Analogue to Digital Converters

- To process continuous waveforms (analogue) in a computer, signals must be converted into a digital (binary) form.
- Require A/D converter
  - samples the analogue waveform at regular intervals of time, frequently enough so that all the information from the original analogue signal is included in the sampled version
  - ***Nyquist criteria requires sampling at twice the frequency of the highest frequency component of the signal.***

- each analogue sample is quantized and encoded
  - approximated to the nearest digital level, according to the number of bits available for the conversion process.

Examples:

- 8-bit A/D converter
  - 2^8=256 levels are present in the output
- 10-bit A/D converter
  - 2^{10}=1024 levels are present in the output

In A/D conversion

- analogue waveform is sampled and the resulting discrete-time analogue samples are quantized (see quantization) by truncating or rounding
  - quantized values are encoded into a digital word.

- The greater the number of bits, the higher the output resolution, but the more complex, slower and costly is the converter.

- input signal is often passed through an analog scaler/level shifter to adjust its range to fit that of the converter.

- In closed loop control applications, the time required for conversion is important
  - many types of converter availablewith varying conve rsion speeds.

- **A/D converter in the Digital Unit**
  - **eight bit counter type, converter**
  - **maximum conversion time 500 microseconds.**
  - **Input bipolar range -10 to +10 volts**

Digital to Analogue Converters (DAC)

- circuit that provides an analogue voltage (or current) that is weighted sum of the bits in the digital word.

- Digital signals derived from a computer may be required in a continuous (analogue) form
  - examples include
    - motor velocity and position control
    - audio signals in a digital communication system.

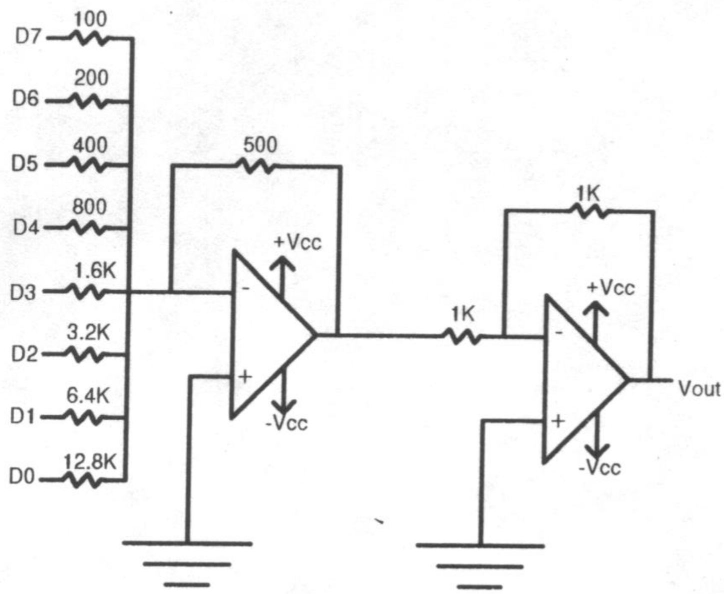
- converted signal has abrupt discontinuities and presents a staircase type of pattern.
  - low pass filtering can be used to smooth the restored analogue signal
    - removes high-frequency content.

- 8 bit device has 256 levels

- 10 bit device has 1024 levels
- output voltage is often passed through an analog scaler/level shifter to obtain the range required for a particular application.

- **D/A converter in the Digital Unit**
  - **8 bit converter**
  - **output analog bipolar range +10 to -10 volts.**

A simple Weighted Resistor, Op Amp D/A Converter

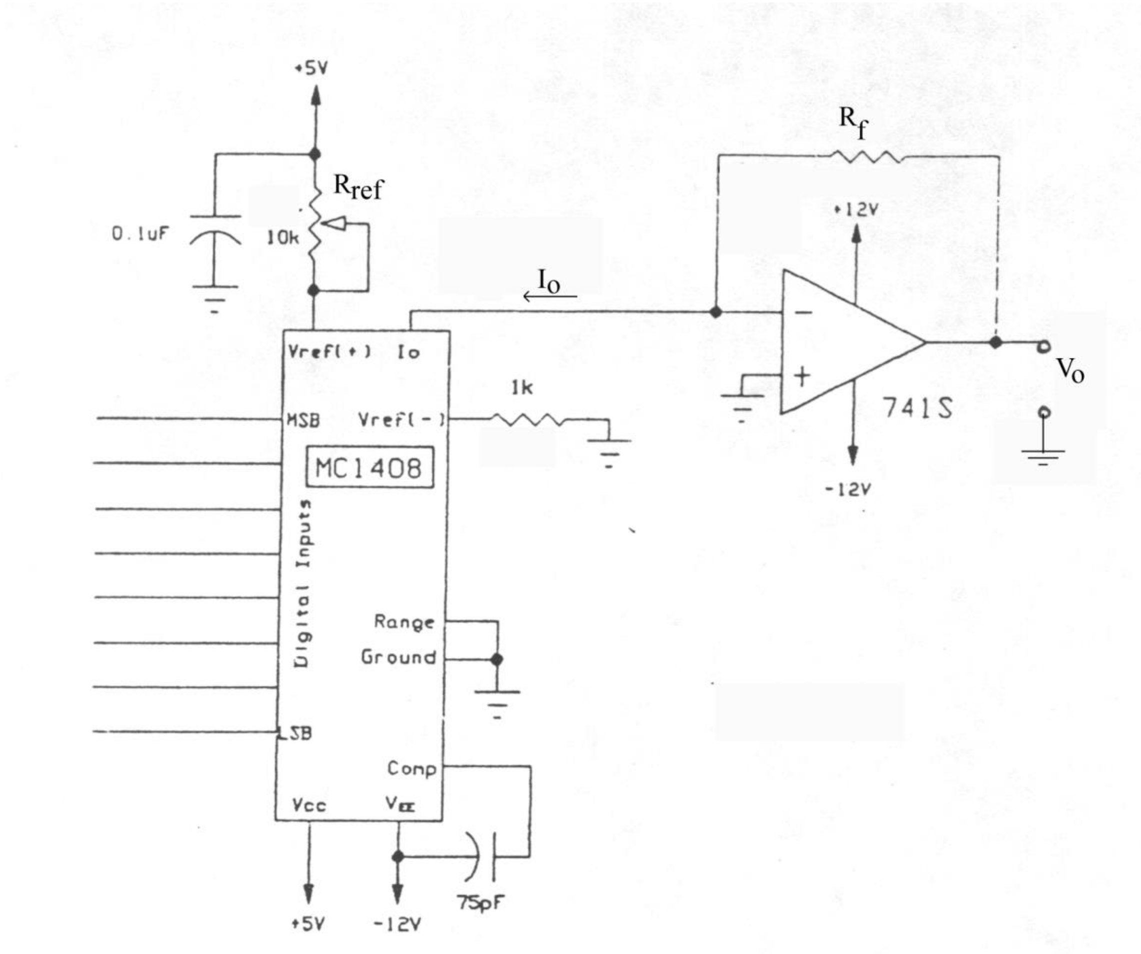


Assume logic level “high” or “1” is 1 volts
Assume logic level “low” or “0” is 0 volts

1) Find all currents and voltages for all data bits “high” or logic level “1”
2) Find Output voltage range of D/A
3) Find Quantization error
4) Find the output voltage for the following data input to D/A

Data Input	Voltage V_o
0x80	
0x40	
0x20	
0x10	
0x08	
0x04	
0x02	
0x01	

Chip Version of 8 Bit D/A



TYPES OF ANALOGUE-TO-DIGITAL CONVERTERS

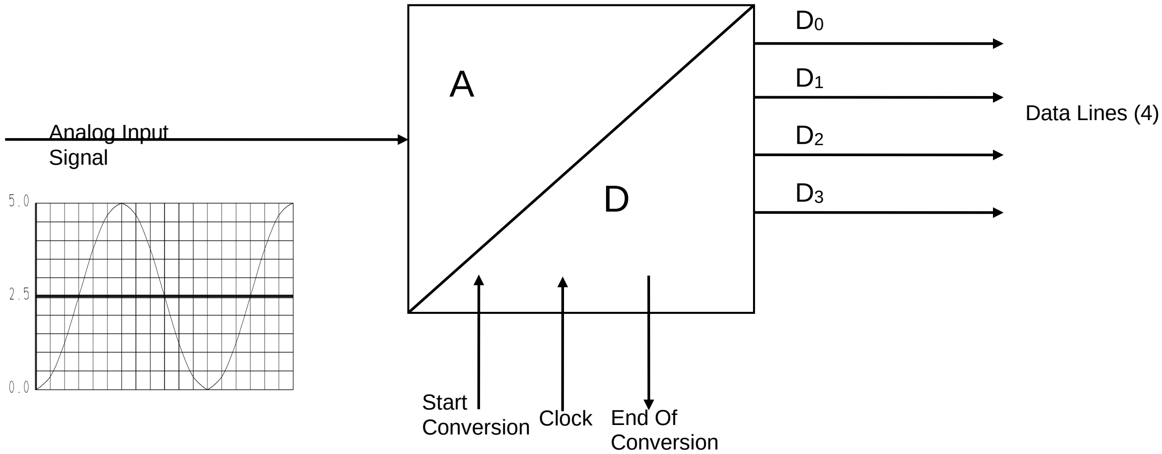
Successive Approximation Converter

- provides a moderate rate of conversion of an analog signal
- works by first comparing the input with a voltage which is half the input range.
    - If the input is over this level it compares it with three-quarters of the range
    - If the input is over this level it compares it with seven-eights of the range etc
    - Twelve such steps gives 12-bit resolution.

