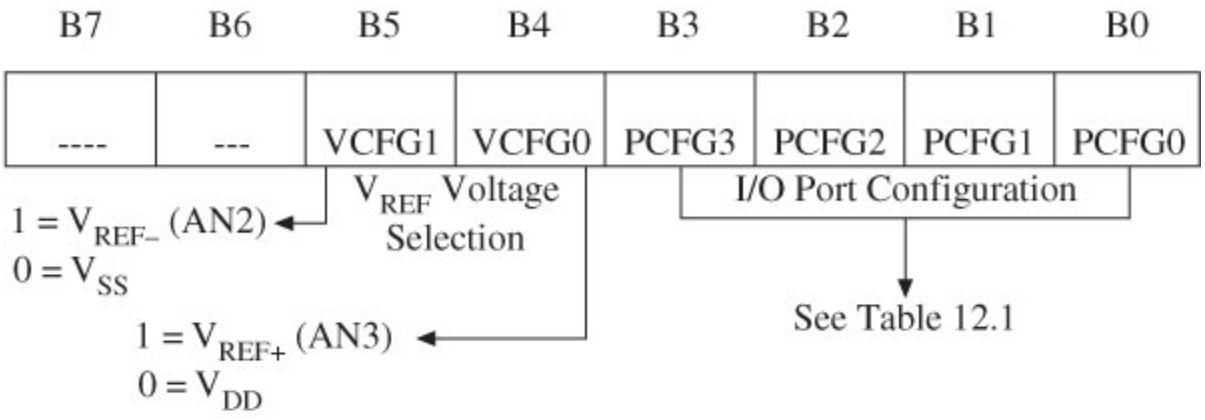
AN13-AN15 are unimplemented

1 = Set it to Start Conversion
0 = It is Reset at the End of Conversion

1 = A/D Powered Up
0 = A/D Shut-off

A to D Control Register1 (ADCON1)

- ADCON1 is primarily used to set up the I/O pins either for analog signal or for digital signals (see Table 12.2) and select V_{REF} voltages (see Table 12.1).



B7	B6	B5	B4	B3	B2	B1	B0
----	---	VCFG1	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0

V_{REF} Voltage Selection

$1 = V_{REF-}$ (AN2)
$0 = V_{SS}$

$1 = V_{REF+}$ (AN3)
$0 = V_{DD}$

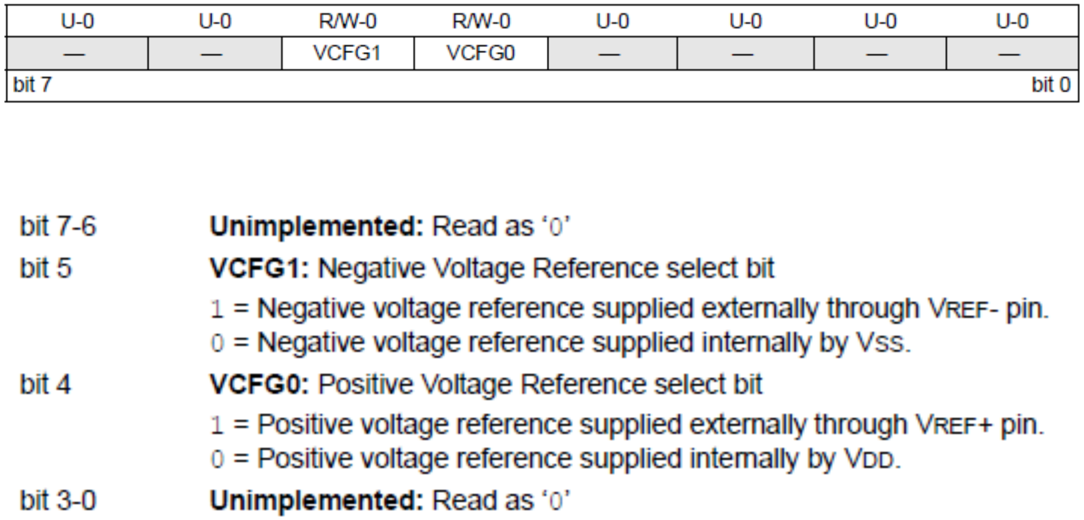
I/O Port Configuration

See Table 12.1

PIC18F2XK20/4XK20

A to D Control Register1 (ADCON1)

- ADCON1 is primarily used to set up the I/O pins either for analog signal or for digital signals



U-0	U-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	—	VCFG1	VCFG0	—	—	—	—
bit 7							bit 0

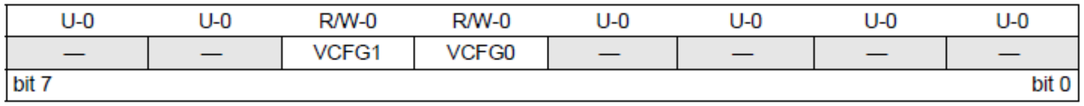
bit 7-6	**Unimplemented:** Read as ‘0’
bit 5	**VCFG1:** Negative Voltage Reference select bit 1 = Negative voltage reference supplied externally through VREF- pin. 0 = Negative voltage reference supplied internally by VSS.
bit 4	**VCFG0:** Positive Voltage Reference select bit 1 = Positive voltage reference supplied externally through VREF+ pin. 0 = Positive voltage reference supplied internally by VDD.
bit 3-0	**Unimplemented:** Read as ‘0’

Refer to ANSEL and ANSELH
Registers to setup inputs as analog ports

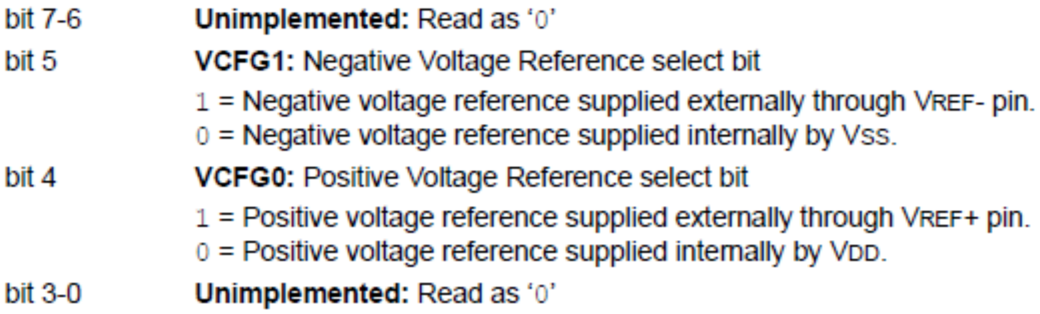
PIC18F2XK20/4XK20

A to D Control Register1 (ADCON1)

- ADCON1 is primarily used to set up the I/O pins either for analog signal or for digital signals



U-0	U-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	—	VCFG1	VCFG0	—	—	—	—
bit 7							bit 0

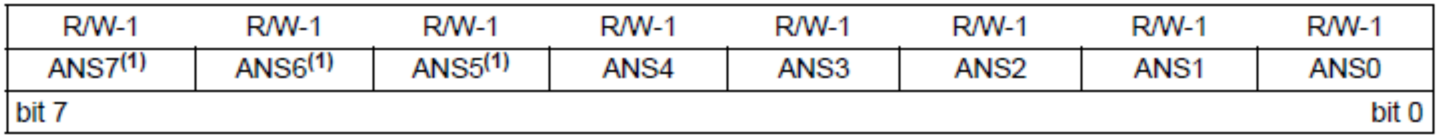


bit 7-6	**Unimplemented:** Read as '0'
bit 5	**VCFG1:** Negative Voltage Reference select bit 1 = Negative voltage reference supplied externally through VREF- pin. 0 = Negative voltage reference supplied internally by VSS.
bit 4	**VCFG0:** Positive Voltage Reference select bit 1 = Positive voltage reference supplied externally through VREF+ pin. 0 = Positive voltage reference supplied internally by VDD.
bit 3-0	**Unimplemented:** Read as '0'

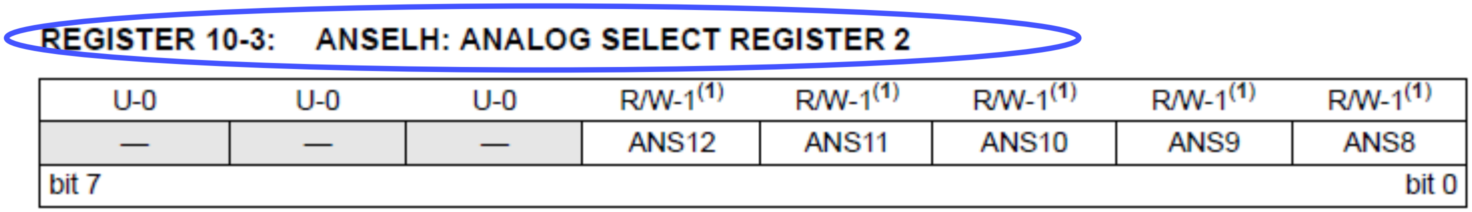
Refer to ANSEL and ANSELH
Registers to setup inputs as analog ports

Selecting the Analog Port

ANSEL



R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
ANS7[(1)]	ANS6[(1)]	ANS5[(1)]	ANS4	ANS3	ANS2	ANS1	ANS0
bit 7							bit 0



REGISTER 10-3: ANSELH: ANALOG SELECT REGISTER 2

U-0	U-0	U-0	R/W-1[(1)]	R/W-1[(1)]	R/W-1[(1)]	R/W-1[(1)]	R/W-1[(1)]
—	—	—	ANS12	ANS11	ANS10	ANS9	ANS8
bit 7							bit 0

PIC18F2XK20 devices:

RE2
RE1
RE0
RA5
RS3
RA2
RA1
RA0 (bit 0)

RB6
RB4
RB1
RB3
RB0 (bit 0)

