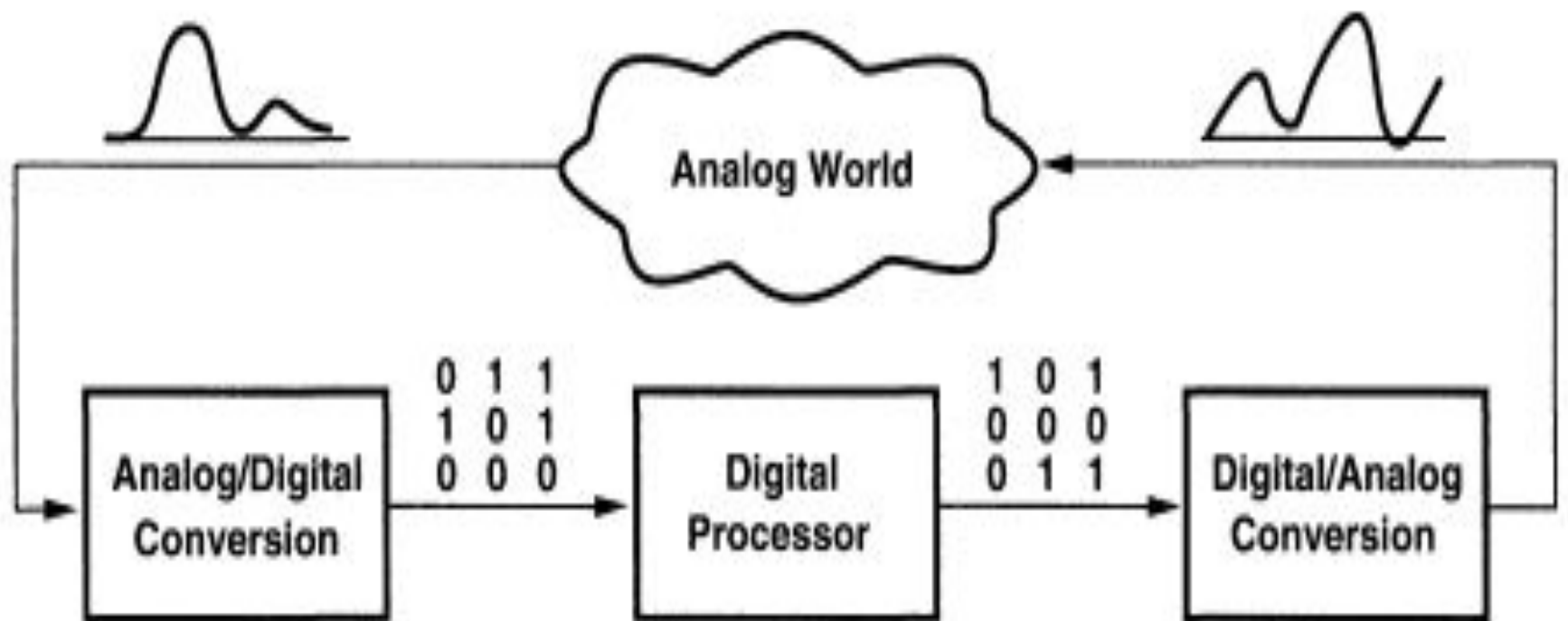


# **Interfacing Data Conversion**

# Data Conversion System

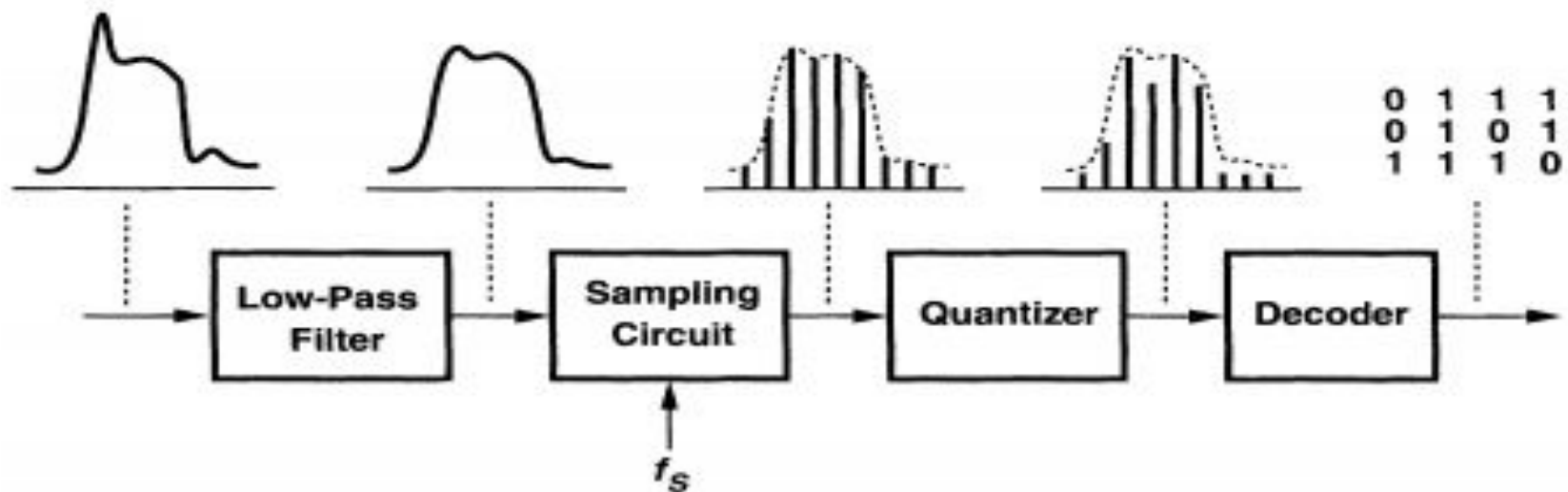
- **Data Conversion** is the process of changing or converting one form of data in to another form. In processing and communication there are only two types of data forms i.e analog and digital data.
- Real world signals are analog signals continuous time & amplitude (temp, pressure, position, sound, light, speed, etc.).
- Digital processors can only process digital signals which are discrete time and discrete amplitude.

- In order to interface digital processors with the analog world, data acquisition and reconstruction circuits must be used: analog-to-digital converters (ADCs) to acquire and digitize the analog signal at the front end, and digital-to-analog converters (DACs) to reproduce the analog signal at the back end.
- [?] Digital data conversion system requires ADC and DAC.



# Analog-to-Digital Converter (ADC)

- The analog-to-digital converter (ADC) converts a continuous-amplitude, continuous-time input to a discrete-amplitude, discrete-time signal.



1. An analog low-pass (anti-alias) filter limits the input signal bandwidth so that subsequent sampling does not alias any unwanted noise or signal components into the actual signal band.
2. The filter output signal is sampled at a rate of  $f_s$  samples per second to produce a discrete-time signal.
3. The amplitude of this discrete-time signal is "quantized," i.e., approximated with a level from a set of fixed references, thus generating a discrete-amplitude signal.
4. The discrete-amplitude signal is decoded to a digital representation of that level is established at the output.

# ADC Applications

- ADC are used virtually everywhere an analog signal has to be processed, stored, or transported in digital form
  - Microphones
  - Strain Gages
  - Thermocouple
  - Voltmeters
  - Digital Multimeters