- While comparisons are taking place the signal is frozen in a sample and hold circuit.
- After A-D conversion the resulting bytes are placed into either a pipeline or buffer store.
    - pipeline store enables the A-D converter to do another conversion while the previous data is transferred to the computer.
    - Buffered A-D converters place the data into a queue held in buffer memory
        - computer can read the converted value immediately, or
        - can allow values to accumulate in the buffer and read them when it is convenient.
            - frees the computer from having to deal with the samples in real time
                - allows buffered samples to be processed in batches without losing data.

Analog to Digital Conversion and Quantization Error

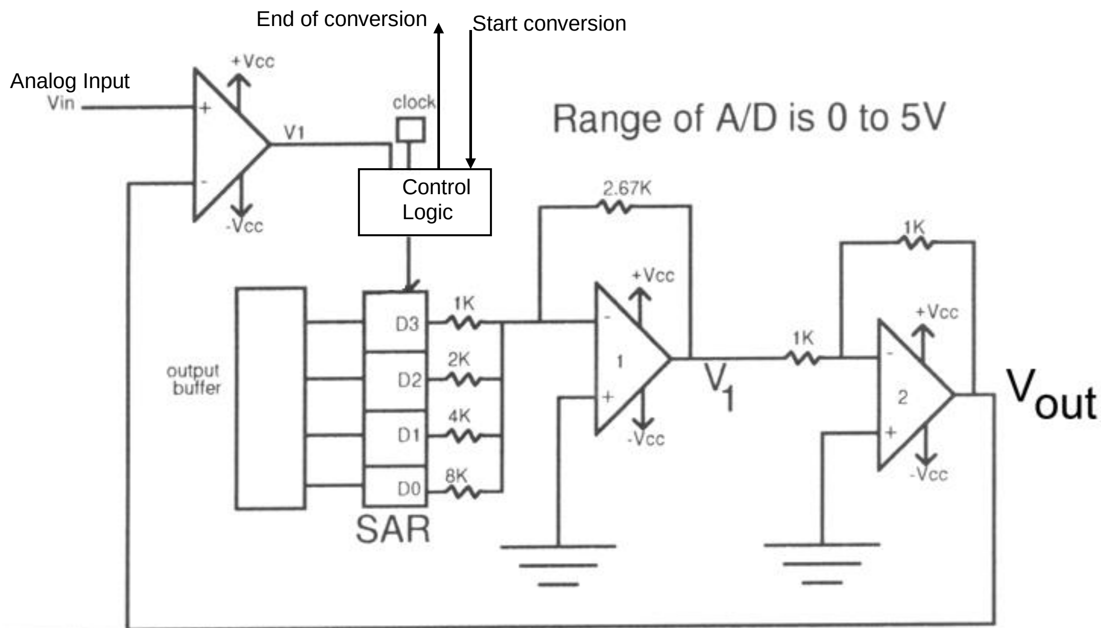
Range of A/D is 0 to 5 volts



Analog Signal	D_3	D_2	D_1	D_0
0 V	0	0	0	0
0.333	0	0	0	1
0.666	0	0	1	0
1.000	0	0	1	1
1.333	0	1	0	0
1.666	0	1	0	1
2.000	0	1	1	0
2.333	0	1	1	1
2.666	1	0	0	0
3.000	1	0	0	1
3.333	1	0	1	0
3.666	1	0	1	1
4.000	1	1	0	0
4.333	1	1	0	1
4.666	1	1	1	0
5.000	1	1	1	1

- Quantization error
  - A/D cannot represent every single analog value.
  - Data lines with the bits 0001 could represent an analog voltage between 0.333 volts and 0 .666 volts.
  - uncertainty of 0.333v called the quantization error of the A/D
    - function of the number of data lines (bits) associated with the A/D.
    - Quantization Error = $\frac{1}{2^n - 1} V_R$ where n is number of data lines (bits) and V_R is range of A/D. In
    - example above Quantization Error = $\frac{1}{16-1} x5 = 0.333$ volts

4 Bit Successive Approximation Converter – Range 0 to 5 V
Data Bits – Logic Level “High” or “1” is 1 Volt and Logic Level “Low” or “0” is 0 Volts



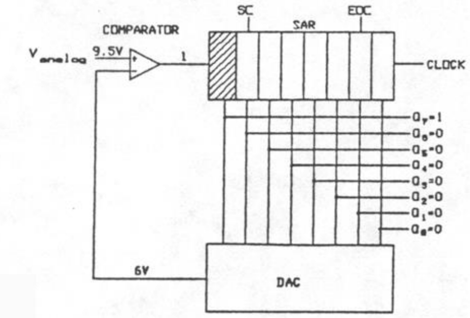
D3	D2	D1	D0

What is the voltage V_1 at the output of op amp 1 and the voltage V_{out} at the output of op amp 2.

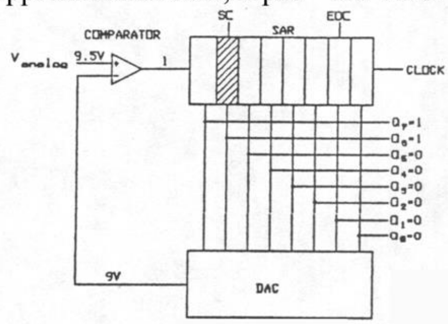
V_1 =

V_{out} = .

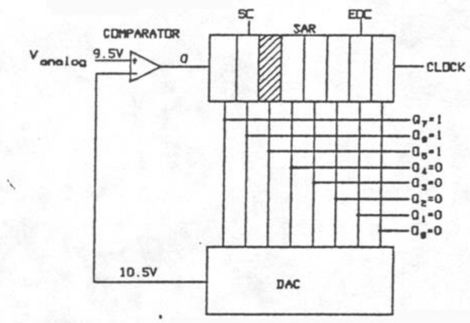
12 V 8 bit Successive Approximation A/D, Input = 9.5 VDC