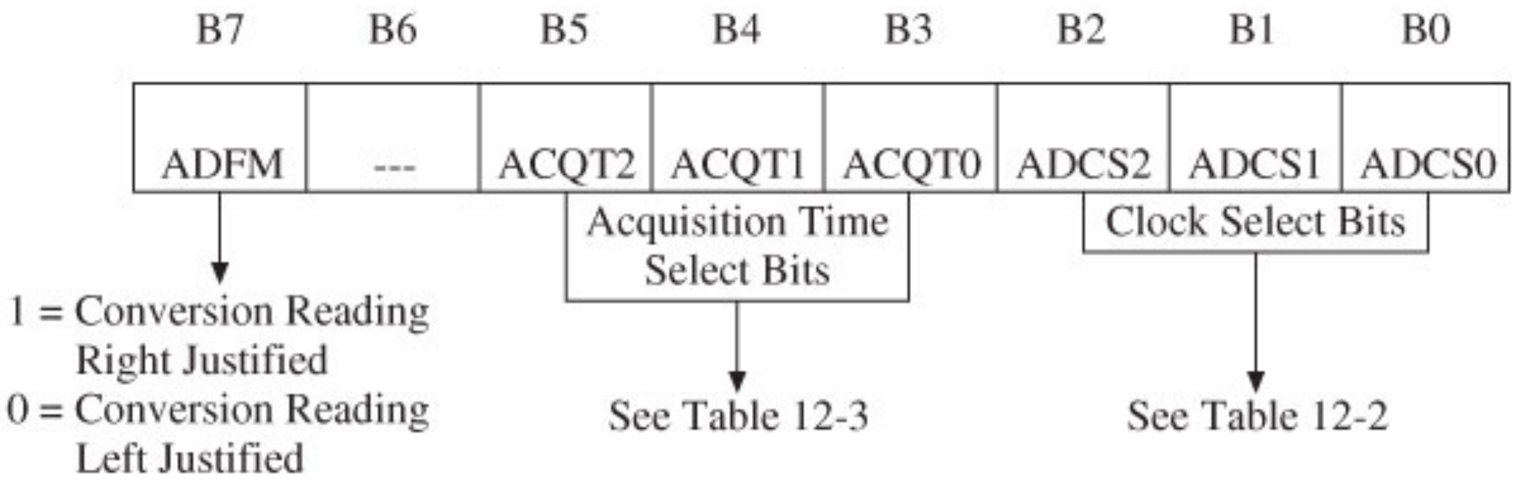
A to D Control Register2 (ADCON2)

(1 of 2)

- Used to:
  - Select an acquisition time and clock frequency
  - Right or left justify output reading
- The output reading, after a conversion, is stored in the 16-bit register ADRESH and ADRESL. However, this is a 10-bit A/D converter leaving six bit positions unused.
- Bit7 ADFM enables the user either to right justify or left justify the 16-bit reading leaving the unused positions as 0s.

A to D Control Register2 (ADCON2)

(2 of 2)



B7	B6	B5	B4	B3	B2	B1	B0
ADFM	---	ACQT2	ACQT1	ACQT0	ADCS2	ADCS1	ADCS0

Acquisition Time Select Bits

See Table 12-3

Clock Select Bits

See Table 12-2

1 = Conversion Reading
Right Justified
0 = Conversion Reading
Left Justified

A to D Control Register2 (ADCON2)

(2 of 2)

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	—	ACQT2	ACQT1	ACQT0	ADCS2	ADCS1	ADCS0
bit 7							bit 0

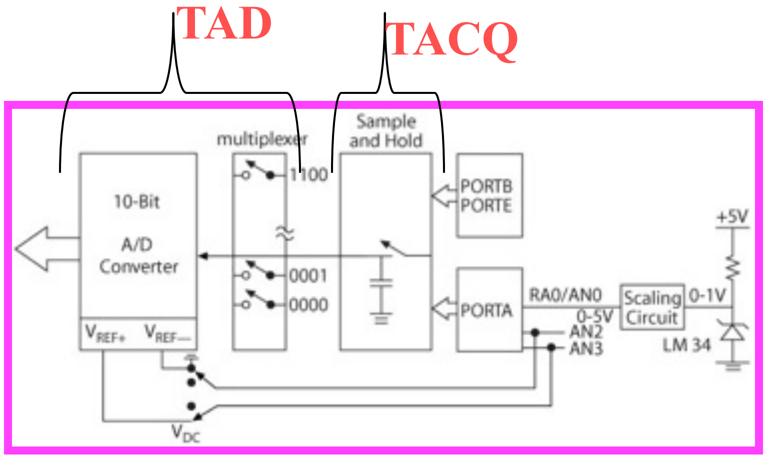
bit 7 **ADFM:** A/D Conversion Result Format Select bit

1 = Right justified
0 = Left justified

bit 6 **Unimplemented:** Read as '0'

bit 5-3 **ACQT<2:0>:** A/D Acquisition time select bits. Acquisition time is the duration that the A/D charge holding capacitor remains connected to A/D channel from the instant the GO/$\overline{\text{DONE}}$ bit is set until conversions begins.

000 = 0(1)
001 = 2 TAD
010 = 4 TAD
011 = 6 TAD
100 = 8 TAD
101 = 12 TAD
110 = 16 TAD
111 = 20 TAD

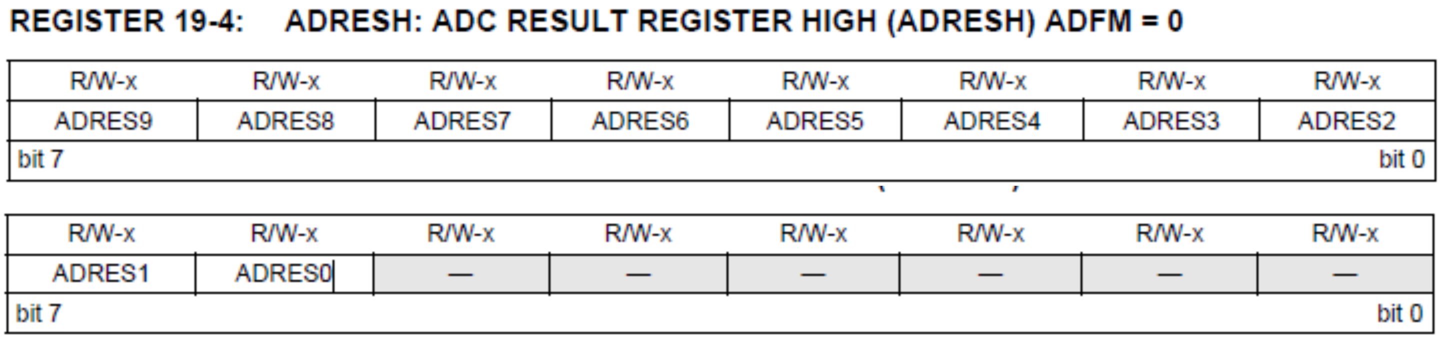


bit 2-0 **ADCS<2:0>:** A/D Conversion Clock Select bits

000 = Fosc/2
001 = Fosc/8
010 = Fosc/32
011 = FRC(1) (clock derived from a dedicated internal oscillator = 600 kHz nominal)
100 = Fosc/4
101 = Fosc/16
110 = Fosc/64
111 = FRC(1) (clock derived from a dedicated internal oscillator = 600 kHz nominal)

Note this is in Hz

Reading the OUTPUT Results



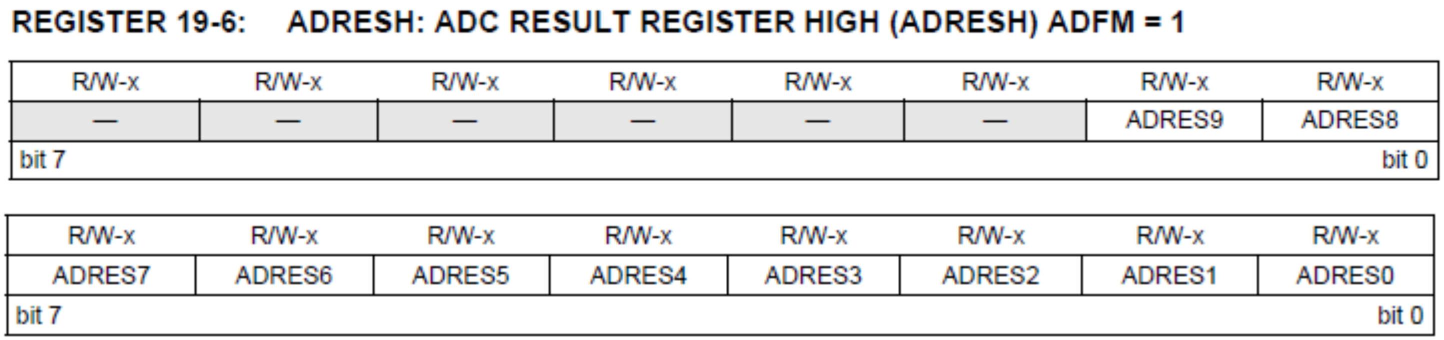
REGISTER 19-4: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 0

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
ADRES9	ADRES8	ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2
bit 7							bit 0

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
ADRES1	ADRES0	—	—	—	—	—	—
bit 7							bit 0