# A/D Converter Types

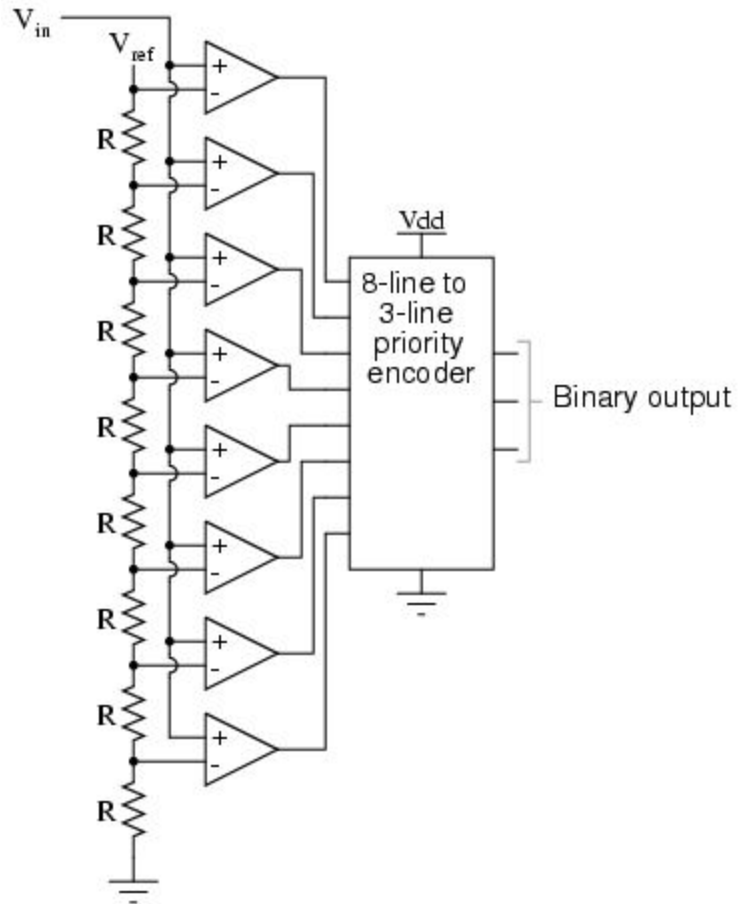
- Flash ADC
- Delta-Sigma ADC
- Dual Slope (integrating) ADC
- Successive Approximation ADC



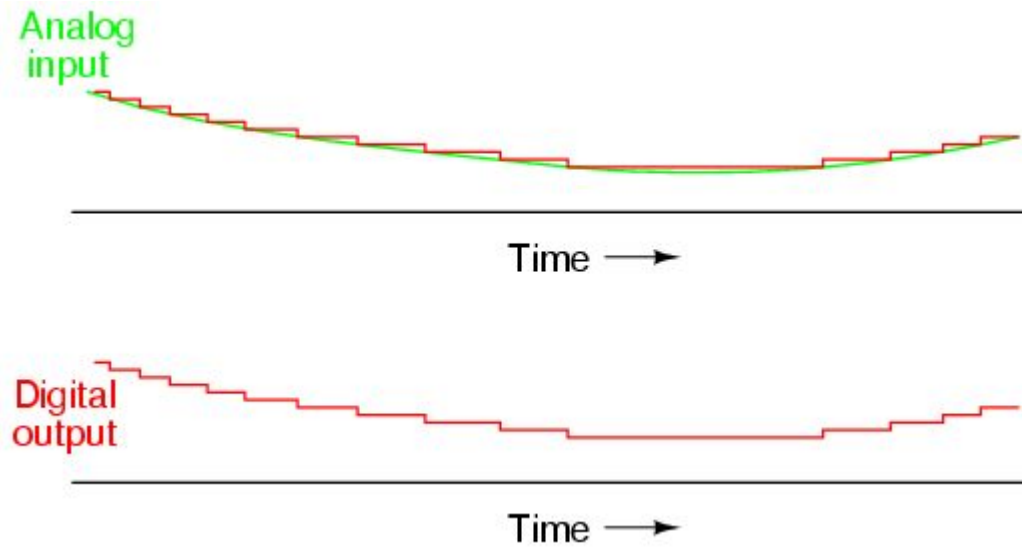
# Flash ADC

- Consists of a series of comparators, each one comparing the input signal to a unique reference voltage.
- The comparator outputs connect to the inputs of a priority encoder circuit, which produces a binary output