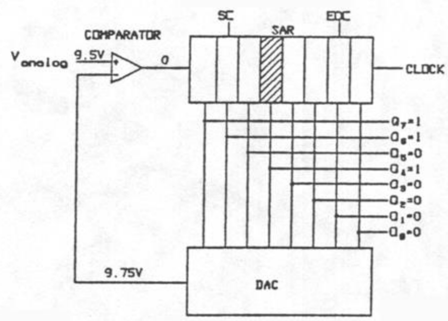
10000000



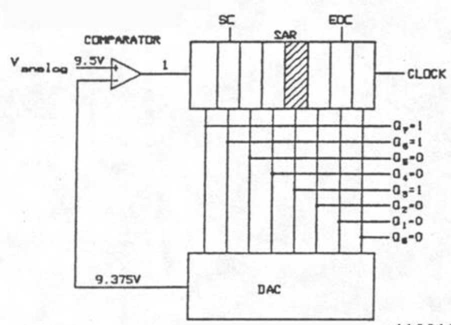
11000000



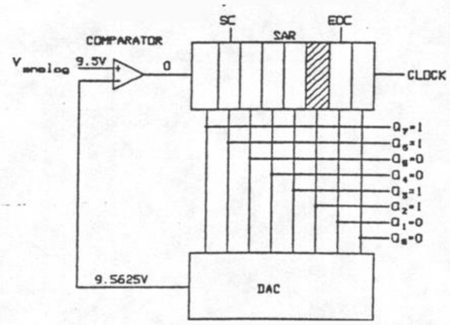
11000000



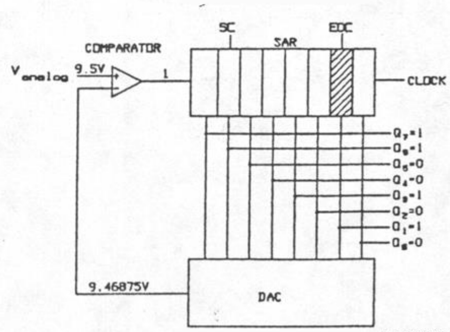
11000000



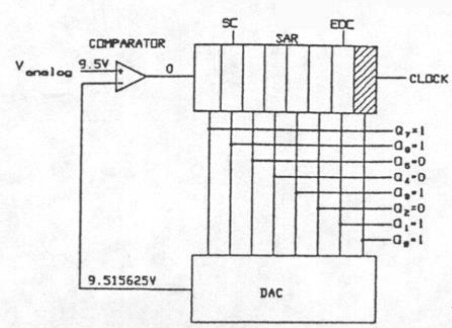
11001000



11001000

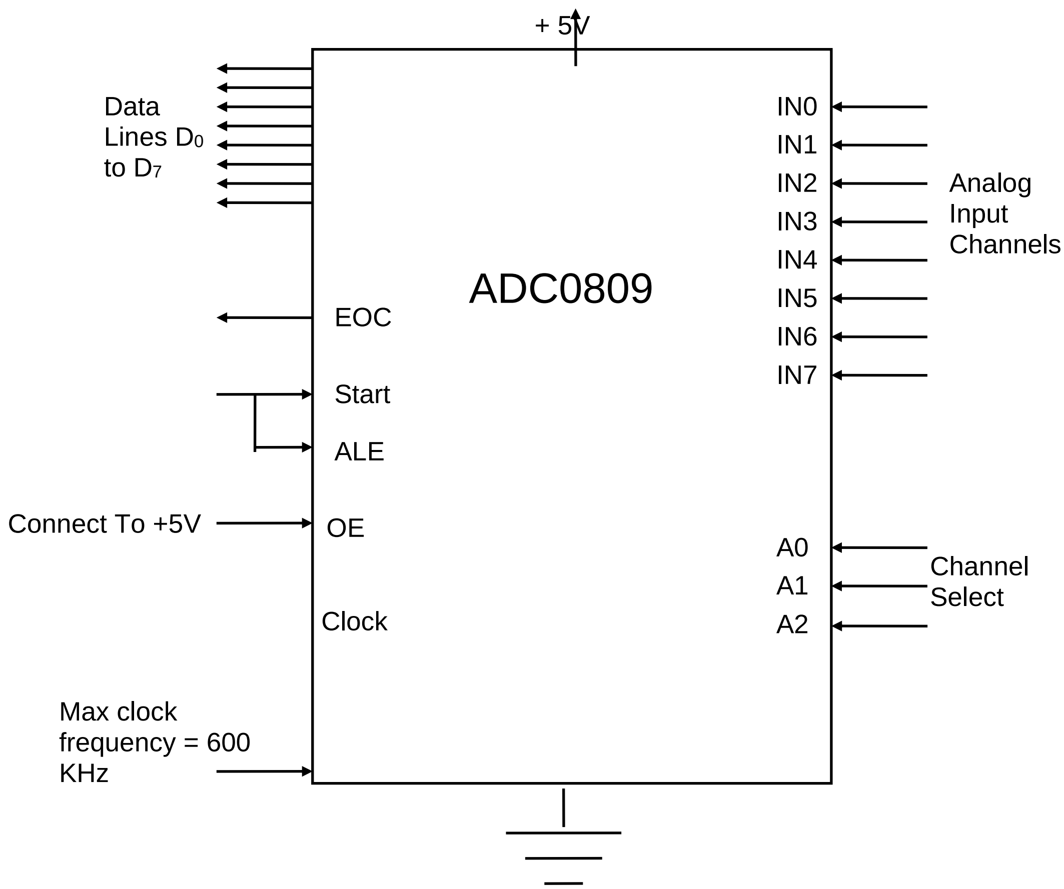


11001010



11001010

ADC0809 8 channel, 8 bit Successive Approximation Converter



Selected Channel	A2	A1	A0
IN0	0	0	0
IN1	0	0	1
IN2	0	1	0
IN3	0	1	1
IN4	1	0	0
IN5	1	0	1
IN6	1	1	0
IN7	1	1	1

Conversion Process
1) Set Address lines A0, A1, A2 to select input channel
2) send a start conversion bit - causes EOC line to go low
3) Wait for the EOC line to go back high indicating conversion complete
4) Read data in data in data buffer

Dual Slope Integrating Converter

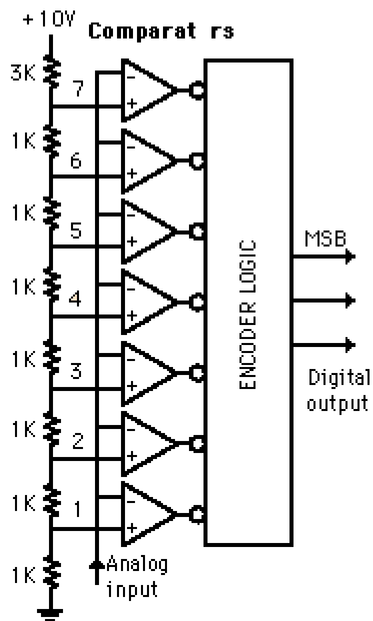
- slower than the successive approximation type.
- input signal charges a capacitor for a fixed period and then measures the time for the capacitor to fully discharge at a fixed rate.
- time is a measure of the integrated input voltage
    - reduces the effects of noise.

Flash Converter

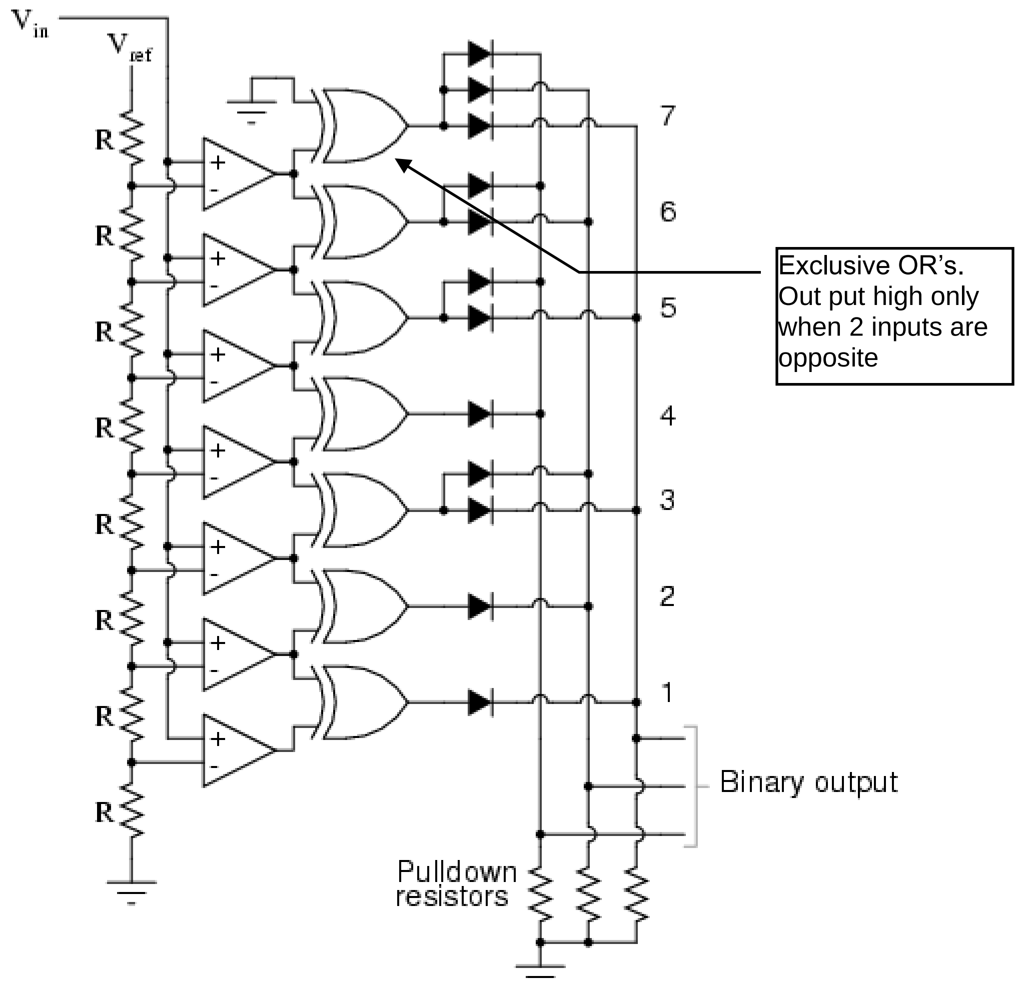
- fastest type of converter
- works by comparing the input signal to a reference voltage
    - has as many comparators as there are steps in the comparison.
    - 8-bit converter has 2 to the power 8, or 256, comparators.

- resistor net and comparators provide an input to the combinational logic circuit
- conversion time is propagation delay through the network
- not limited by the clock rate or some convergence sequence.
- encoder logic executes truth table to convert the ladder of inputs to the binary number output.

3-bit flash ADC with resolution 1 volt



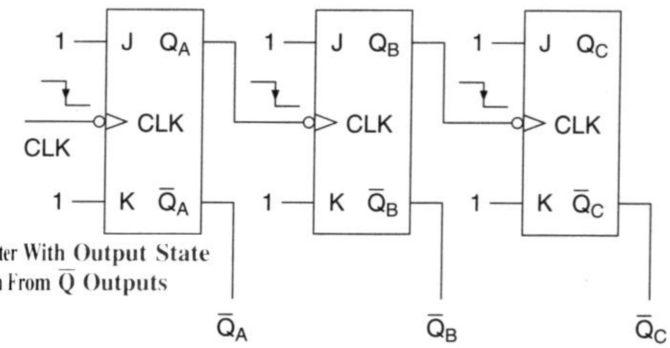
3 Bit Flash Converter with Encoder Circuitry



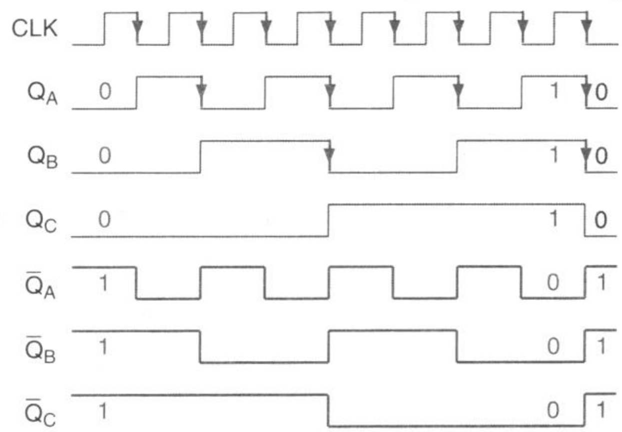
Comparator # High	MSB		LSB
1	0	0	1
1,2	0	1	0
1,2,3	0	1	1
1,2,3,4	1	0	0
1,2,3,4,5	1	0	1
1,2,3,4,5,6	1	1	0
1,2,3,4,5,6,7	1	1	1