Left Justification

Right Justification



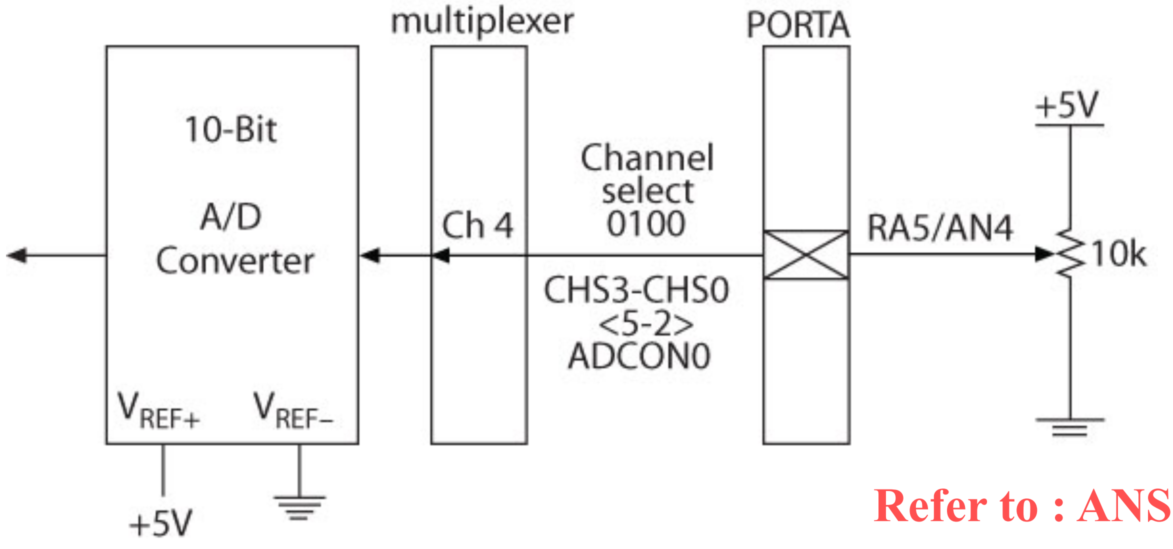
REGISTER 19-6: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 1

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	—	—	—	ADRES9	ADRES8
bit 7							bit 0

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2	ADRES1	ADRES0
bit 7							bit 0

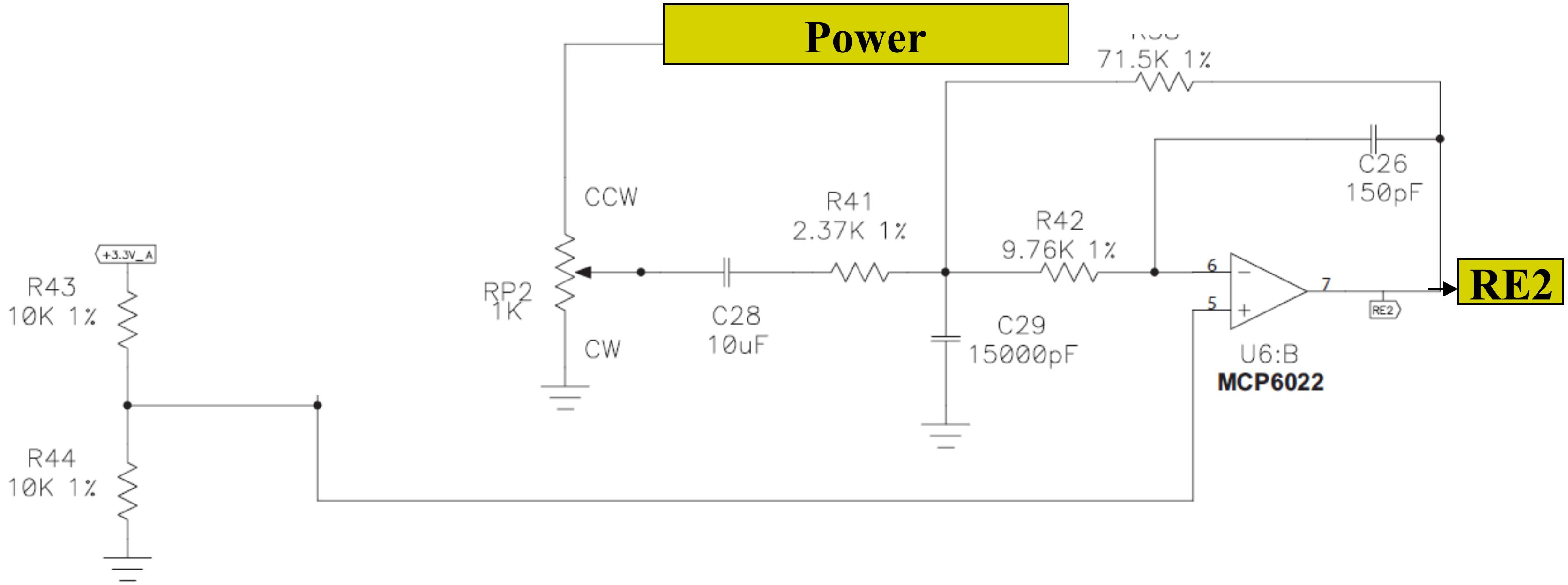
Example 12.3

- Interfacing a 10 k Pot



Refer to : ANSEL

Board Connection



Example:

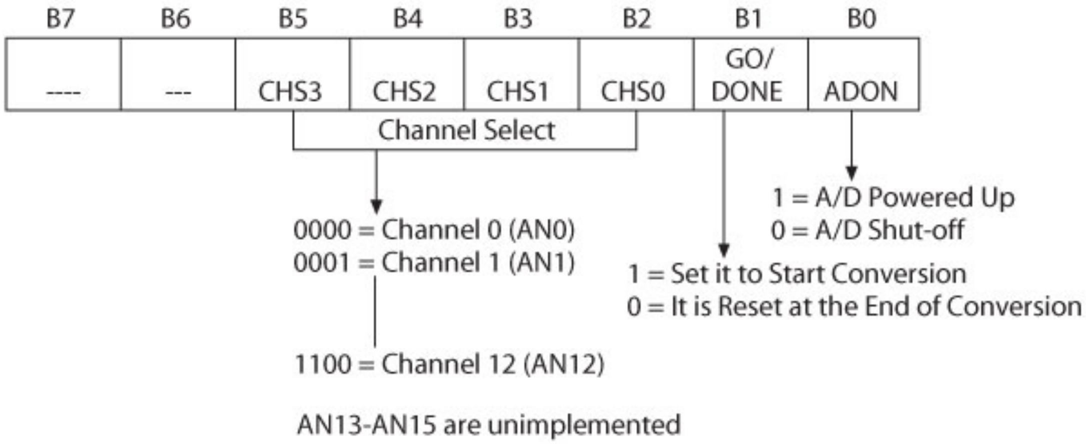
- What are the right questions?
  - What is the input connected to?
  - Which channel is connected to the A/D
  - Using external or internal clock
  - What is the Vref?
  - What is the minimum sampling time?
  - What is the acquisition time?

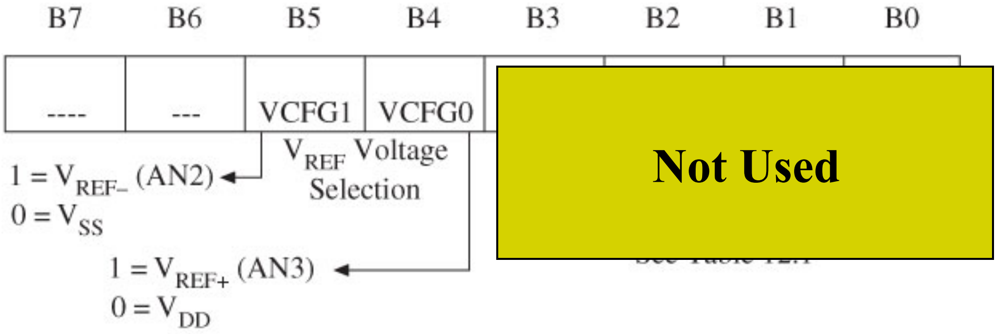
Example

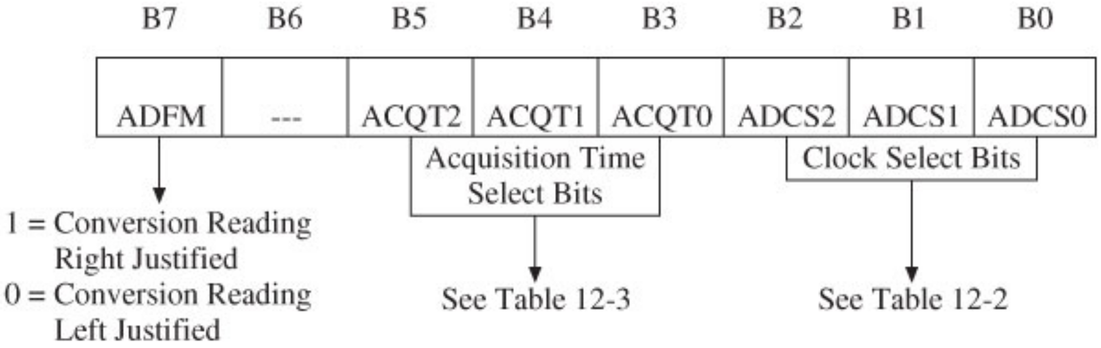
PIC18F2XK20/4XK20

- Assumptions
  - Use RE0 on the demo board.
  - Use external oscillator
  - Assuming conversion time (TAD) is 4 usec, what is the clock frequency requirement (ADCON2)
  - Assume acquisition time is 48 usec. What will be the acquisition time setting?
  - Write the program
- Set up the following registers properly:
  - ADCON0, ADCON1, ADCON2.