# Flash ADC Circuit



# ADC Output



# Flash

## Advantages

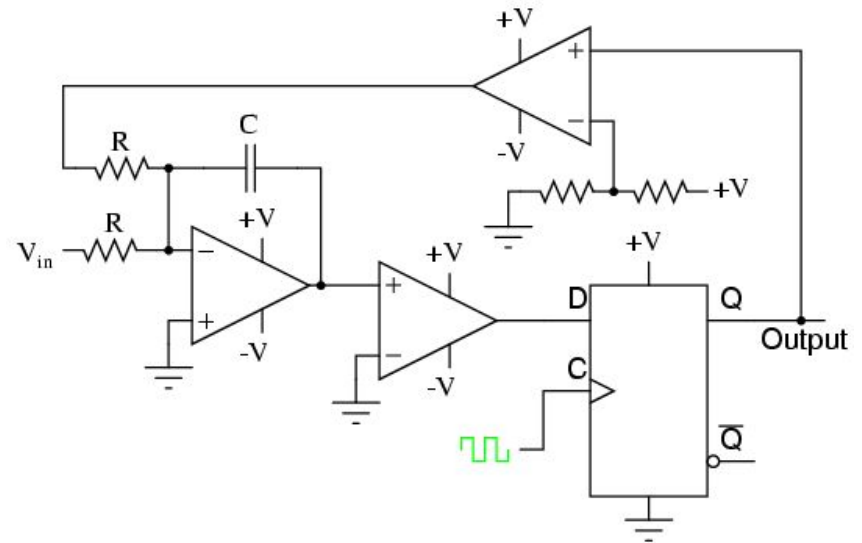
- Simplest in terms of operational theory
- Most efficient in terms of speed, very fast
  - limited only in terms of comparator and gate propagation delays

## Disadvantages

- Lower resolution
- Expensive
- For each additional output bit, the number of comparators is doubled
  - i.e. for 8 bits, 256 comparators needed

# Sigma Delta ADC

- Over sampled input signal goes to the integrator
- Output of integration is compared to GND
- Iterates to produce a serial bit stream
- Output is serial bit stream with # of 1's proportional to  $V_{in}$



# Sigma-Delta

## Advantages

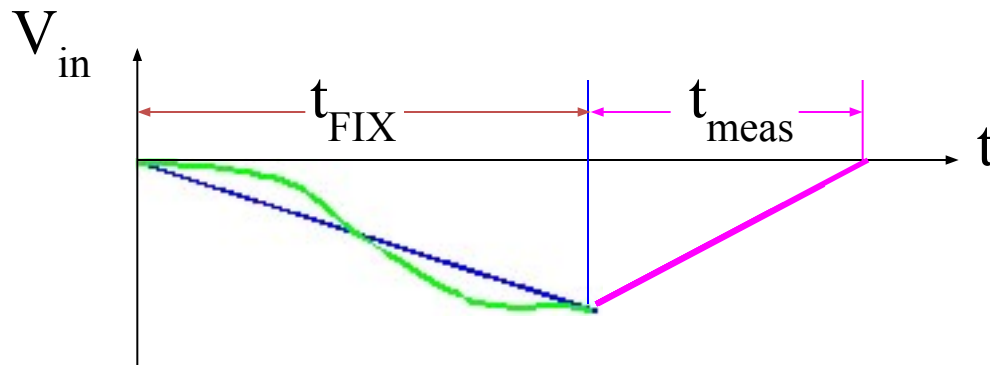
- High resolution
- No precision external components needed

## Disadvantages

- Slow due to oversampling

# Dual Slope Converter

- The sampled signal charges a capacitor for a fixed amount of time
- By integrating over time, noise integrates out of the conversion
- Then the ADC discharges the capacitor at a fixed rate with the counter counts the ADC's output bits. A longer discharge time results in a higher count



# Dual Slope Converter

## Advantages

- Input signal is averaged
- Greater noise immunity than other ADC types
- High accuracy