Counter Type Analog to Digital Converter (used in digital servo board)

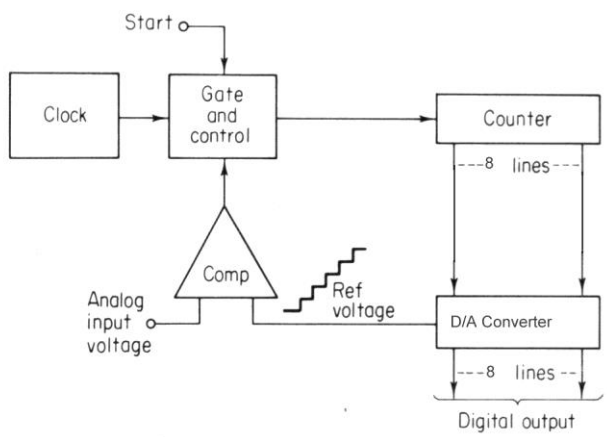
- Counter type requires incrementing a register one bit at a time
- Register output is converted to an analog voltage using a D/A Converter
- Analog Voltage compared to input signal.
- Conversion is complete when analog inout reference signal equals the D/A output
- Slow, and inefficient



Counter With Output State Taken From $\overline{Q}$ Outputs



Timing Diagram of Counter



MAY 1994

DS3005-2.0

ZN425E8

8-BIT D-A/A-D CONVERTER

The ZN425 is a monolithic 8-bit D-A converter containing an R-2R ladder network of diffused resistors with precision bipolar switches, and in addition a counter and a 2.5V precision voltage reference. The counter is a powerful addition which allows a precision staircase to be generated very simply by clocking the counter.

FEATURES

- $\pm^1/_2$ LSB Linearity Error
- 0°C to +70°C
- TTL and 5V CMOS Compatible
- Single +5V Supply
- Settling Time (D-A) 1µs Typical
- Conversion Time (A-D) 1ms Typical, using Ramp and Compare Technique
- Extra Components Required
  D-A: Reference Capacitor (Direct Voltage Output through 10kOhms Typ.)
  A-D: Comparator, Gate, Clock and Reference Capacitor

ORDERING INFORMATION

Ambient operating temperature 0°C to +70°C
PackageDP16

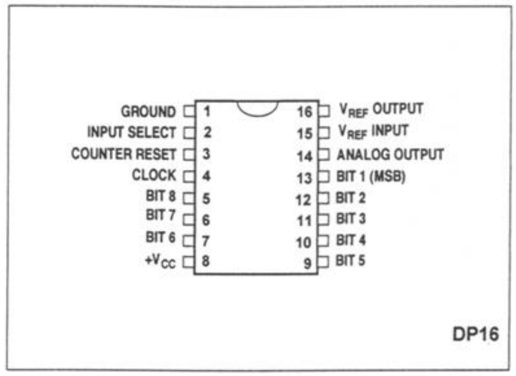


Fig.1 Pin connections (not to scale) - top view

ABSOLUTE MAXIMUM RATINGS

Supply voltage, V_{CC}	+7.0V
Max. voltage, logic and V_{REF} inputs	+5.5V See note 3
Operating temperature range	0°C to +70°C
Storage temperature range	-55°C to +125°C

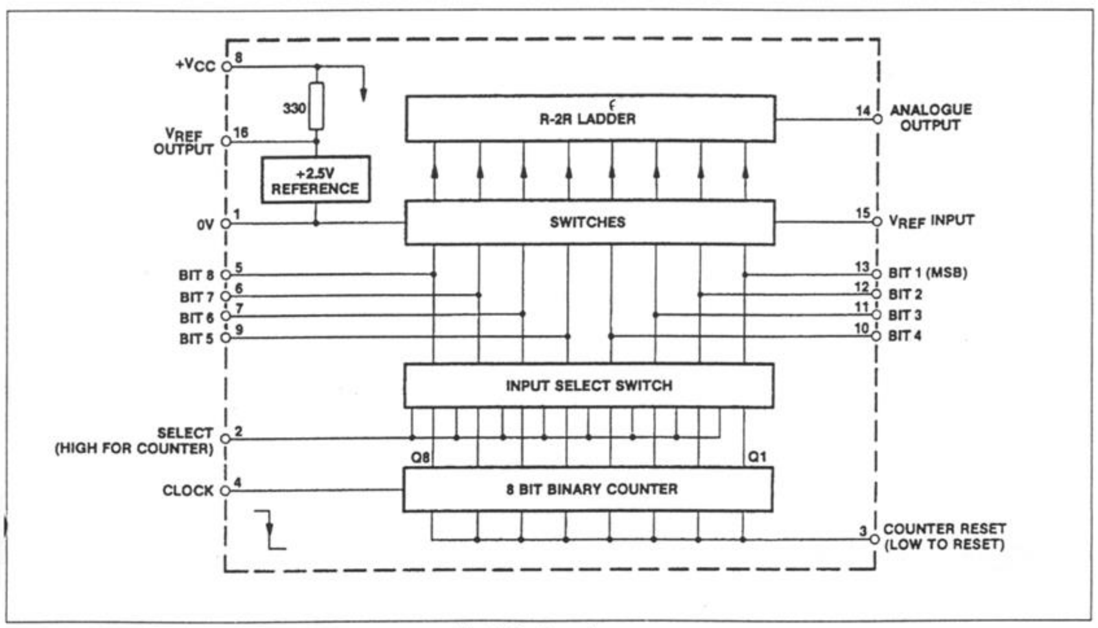
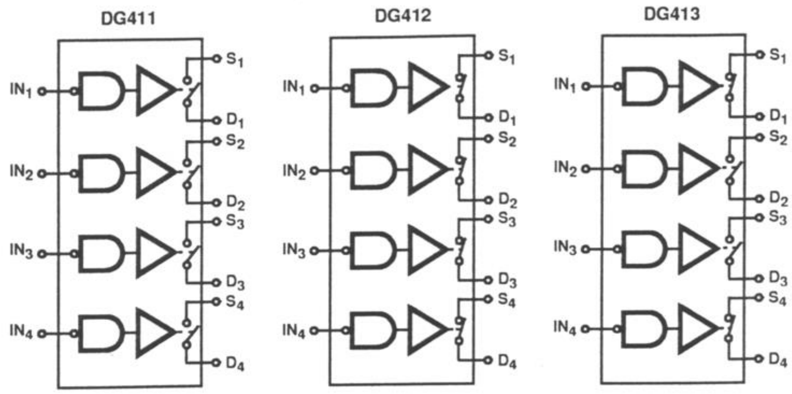


Fig.2 System diagram

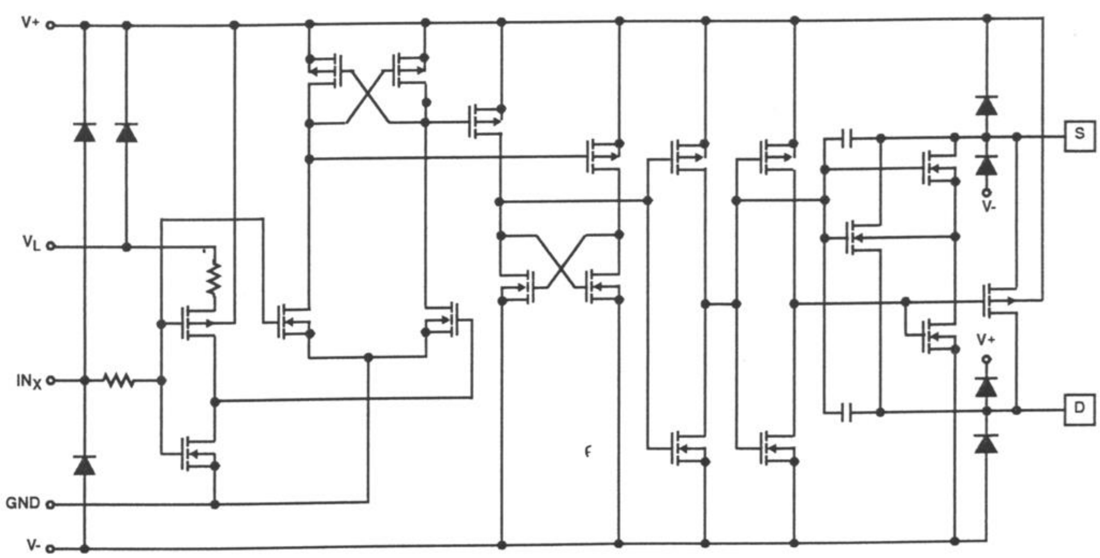
Functional Diagrams

Four SPST Switches per Package Switches Shown for Logic "1" Input



Typical Schematic Diagram

(Typical Channel)



Specifications of Analogue-to-Digital Converters

Resolution

- number of steps the input range is divided into.
- expressed as bits (n) and the number of steps is 2 to the power n
- example:
  - converter with 12-bit resolution divides the range into 2^{12}, or 4096 steps (0 to 4095)
    - a 0-10 V range will be resolved to 10/4095 or 2.44 mV
    - a 0-100 mV range will be resolved to 0.0244 mV.

Linearity

- ***Ideally*** an A-D converter with n-bit resolution will convert the input range into (2 to the power n)-1 equal steps (4095. Steps in the case of a 12-bit converter).
- ***Practically***, steps are not exactly equal
  - leads to non-linearity in a plot of A- D output against input voltage.