PIC18F2XK20/4XK20 Example







Basic calculations:

Fosc = 4MHz

TAD = 4usec = 1/(Fosc/x)→ x=16, hence, select Fosc/16 as clock source

Taqu-time = 48usec = y. Ttotal_conv_time → y=12, hence select 12.TAD

Setting:

ADCON0 = 00 000 01

ADCON1 = 00 00 00 11

ADCON2 = 10 101 101

Interfacing a Temperature Sensor
(1 of 7)

- Temperature sensor
  - Transducer that converts temperature into an analog electrical signal
  - Many are available as integrated circuits, and their outputs (voltage or current) are, in general, linearly proportional to the temperature
  - However, output voltage ranges of these transducers may not be ideally suited to reference voltages of A/D converters
  - Therefore, it is necessary to scale the output of a transducer to range of the reference voltages of an A/D converter
  - Scaling may require amplification or shifting of voltages at a different level

Interfacing a Temperature Sensor

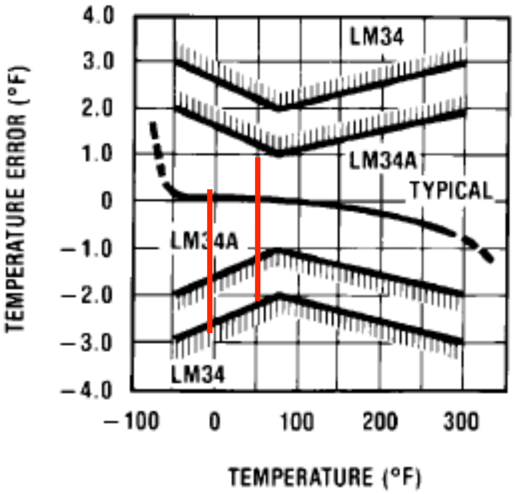
(2 of 7)

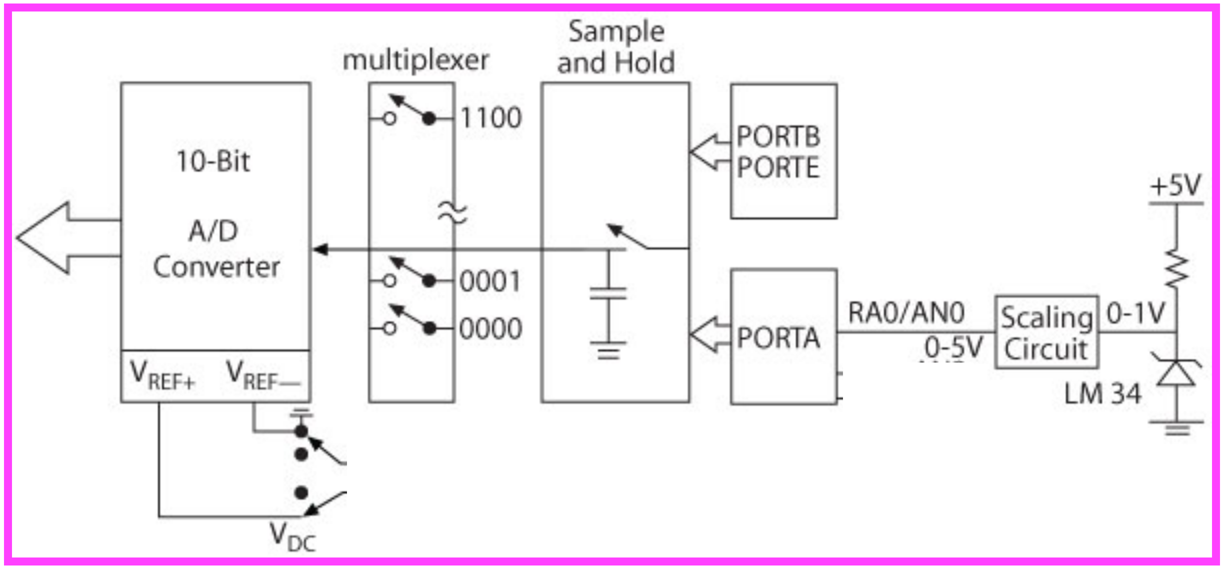
- Temperature Sensor
  - Interface the National Semiconductor LM34 temperature sensor to channel 0 (AN0) of the A/D converter module as shown in Figure 12.11.
  - Assume the output voltage of LM34 for the temperature ranges from 0°F to 100°F is properly scaled to 0 to +5 V.
  - Write instructions to start a conversion, read the digital reading at the end of the conversion, calculate the equivalent temperature reading in degrees Fahrenheit, convert it into BCD, and store the reading in ASCII code to the accuracy of one decimal point.
    - The expected range of temperatures is 0°F to 99.9°F.

Interfacing a Temperature Sensor

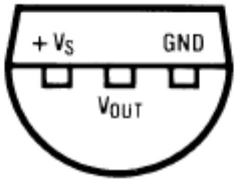
(3 of 7)

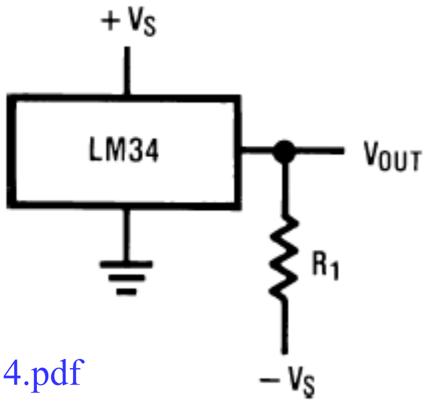
Accuracy vs. Temperature (Guaranteed)





TO-92 Plastic Package





CHOOSE $R_1 = (-V_S)/50\ \mu A$

V_{OUT} = +3,000 mV AT +300°F

= +750 mV AT +75°F

= −500 mV AT −50°F

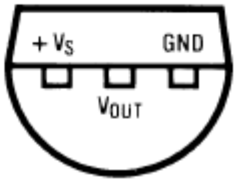
http://users.ipfw.edu/broberg/documents/LM34.pdf

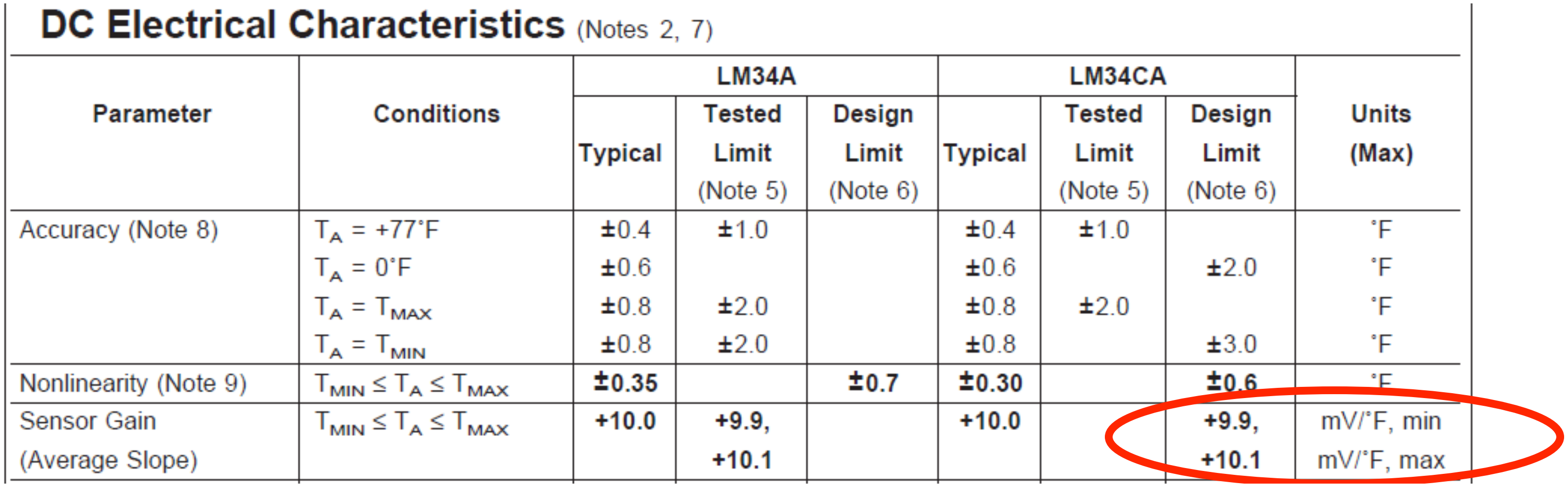
Interfacing a Temperature Sensor

(4 of 7)

- Hardware
  - Temperature transducer LM34
    - Three-terminal integrated circuit device that can operate in the +5 V to +30 V power supply range
    - Outputs 10 mV/°F linearly
  - For the temperature range from 0°F to +99.9°F; the output voltage range is 0 to 1 V (rounded off to 100°F).

TO-92
Plastic Package





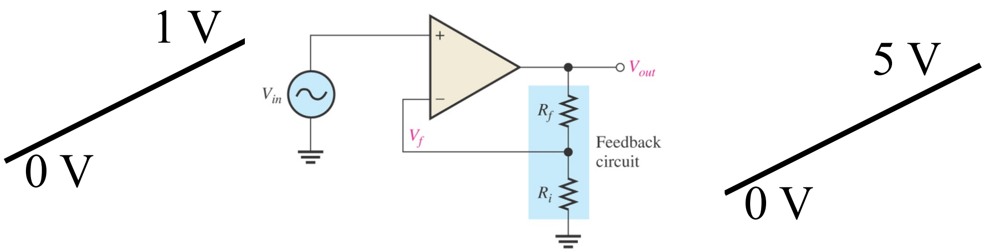
DC Electrical Characteristics (Notes 2, 7)

Parameter	Conditions	LM34A			LM34CA			Units (Max)
		Typical	Tested Limit (Note 5)	Design Limit (Note 6)	Typical	Tested Limit (Note 5)	Design Limit (Note 6)	
Accuracy (Note 8)	$T_A = +77°F$	±0.4	±1.0		±0.4	±1.0		°F
	$T_A = 0°F$	±0.6			±0.6		±2.0	°F
	$T_A = T_{MAX}$	±0.8	±2.0		±0.8	±2.0		°F
	$T_A = T_{MIN}$	±0.8	±2.0		±0.8		±3.0	°F
Nonlinearity (Note 9)	$T_{MIN} \le T_A \le T_{MAX}$	**±0.35**		**±0.7**	**±0.30**		**±0.6**	°F
Sensor Gain (Average Slope)	$T_{MIN} \le T_A \le T_{MAX}$	**+10.0**	**+9.9,** **+10.1**		**+10.0**		**+9.9,** **+10.1**	mV/°F, min mV/°F, max

Interfacing a Temperature Sensor

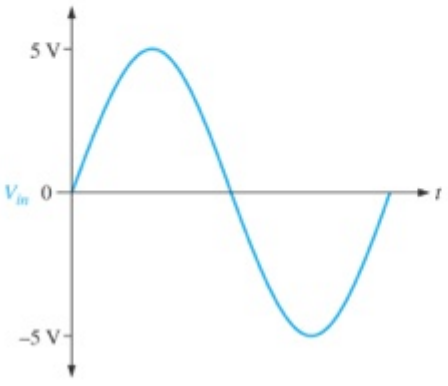
(5 of 7)

- Scaling circuit
  - To get the full dynamic range of the A/D conversion for the output voltage range 0 to 1V of LM34:
    - We can connect $+V_{REF}$ to +1 V or
    - Scale the output voltage +1V to the voltage of the power supply +5 V
  - This scaling enables us to connect PIC18 power supply V_{DD} as voltage reference $+V_{REF}$ and ground V_{ss} as $-V_{REF}$.