## Disadvantages

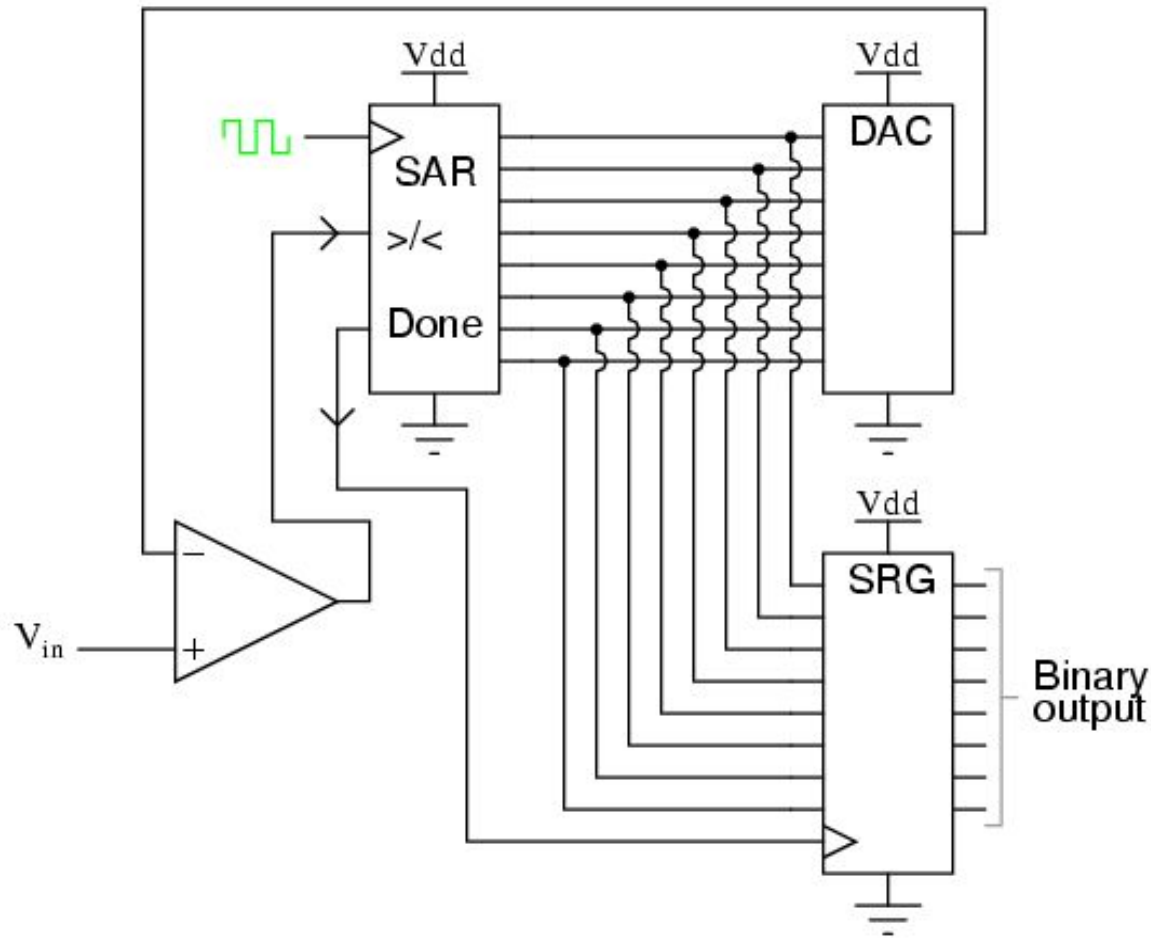
- Slow
- High precision external components required to achieve accuracy