Sample and Hold Acquisition Time

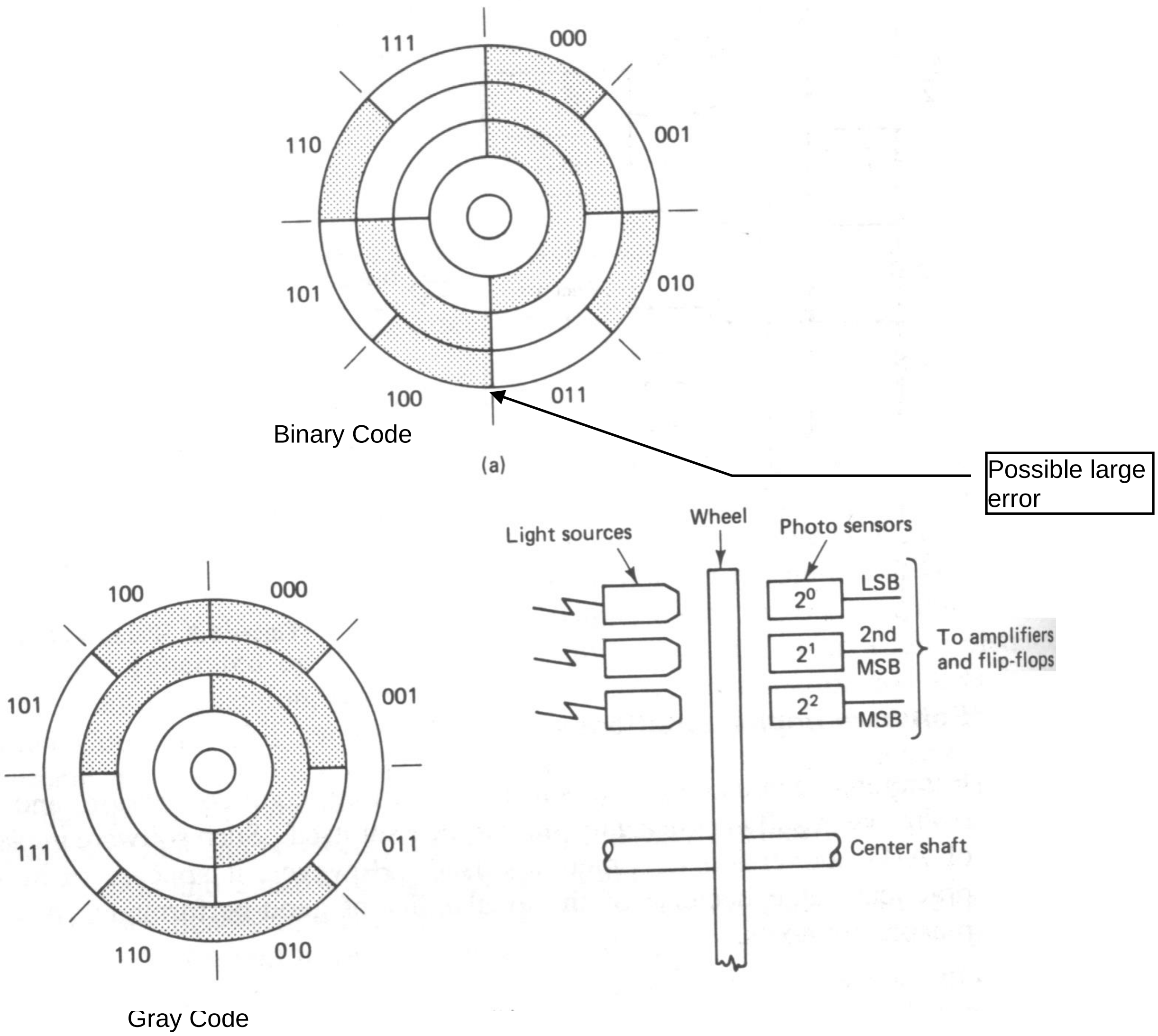
- freezes the analogue input voltage at the moment the sample is required.
  - voltage is held constant while the A-D converter digitizes it.
- acquisition time
  - time between releasing the hold state and the output of the sample circuit settling to the new input voltage value.

Throughput

- maximum rate at which the A-D converter can output data values.
- inverse of the (conversion time + the acquisition time) of the A-D converter.
- Example:
  - converter that takes 10 microseconds to acquire and convert will be able to generate about 100 000 samples per second.
- Throughput increased by using a pipelined A-D converter
  - a second conversion can start while the first is still in progress.

Measuring Motor Shaft Position

3 Bit Absolute Encoder Example Implemented in Binary and Gray Code



Absolute Encoder Patterns for 3 Bit Resolution

Decimal	**Pure Binary**	**Gray Code**
0	000	000
1	001	001
2	010	011
3	011	010
4	100	110
5	101	111
6	110	101
7	111	100

Feedback Unit
6 bit Absolute
Encoder Disk

- Absolute encoder consists of main disk attached to rotating shaft.
- On one side of disk is pattern of infra-red leds.
- On other side of disk is mirrored pattern of infra red receivers
- Disk has an etched patter which either blocks or allows light from led to pass through to receiver
- As shaft rotates, pattern of receivers which are either on or off changes.
- pattern never repeats itself during one revolution of main shaft and can thus be used to determine position absolutely.
- Software translates Gray code to a binary or decimal

The Following Function converts 6 bit Feedback unit Gray Code to a decimal number. Resolution of 6 bit encoder is 360/64 or 5.625 deg

```
int gray_decimal(int gray_value)
{
        int bit5=0, bit4=0, bit3=0, bit2=0, bit1=0, bit0=0;

        bit5 = (gray_value&0x20)>>5;
        bit4 = (gray_value&0x10)>>4;
        bit3 = (gray_value&0x08)>>3;
        bit2 = (gray_value&0x04)>>2;
        bit1 = (gray_value&0x02)>>1;
        bit0 = gray_value&0x01;
        bit4 = bit5&!bit4|bit4&!bit5;
        bit3 = bit4&!bit3|bit3&!bit4;
        bit2 = bit3&!bit2|bit2&!bit3;
        bit1 = bit2&!bit1|bit1&!bit2;
        bit0 = bit1&!bit0|bit0&!bit1;
        return 32*bit5+16*bit4+8*bit3+4*bit2+2*bit1+bit0;
}
```

Incremental Shaft Encoder (Quadrature Encoder)

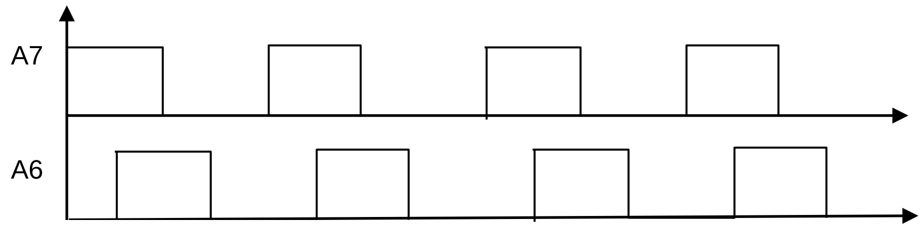
- less complex than the absolute encoder
- consists of a rotating disc having a track of transparent windows
- On one side of the disc, there is a light source, and on the other side of the disc there is a light sensor.
- outputs a voltage pulse every time a transparent window passes the light source
- electronic circuitry and sofware used to count the pulses, in order to determine the angle of rotation.

- called an incremental encoder, because the generation of a pulse indicates an incremental change in position, and not the actual (absolute) position.

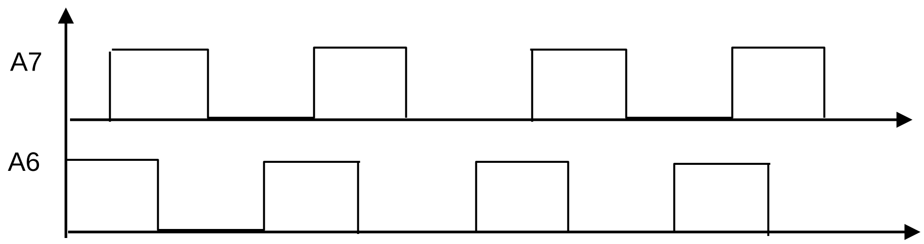
- reference or index pulse is required to determine initial absolute position.
- Initial position sometimes referred to as the “home” position
- to determine the direction of rotation, a second light sensor is required

- 2 sensors generate quadrature signals (signals 90 deg out of phase)
    - sensor 1 square wave output leading sensor 2 output by 90° indicates one direction of rotation;
    - sensor 1 output lagging sensor 2 output by 90° indicates the opposite direction of rotation.