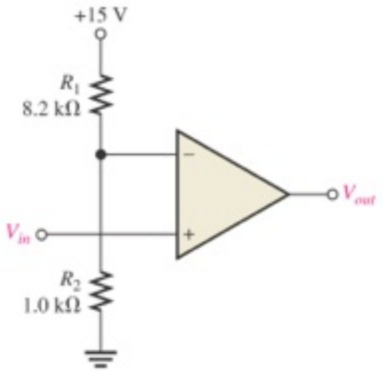


Non-inverting opamp:
Av = 1 + Rf/Ri

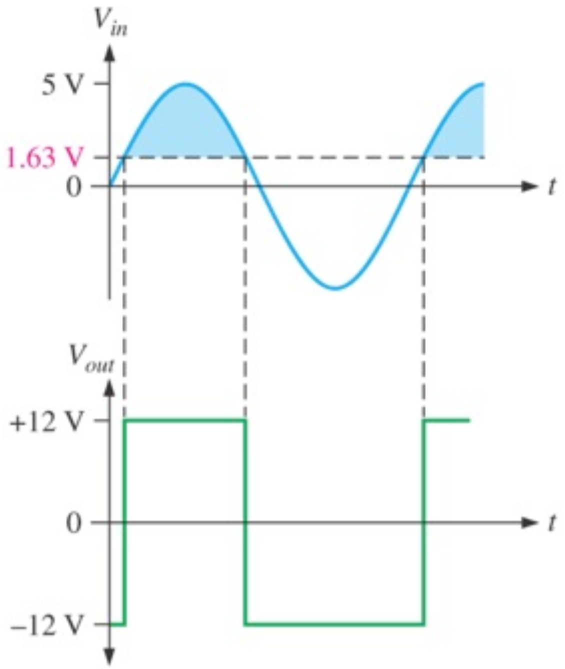
Remember ….



(a)



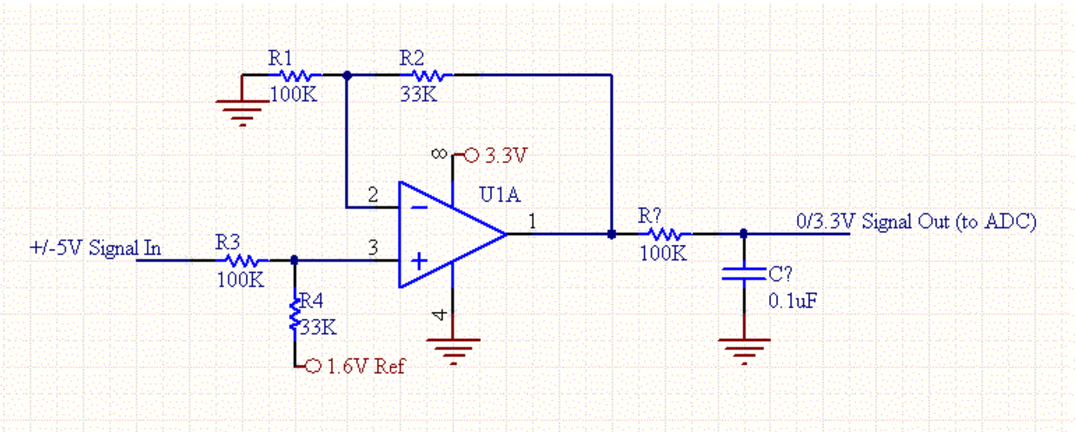
(b)



Vref = Vin(max).R2/(R1+R2)=1.63 V

Non-Inverting Voltage Level Shifter

- **Equations:**
  - A = (R4/R1) x (R1+R2)/(R3+R4)
  - If R1= R3, and R2=R4, then A= (R4/R1)
  - We want to convert a 10Vpp signal to a 3.3V signal so the gain should be 1/3. We can choose R4 to be 33K and R1 to be 100K.
  - We need to choose the positive offset such that the signal is centered at 1.6V.
  - The gain of the offset voltage is:
    - offset= (R2+R1)/R1 x R3/(R3+R4) = R3/R1.
    - For the previous resistor values, the gain is 1 since R3=R1, and so we use an offset voltage of 1.6V.



Interfacing a Temperature Sensor

(6 of 7)

- Temperature calculations
  - A/D converter has 10-bit resolution
  - For temperature range 0°F to +100°F, the digital output should be divided into 1024 steps (0 to $3FF_H$).
  - Therefore, the digital value per degrees Fahrenheit is 10.23 (1023/100 = 10.23_{10}). Each step → 10.23
  - To obtain temperature reading from a digital reading of the A/D converter, the digital reading must be divided by the factor of 10.23.
    - A digital value of 10.23 → 1 degree

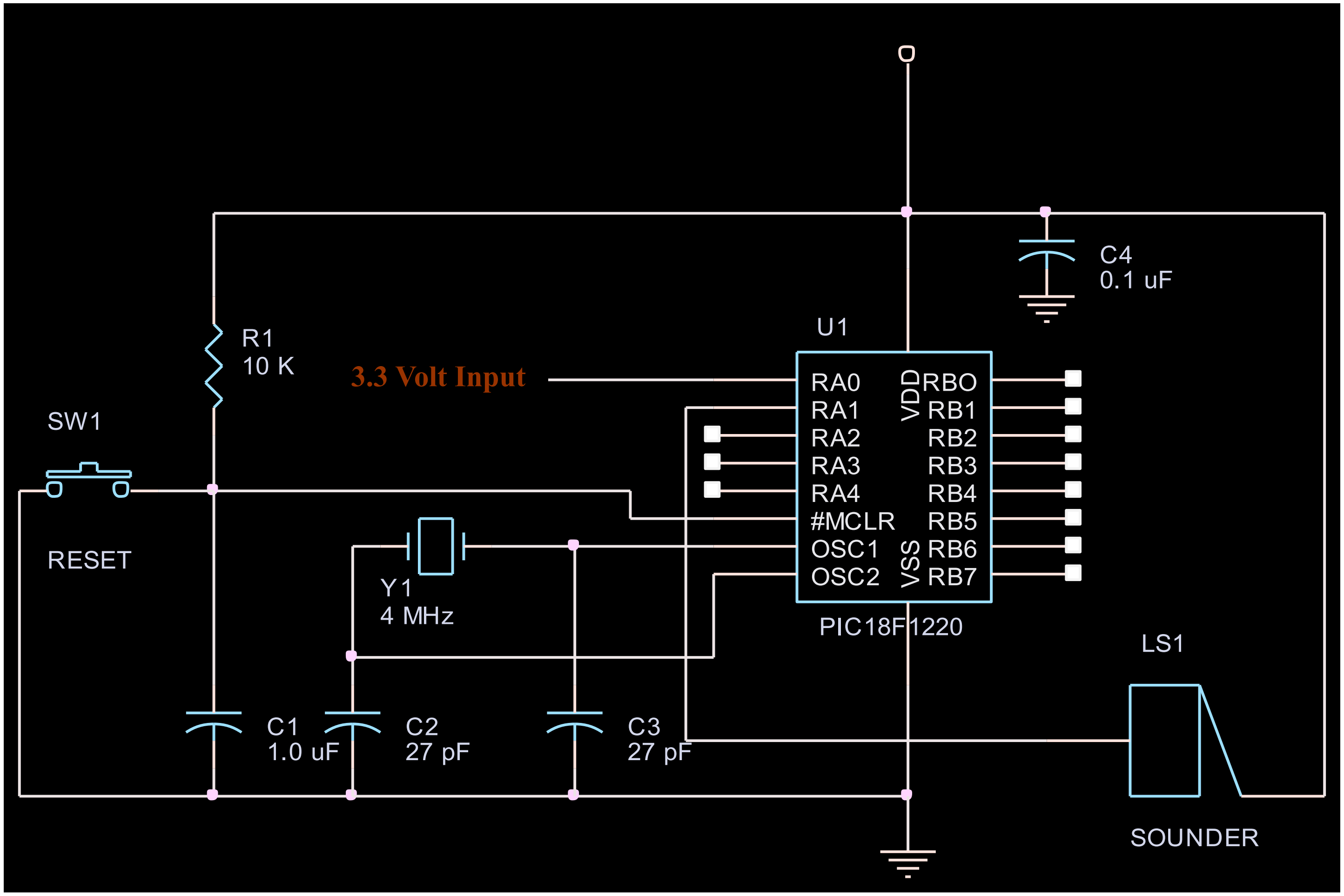
Interfacing a Temperature Sensor

(7 of 7)

- Software modules
  - Program should be divided into the following:
    - Setup all analog ports and channels
    - Assume ACQT=(TAD . 12) and Fosc / 16
    - Initialize A/D module (acquire analog input)
    - Start a conversion and read the digital reading at the end of the conversion.
      - Multiply the temperature reading by 10
      - Divide the 16-bit result by 102 → QUO and REM → This is the equivalent temperature reading.
    - Convert the result in BCD.
    - Convert the BCD numbers in ASCII code.

This is basically what you do in your HW; you can use C code!

Another Example in C –
Monitor 3.3 V supply in a PC Power Supply



Example in C –
Monitor 3.3 V supply in a PC Power Supply

Basic Idea:

Check PORT RA0 . If the Voltage is changed by 5 percent activate the alarm!