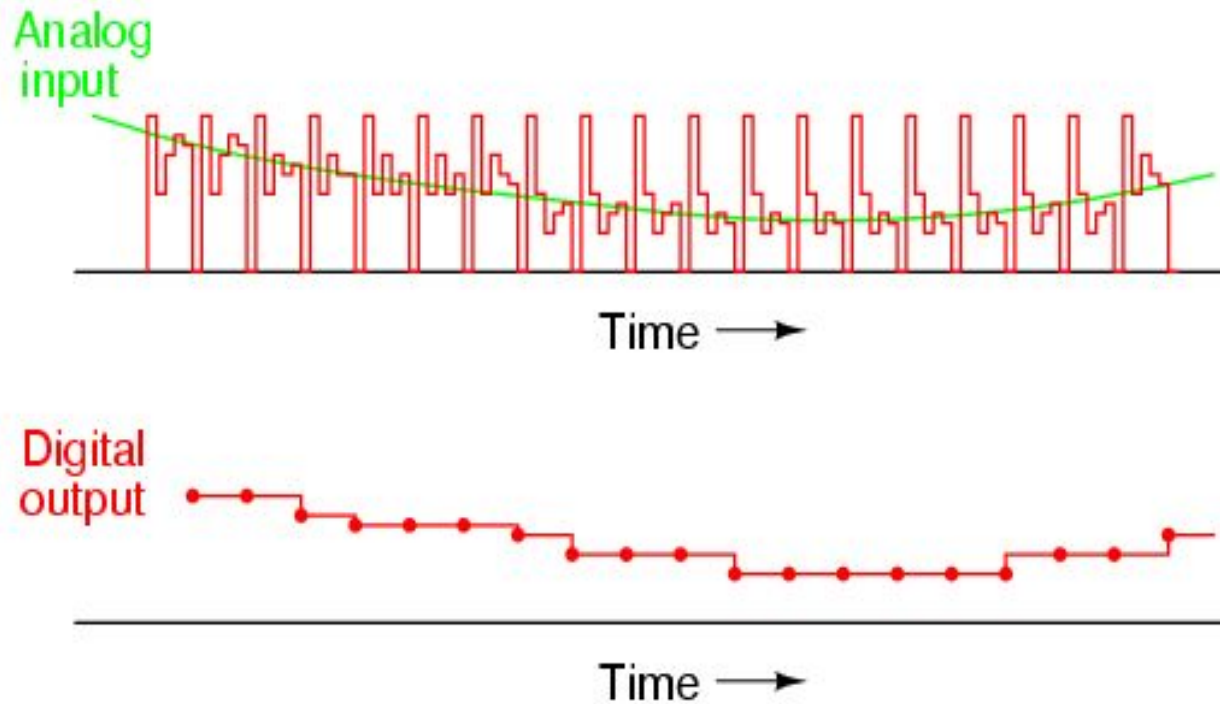
# Successive Approximation ADC

- A Successive Approximation Register (SAR) is added to the circuit
- Instead of counting up in binary sequence, this register counts by trying all values of bits starting with the MSB and finishing at the LSB.
- The register monitors the comparators output to see if the binary count is greater or less than the analog signal input and adjusts the bits accordingly

# Successive Approximation ADC Circuit



# ADC Output



# Successive Approximation

## Advantages

- Capable of high speed and reliable
- Medium accuracy compared to other ADC types
- Good tradeoff between speed and cost
- Capable of outputting the binary number in serial (one bit at a time) format.

## Disadvantages

- Higher resolution successive approximation ADC's will be slower
- Speed limited to ~5Msps

# ADC Types Comparison

| Type             | Speed (relative) | Cost (relative) |
|------------------|------------------|-----------------|
| Dual Slope       | Slow             | Med             |
| Flash            | Very Fast        | High            |
| Successive Appox | Medium – Fast    | Low             |
| Sigma-Delta      | Slow             | Low             |

# AD570

- The AD570 is an 8-bit successive approximation A/D converter consisting of a DAC, voltage reference, clock, comparator, successive approximation register and output buffers---all fabricated on a single chip.
- The AD570 incorporates the most advanced integrated circuit design and processing technology available today.

# ADC 0801

- The ADC 0801 is a 8-bit successive-approximation A/D converter.