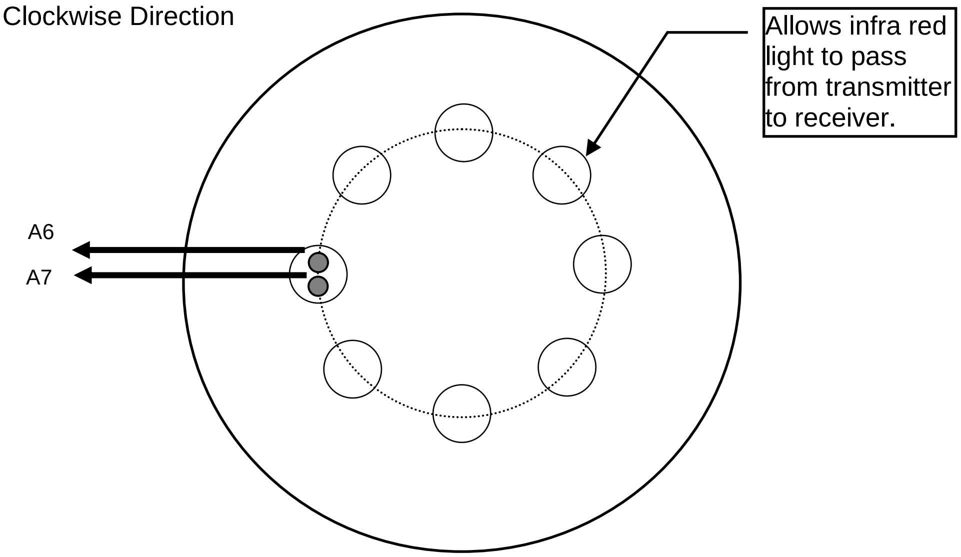
- For the Feedback unit, two photo sensors A6, and A7 are used to generate the quadrature signals
- signals could be displayed on an oscilloscope
- waveforms would resemble those shown below:



Clockwise Direction



Clockwise Direction



Servo Control

- servo control system is usually meant to make the position, velocity. or acceleration of some moving object accurately follow some sort of motion control command signal which frequently or constantly changes

- robot is basically a collection of one or more coordinated servo control systems
    - science of " robotics " largely involves consideration of servo control systems

- movements of the flaps and ailerons of an airplane are all controlled by servo systems
    - when an airplane flies in " auto-pilot " mode, the entire airplane is essentially operating as one very large collection of interacting computer coordinated servo control systems

Open Loop and Closed Loop Control

Descriptive Example no- 1 - Control of Temperature in Car or in House

Open Loop (Manual) Control

- Suppose you are driving along in an older inexpensive car in winter-time and the temperature inside the car feels uncomfortably cold ; what do you do ?
    - turn the interior heater on, and, if the temperature gets too warm
        - you can turn the heater off again, or
        - you can reduce its heat output to a lower level
- if, after a while, the temperature becomes too cold again, you. just turn the heater output back up again

- a human operator (*you*) necessary to make all decisions and perform all actions needed to control the interior car temperature to a comfortable level

- type of "control ", requiring continuous human operator interaction. is called oven-loop or manual control

Closed Loop (Automatic) Control

- " Automatic " able to occur without requiring human interaction

- house heating furnace is automatically turned on and off by a thermostat

- thermostat is a temperature sensing switch which automatically closes (turns furnace on) whenever the temperature goes slightly below the thermostat switching temperature value, and automatically opens (turns furnace off) whenever the temperature goes slightly above this value

- thermostat switching temperature value is adjustable

- actual house temperature can remain very close to a constant comfortable value with ***no*** human operator interaction

Feedback

- In the house furnace heating system thermostat provides closed loop control
    - produces a physical measure of the actual house temperature in a way which can be used to control the furnace automatically
        - switch open = house temperature > thermostat setting =:> furnace off
        - switch closed = house temperature < thermostat setting =:> furnace on
- action of using measurement of the actual value of the controlled variable (temperature in this example) in a way that can be used to implement automatic control is called automatic feedback

- Closed-loop control systems always have some form of automatic feedback; open-loop control does not have any automatic feedback

Descriptive Example -no. 2 - Control of Car Sneed

Open Loop (Manual) Control

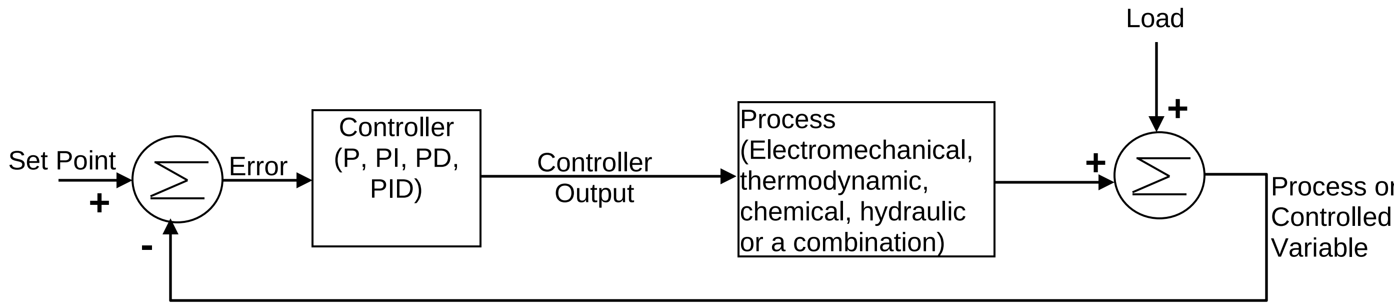
- maintain the speed of a car as close as possible to a constant 60 (km / hour)
- driver looks at the speedometer
    - if the speedometer reads less than 60, increase foot force on the gas pedal
    - if the speedometer reads more than 60, decrease force on the gas pedal
- human operator necessary to make all control decisions and perform all control actions
- no automatic speed sensing device providing a usable feedback action or output

Closed Loop (Automatic) Control

- calibrated control knob, mounted close to the steering wheel, allows a desired target speed (e.g., 60 km / h) to be dialed in
- adjusting the control knob actually adjusts an internal potentiometer which in turn provides a variable dc voltage whose value represents the desired target speed (set point or reference)
- dc voltage is one input to an internal electronic speed control circuit
- Another voltage that represents the actual car speed
- Generated by the speedometer
    - voltage is connected to a second input to the electronic speed control circuit
- speed control circuit compares the speedometer voltage with the target speed voltage.
- If speed voltage is less than the target speed voltage
    - electronic speed control circuit sends a command voltage to an actuator which increases the car engine throttle position and the car speeds up;
- If speed voltage is greater than the target speed voltage
    - electronic speed control circuit sends a command voltage to an actuator which decreases the car engine throttle position and the car slows down

- car travels along at a speed very close to the target speed dialed in on the control knob with no human operator action
- automatic feedback of the actual car speed is provided by using the speedometer voltage.

Control System Block Diagrams



- Process or Controlled Variable
    - physical process parameter that the control system is meant to control
        - Examples include, position, velocity, pressure, level, flow, temperature

- Set Point
    - physical quantity or measure that represents the desired target value of process variable/controlled variable

- Process Variable/Controlled Variable and Set Point must be physically of the same type

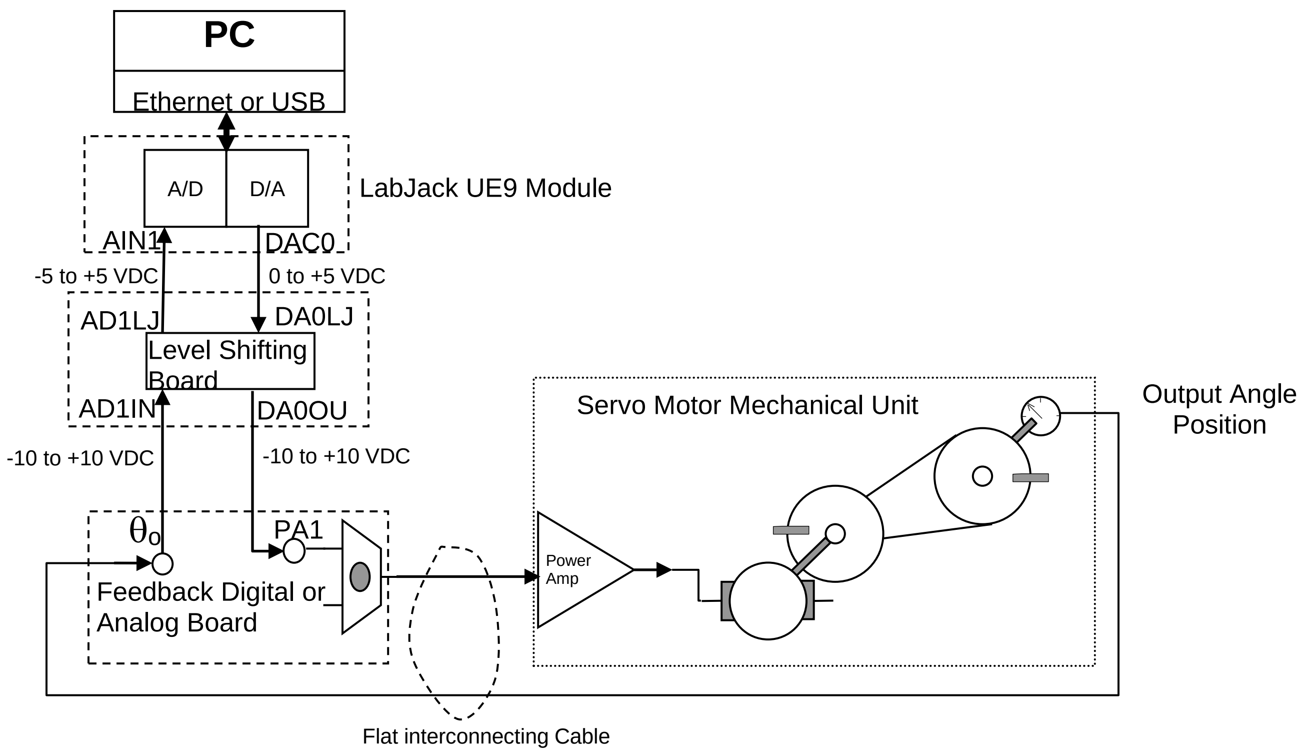
- Error
    - Set Point – Process Variable for a reverse- acting controller
    - Process Variable - Set Point for direct acting controller