Change the program so the port is checked every sec!

```
/*
 * Sampling the 3.3 Volt supply for a PIC18F1220
 */
#include <p18cxxx.h>
/* Set configuration bits
 *  - set OSC input external oscillator
 *  - disable watchdog timer
 *  - disable low voltage programming
 *  - disable brownout reset
 *  - enable master clear
 */
#pragma config OSC = HS
#pragma config WDT = OFF
#pragma config LVP = OFF
#pragma config BOR = OFF
#pragma config MCLRE = ON
#pragma code
```

```
float getVoltage(void)
{
          ADCON0bits.GO = 1;                    //GO bits → start a conversion

          while ( ADCON0bits.GO == 1 );         // wait for completion
          return ( ADRESL | ( ADRESH << 8 ) ) * 0.00489;
          // 5/1024 ;wait for completion; return the concatenated value in decimal

}
// main program
void main (void)
{
          ADCON0 = 0x01;              // select input AN0, enable ADC
          ADCON1 = 0x0e;              // AN0 is analog, VDD and VSS are references
          ADCON2 = 0x84;              // convert using 1 MHz
          TRISA = 1;                  // Port A bit 0 = input for ADC
          PORTA = 0;                  // alarm off
          while (1)
          {
                    if ( getVoltage() > 3.465 || getVoltage() < 3.135 )
                              PORTAbits.RA1 = 1;                  // alarm on
                    else
                              PORTAbits.RA1 = 0;                  // alarm off
          }
}
```

ADCON0

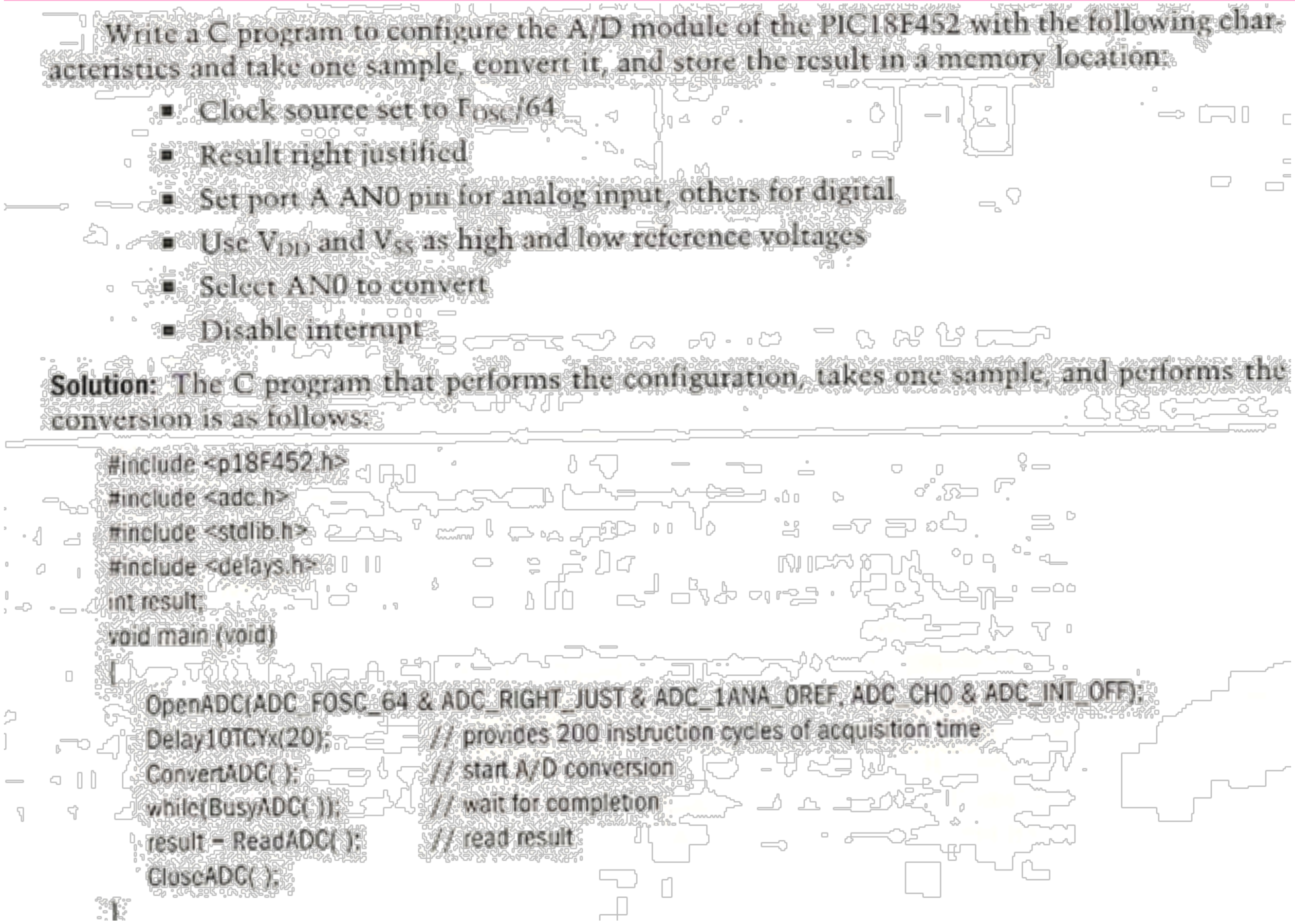
		CHS3	CHS2	CHS1	CHS0	Go/$\overline{\text{Done}}$	ADON

Example – in C

Read the link below for full library functions in <adc.h>

http://ww1.microchip.com/downloads/en/devicedoc/mplab_c18_libraries_51297f.pdf

Read Chapter 2



Write a C program to configure the A/D module of the PIC18F452 with the following characteristics and take one sample, convert it, and store the result in a memory location:

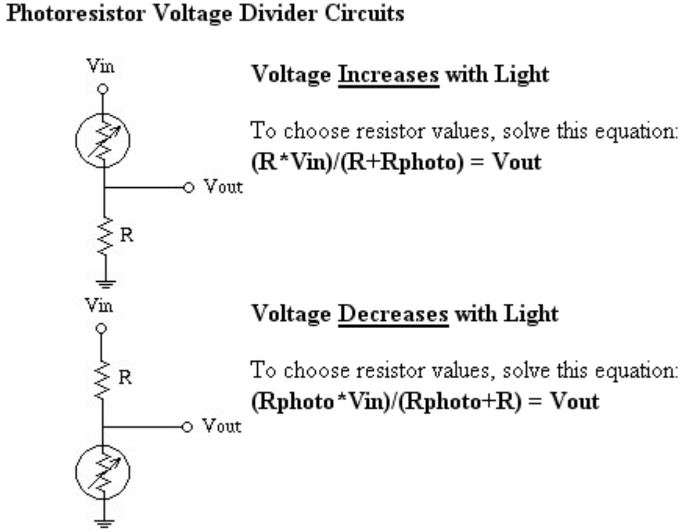
- Clock source set to $F_{OSC}/64$
- Result right justified
- Set port A AN0 pin for analog input, others for digital
- Use V_{DD} and V_{SS} as high and low reference voltages
- Select AN0 to convert
- Disable interrupt

Solution: The C program that performs the configuration, takes one sample, and performs the conversion is as follows:

```
#include <p18F452.h>
#include <adc.h>
#include <stdlib.h>
#include <delays.h>
int result;
void main (void)
{
    OpenADC(ADC_FOSC_64 & ADC_RIGHT_JUST & ADC_1ANA_0REF, ADC_CH0 & ADC_INT_OFF);
    Delay10TCYx(20);          // provides 200 instruction cycles of acquisition time
    ConvertADC( );            // start A/D conversion
    while(BusyADC( ));        // wait for completion
    result = ReadADC( );      // read result
    CloseADC( );
}
```

LAB

- Show that Example 12.3 on page 381 works.
- What is the range of the photoresistor
  - **Photoresistors** (also often called **phototransistors** or **CdS photoconductive photocells)**
  - Simple resistors that **altar resistance** depending on the amount of **light** place over them.
  - Used in photovore is a robots (robots chasing light)



Photoresistor Voltage Divider Circuits

Voltage Increases with Light

To choose resistor values, solve this equation:
(R*Vin)/(R+Rphoto) = Vout

Voltage Decreases with Light

To choose resistor values, solve this equation:
(Rphoto*Vin)/(Rphoto+R) = Vout

You should find out the range of your photoresistor as the light intensity changes.

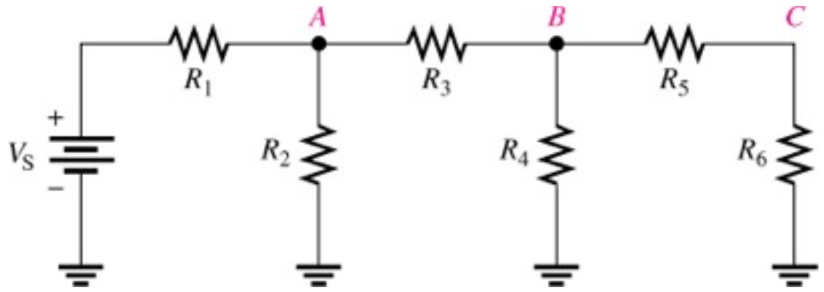
http://www.societyofrobots.com/sensors_color.shtml

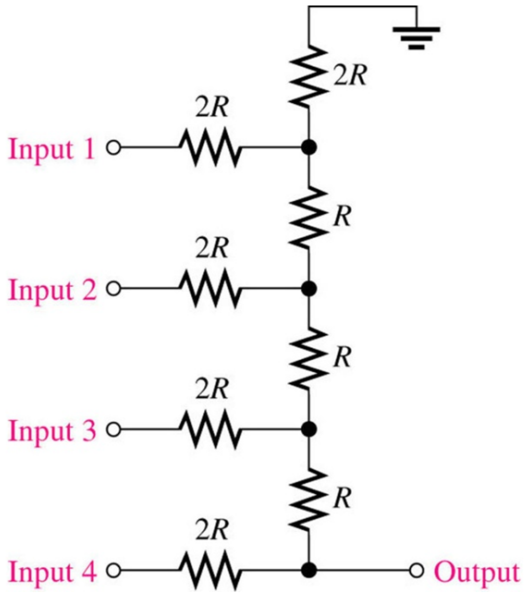
Digital to Analog (D/A, DAC, or D-to-A) Conversion

- Converting discrete signals into discrete analog values that represent the magnitude of the input signal compared to a standard or reference voltage
  - The output of the DAC is discrete analog steps.
  - By increasing the resolution (number of bits), the step size is reduced, and the output approximates a continuous analog signal.

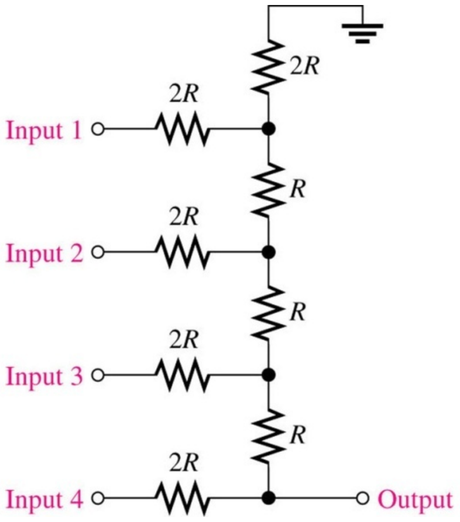
Analysis of a Ladder Network

- A resistive ladder network is a special type of series-parallel circuit.
- One form of ladder network is commonly used to scale down voltages to certain weighted values for digital-to-analog conversion
  - Called R/2R Ladder Network
- To find total resistance of a ladder network, start at the point farthest from the source and reduce the resistance in steps.



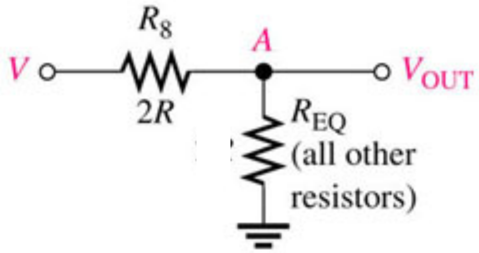


The R/R2 Ladder Network

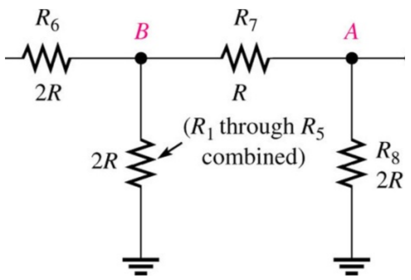


Used for Digital-to-analog converter!

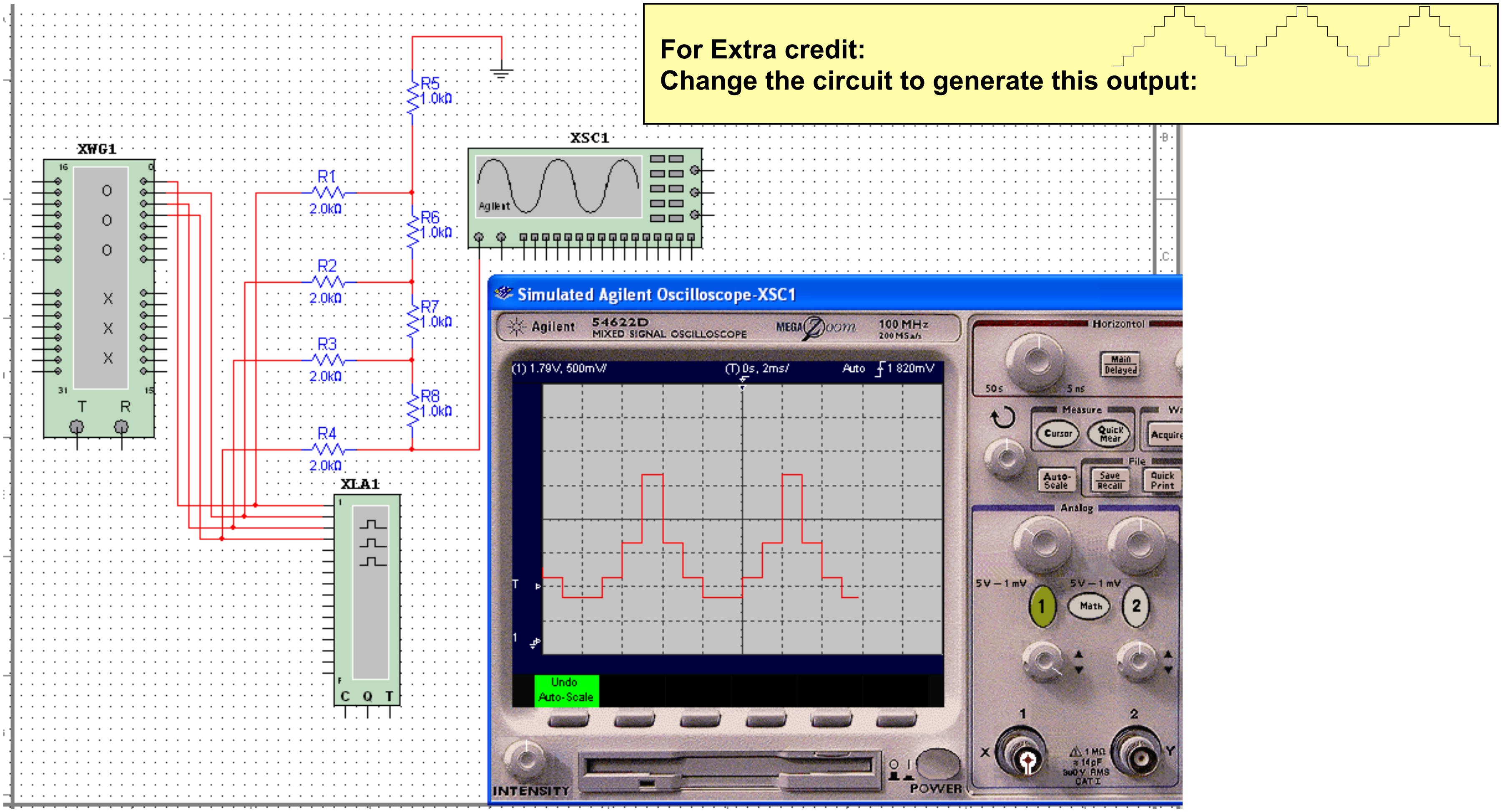
Only Input 4 is HIGH



Only Input 3 is HIGH



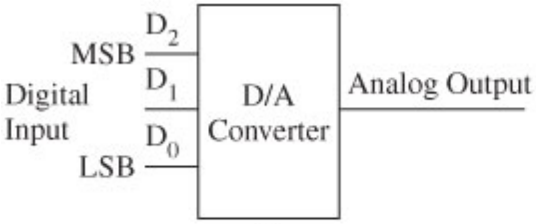
Examining Digital-to-Analog Conversion



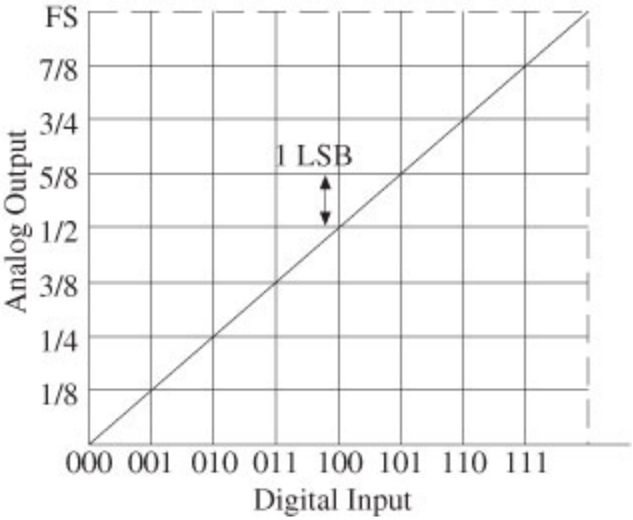
For Extra credit:
Change the circuit to generate this output:

Digital to Analog Conversion

- The resolution of a DAC is defined in terms of bits—the same way as in ADC.
- The values of LSB, MSB, and full-scale voltages calculated the same way as in the ADC.
- The largest input signal 111 is equivalent of 7/8 of the full-scale analog value.



(a)

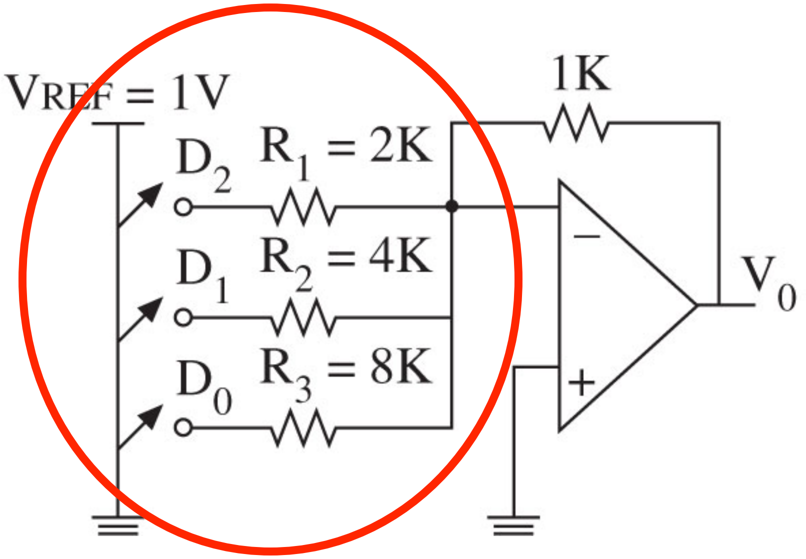


D/A Converter Circuits (1 of 4)

- Can be designed using an operational amplifier and appropriate combination of resistors
- Resistors connected to data bits are in binary weighted proportion, and each is twice the value of the previous one.
- Each input signal can be connected to the op amp by turning on its switch to the reference voltage that represents logic 1.
  - If the switch is off, the input signal is logic 0.