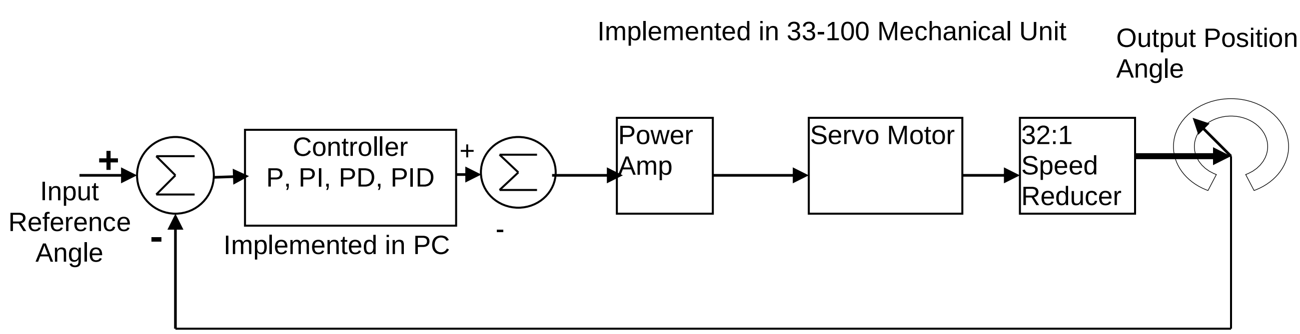
- perfect control system operation is achieved when error = 0

- Controller Output
    - command given to the final control element
    - depends on the value. of error and on the type of control performed by the controller

- Final Control Element
    - device that supplies energy or matter to the process in a form suitable to change CV such as to reduce error
- Process
    - physical configuration of interacting, dynamic components and parameters, one or more of which is controlled by the control system

Digital Position Control -- Using Output Potentiometer To Measure Position and PC To Input Reference Angle





The Digital PID Algorithm

$$C_{out} = K_P[e + \frac{1}{T_I}\int edt + T_d \frac{de}{dt}]$$

Proportional Control

K_p*(SP-PV)

For the software implementations in the lab:

- SP is normalized 0 to 1.0
- PV is normalized 0 to 1.0

- The Range in engineering units associated with the set point and the process or controlled variable must be the same. For example, if the range of the temperature sensor is 20 to 40 deg C then the set point is also 20 to 40 deg C.

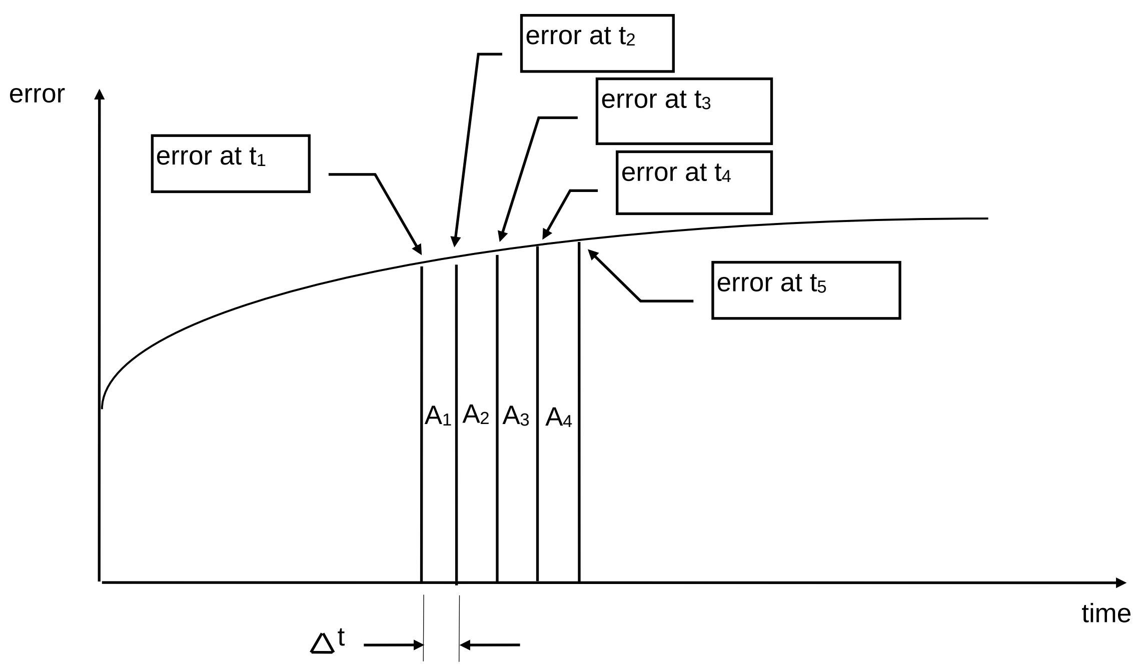
Proportional Component of PID Code:

- P = K_p*(SP - PV);

Integral Control

- Remember that integration involves finding the area under a curve. In our case we are integrating or finding the area under the error (SP-PV) curve.
- The independant variable is always time.
  - That is we are really integrating in real time.

- In digital control, integration is performed by taking a sample of the error at a given time and then a very short time later (the sampling time), taking another sample.



error at $t_1 = e(t_1)$
error at $t_2 = e(t_2)$
error at $t_3 = e(t_3)$
error at $t_4 = e(t_4)$
error at $t_5 = e(t_5)$

Approximate Areas (where Δt is small)
area $A_1 \approx e(t_1)\Delta t$
area $A_2 \approx e(t_2)\Delta t$
area $A_3 \approx e(t_3)\Delta t$
area $A_4 \approx e(t_4)\Delta t$

$A_{total} = A_1+A_2+A_3+A_4$
or

$$A_{total} = \sum_{i=1}^{4} e(t_i)\Delta t$$

or

$$A_{total} \cong \int_{t_1}^{t_4} e(t)dt$$

- In a computer program to implement integration, you successively sum the areas. That is:

$A_{new\ total} = A_{old\ total} + e(t)\,\Delta t$

- Sometimes we can represent this as

$A_{n+1} = A_n + e(t)\,\Delta t$

n+1 refers to the new sample
n is the sample prior to n+1
Δt is the sampling time

- in a computer program all we need for the equation is

$A = A + e^* \Delta t$

- For integral control the following expression can be used:

$$I = I + \frac{K_P}{T_i} eT$$

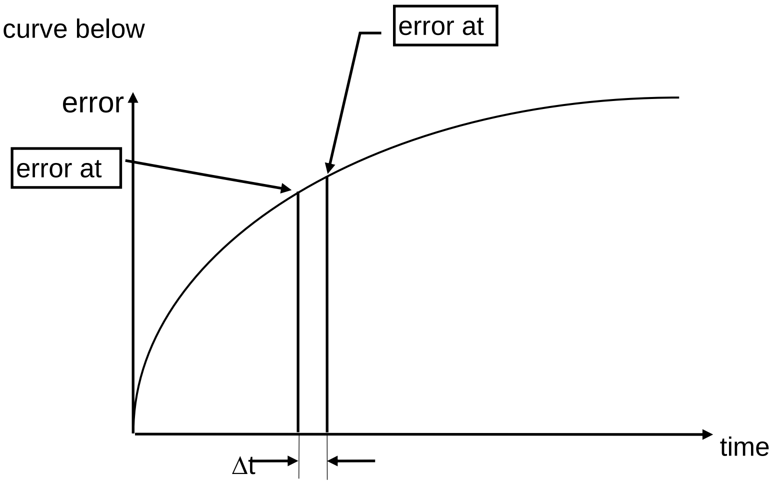
- where K_P is the proportional gain, T_i is the integral time for a PID algorithm. and T is the sampling time.

Differentiating Numerically

- The derivative of a variable e that varies as a function of time is

$\frac{de}{dt}$

- For the curve below



- When the Δt is small, the error curve over Δt is approximately linear. That is the slope over Δt is an approximation to $\frac{de}{dt}$.

- You can then represent the derivative of error as:

$$\frac{de}{dt} = \frac{e_n - e_{n-1}}{\Delta t}$$

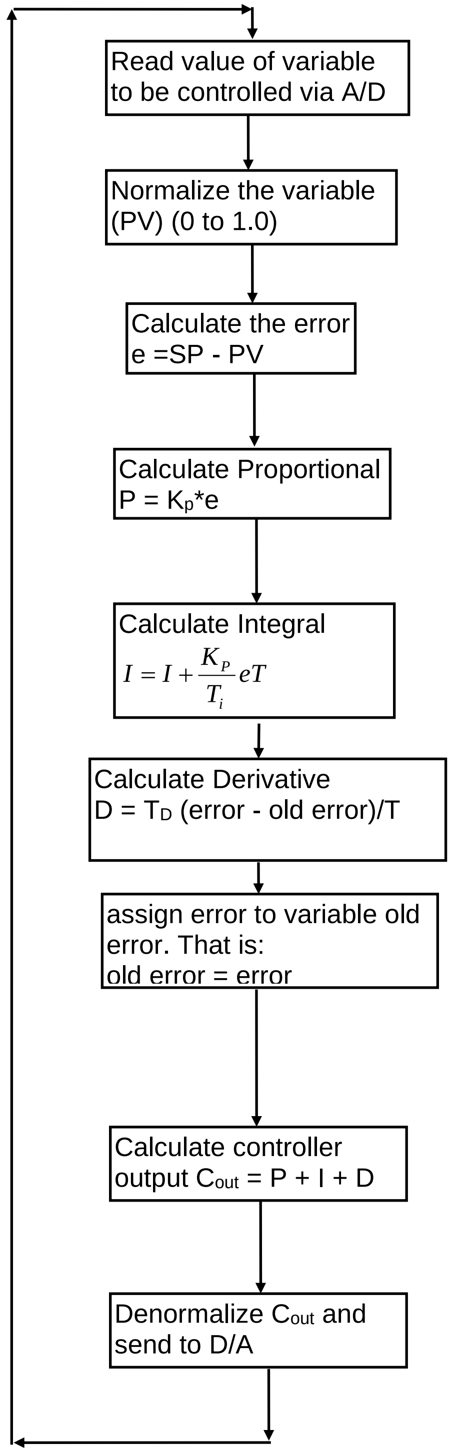
- where n represents the sample number of the error, and Δt the sampling time.