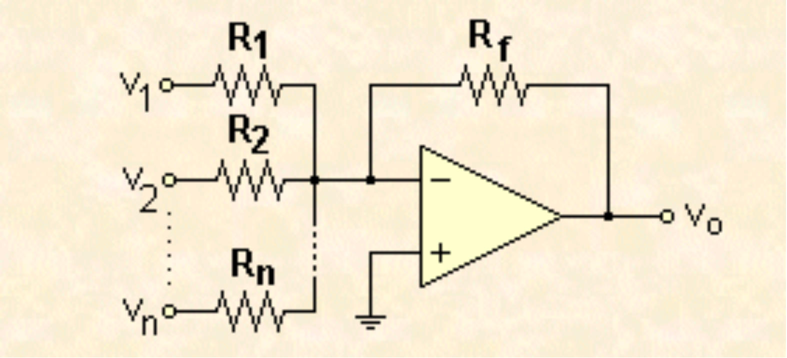
D/A Converter Circuits (2 of 4)

- 3-bit D/A Converter Circuit



Summing amplifier

- R/2R Ladder Network for D/A Converter



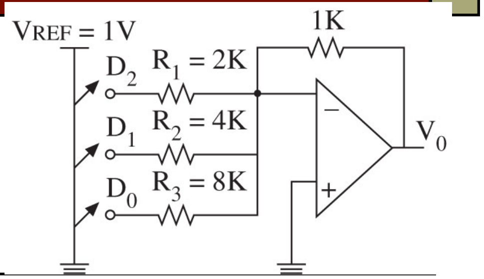
The transfer function of the summing amplifier :

$vo = -(v1/R1 + v2/R2 + \ldots + vn/Rn)Rf$

Thus if all input resistors are equal, the output is a scaled sum of all inputs.

If they are different, the output is a weighted linear sum of all inputs.

D/A Converter Circuits



- If the reference voltage is 1 V, and if all switches are connected, the output current can be calculated as follows:

$$I_o = I_T = I_1 + I_2 + I_3 = \frac{V_{REF}}{R_1} + \frac{V_{REF}}{R_2} + \frac{V_{REF}}{R_3} = \frac{V_{REF}}{1k}\left(\frac{1}{2} + \frac{1}{4} + \frac{1}{8}\right) = 0.875\,mA$$

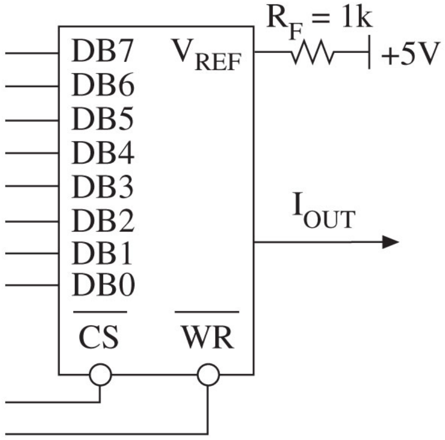
- Output voltage

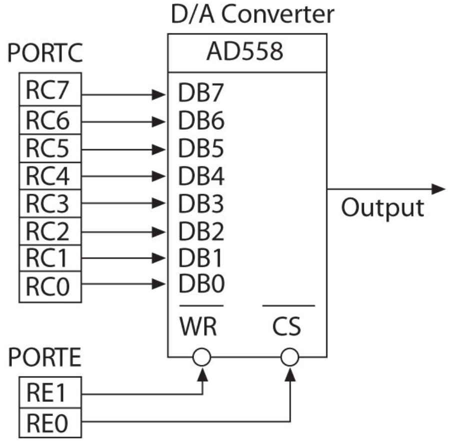
$$V_O = -R_f I_T = -(1k)\times(0.875\text{ mA}) = -0.875\text{ V} = \left|{}^{7}\!/_{8}\text{ V}\right|$$

Note that the output will be inverted!

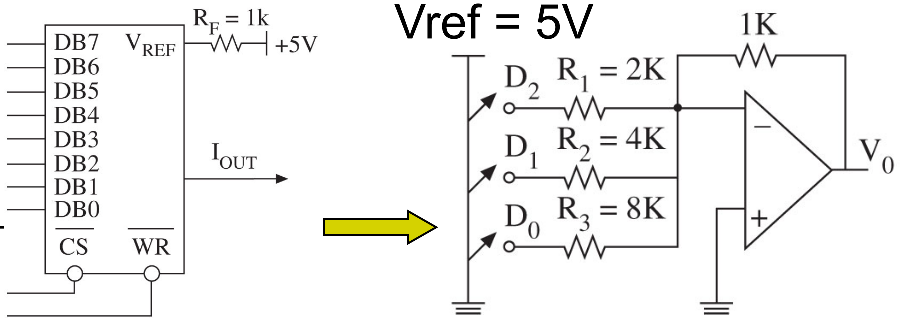
D/A Converters as Integrated Circuits

- D/A converters are available commercially as integrated circuits
- Can be classified in three categories.
  - Current output, voltage output, and multiplying type
    - Current output DAC provides the current I_O as output signal
    - Voltage output D/A converts I_O into voltage internally by using an op amp and provides the voltage as output signal
    - In multiplying DAC, the output is product of the input voltage and the reference source V_{REF}.
  - Conceptually, all three types are similar





Example 12.4



- What will be the analog equivalent of 1001 0001?

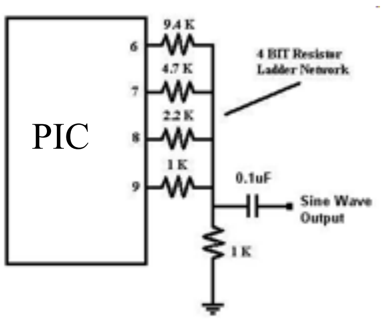
$$I_O = \frac{V_{REF}}{R_{REF}}\left(\frac{A_1}{2} + \frac{A_2}{4} + \frac{A_3}{8} + \ldots\ldots + \frac{A_n}{2^n}\right)$$

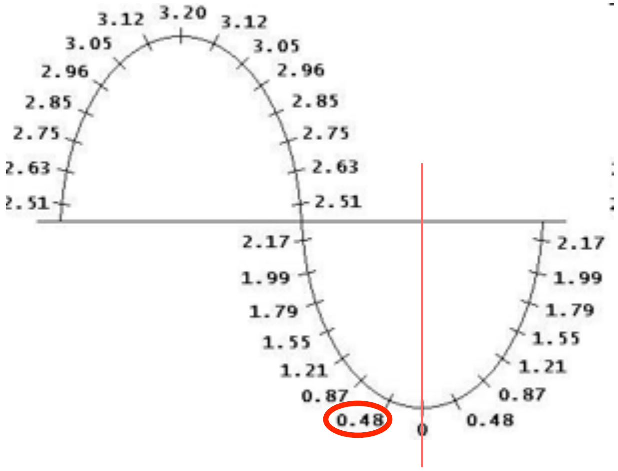
$$I_O = \frac{V_{REF}}{R_{REF}}\left(\frac{DB7}{2} + \frac{DB6}{4} + \frac{DB5}{8} + \ldots\ldots + \frac{DB0}{2^8}\right)$$

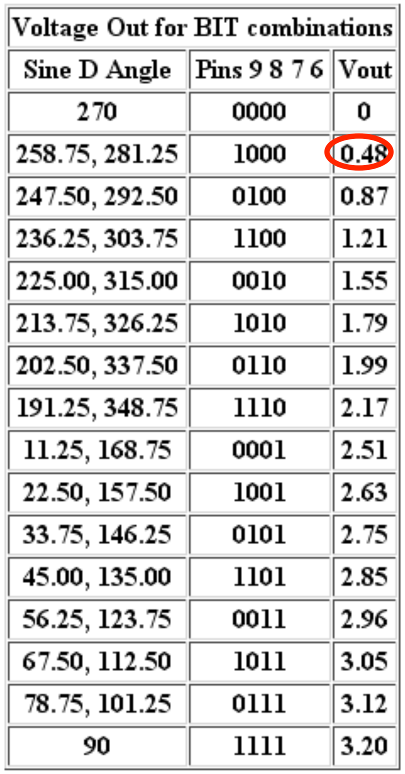
$$I_O = 5\text{ mA}\left(\frac{1}{2} + \frac{0}{4} + \frac{0}{8} + \frac{1}{16} + \frac{0}{32} + \frac{0}{64} + \frac{0}{128} + \frac{1}{256}\right) = 2.832\text{ mA}$$

How can you generate a sine wave?

- Theoretically the voltages would range from 0 to 5
- How do you change the frequency?







Voltage Out for BIT combinations		
Sine D Angle	Pins 9 8 7 6	Vout
270	0000	0
258.75, 281.25	1000	0.48
247.50, 292.50	0100	0.87
236.25, 303.75	1100	1.21
225.00, 315.00	0010	1.55
213.75, 326.25	1010	1.79
202.50, 337.50	0110	1.99
191.25, 348.75	1110	2.17
11.25, 168.75	0001	2.51
22.50, 157.50	1001	2.63
33.75, 146.25	0101	2.75
45.00, 135.00	1101	2.85
56.25, 123.75	0011	2.96
67.50, 112.50	1011	3.05
78.75, 101.25	0111	3.12
90	1111	3.20

References

- http://www.engin.brown.edu/courses/en123/Lectures/DAconv.htm
- http://www.seas.upenn.edu/~ese206/labs/adc206/adc206.html
- Interesting project ideas: http://www.byonics.com/
- More interesting project ideas… http://www.ke4nyv.com/picprojects.htm
- Project RGB LED: http://rgb.kitiyo.com/files/final%20report.pdf
- More interesting projects http://www.semifluid.com/?cat=2