- In a computer you have to read the error and store it.
  - Then you have to read the error again and subtract the old error from it to find the change in error
  - You then divide the change in error by the sampling time to find the derivative. U
- sing pseudo code, the derivative can be expressed as:

<u>Pseudo Code For Derivative of Error</u>

- *get current error;*
- *get difference in error by subtracting old error from current error;*
- *derivative of error equals the error difference divided by the sampling time;*
- *set old error equals current error;*
- *do the above continuously;*

PID Flow Chart



Sample Time

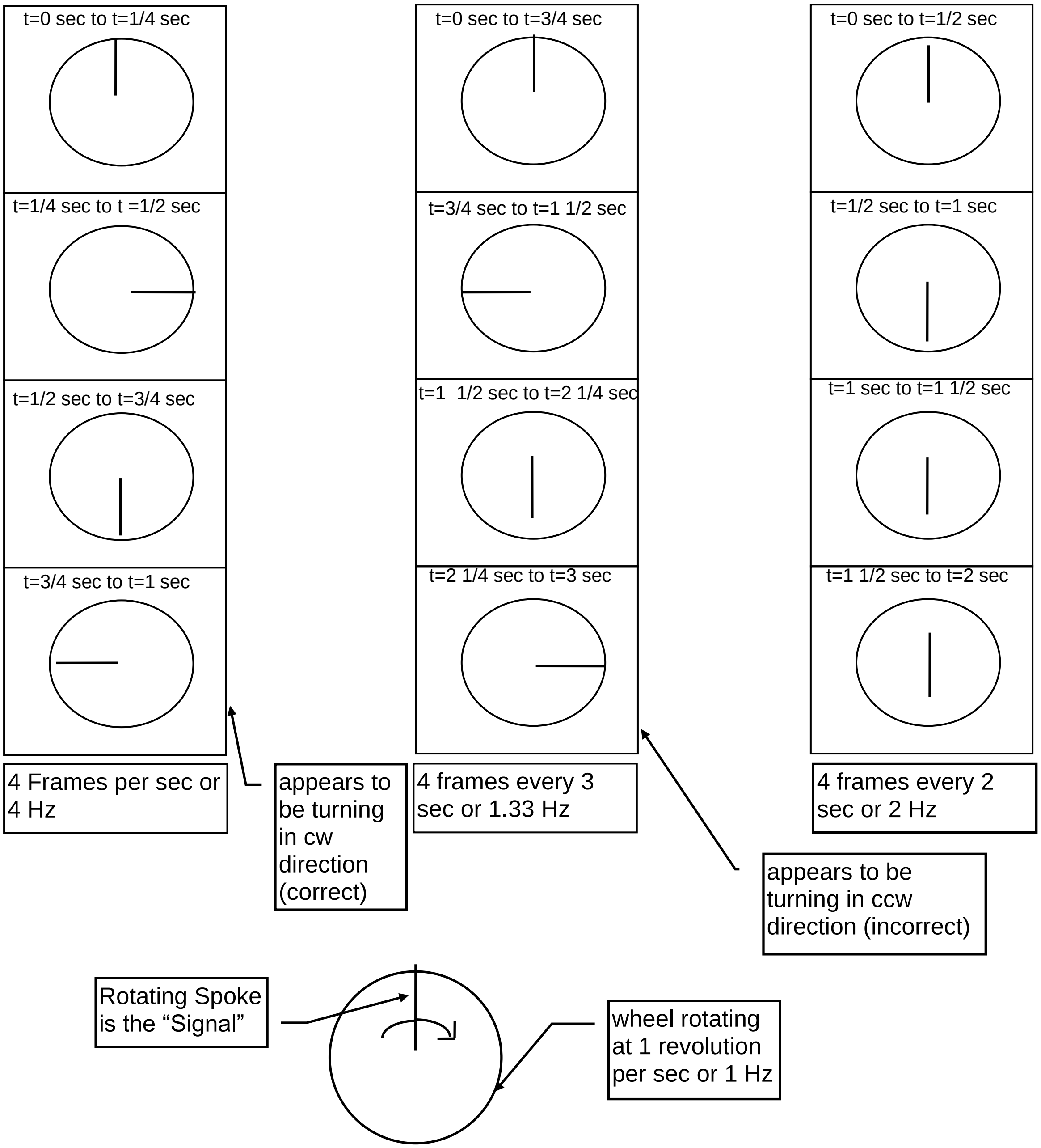
- PC samples input data at regular intervals using the A/D converter
- PC processes the data performing calculations within a fixed time period
- PC outputs new values via the D/A converter.

- Total time to perform all of the above called the **Sample Period T** or the **Sample Time**.

- Inverse of sample time is the **Sample Frequency** or **Sample Rate**.

- Sampling too slowly degrades control system performance and could result in aliasing causing control system to act erratically

Finding the maximum sampling time (minimum sampling frequency)



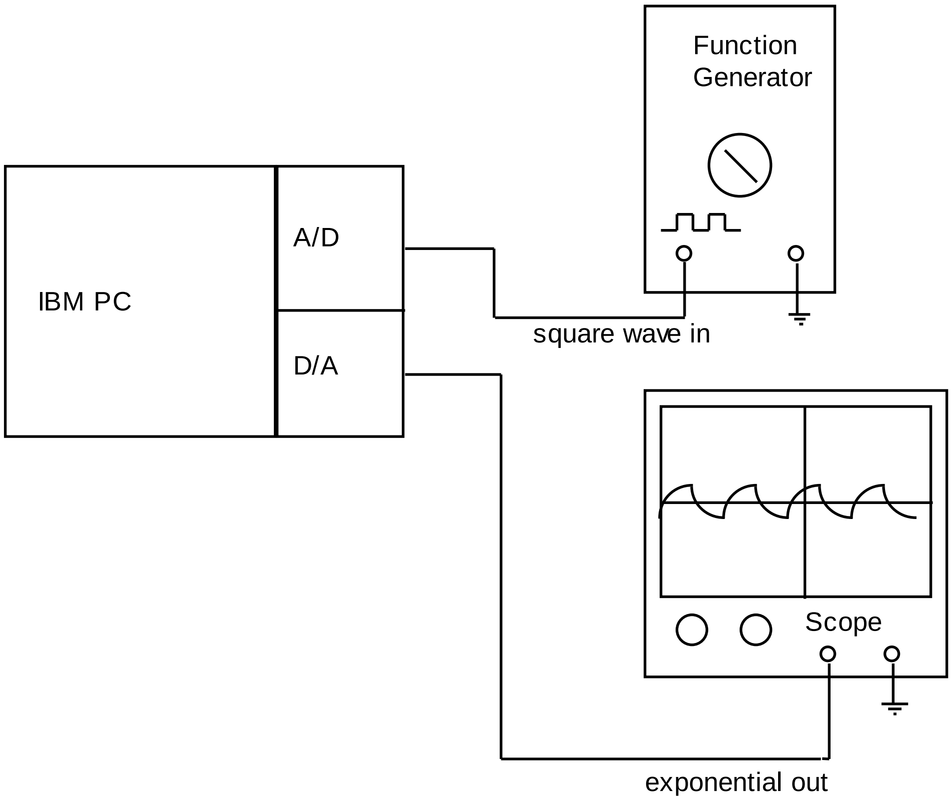
- When frame rate (sampling frequency) is greater than two times spoke revolutions per sec (signal frequency), frames display a correct rotation (clockwise)
- When frame rate (sampling frequency) is less than two times spoke revolutions per sec (signal frequency), frames display an incorrect rotation (ccw).

Nyquist Criteria: Sampling frequency must be greater than 2 times highest frequency component of signal being sampled.

- A very important principle – Note that for compact discs the sampling rate is approximately 44KHz (sound frequencies are less than about 20 KHz)

In control systems, sometimes noisy signals need to be filtered

Digital Low Pass Filter Using IBM PC




```
/* low pass filter */
float low_pass( float value_0, float tau, T_samp )
{
static float filt_out;

filt_out = filt_out + (T_samp/(tau+T_samp))(value_0 - filt_out);
return filt_out;
}

/* build delay into loop using system clock */
void delay(float duration)
{
static time_t time1, time0;
                do
                {
                time1=clock();
                }
                while(time1-time0<=CLOCKS_PER_SEC*duration);
                time0=clock();
        }
}
```

Note that too much filtering adds a lag to the process and results in poorer control system performance.