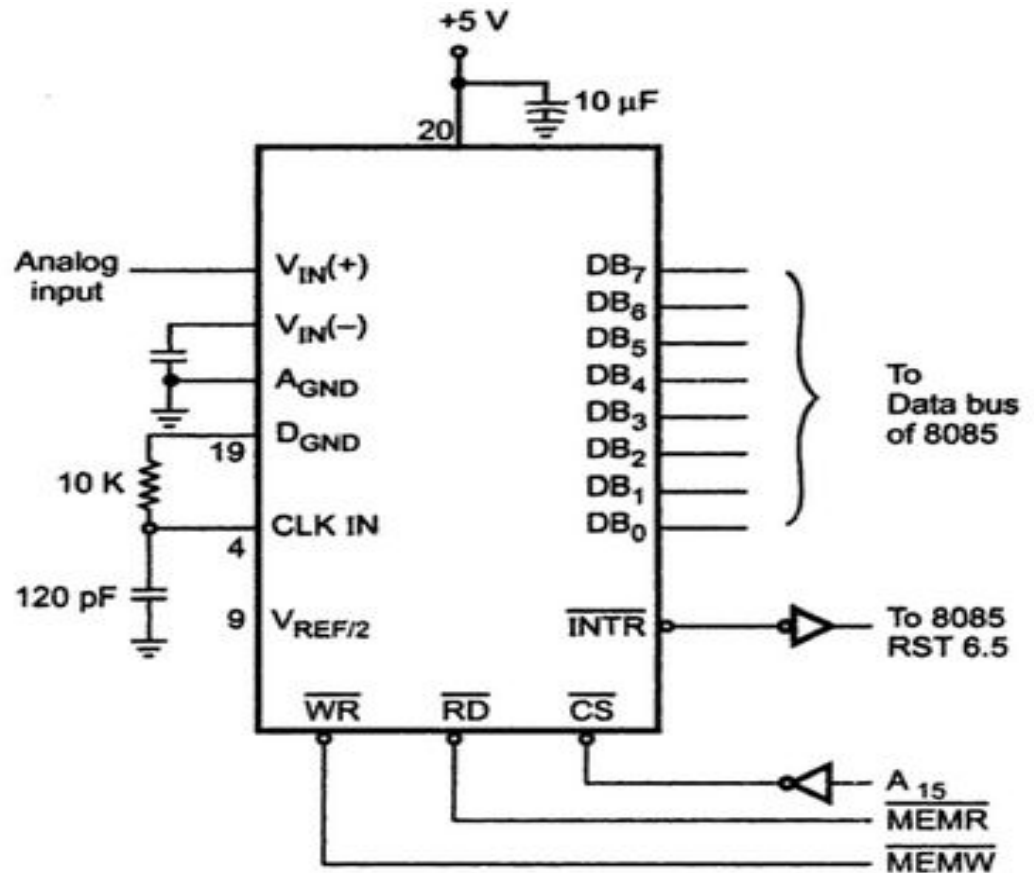


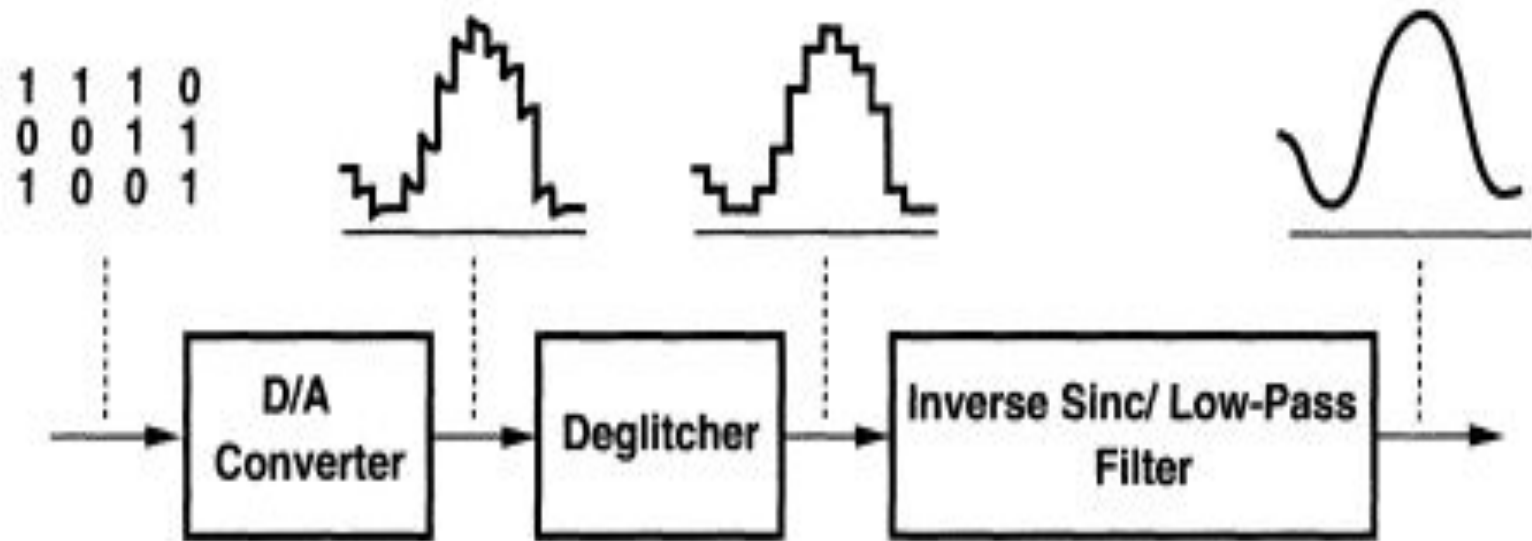
Fig. 1 (a) Interfacing of **ADC 0801** with 8085

# Digital-to-Analog Converter (DAC)

- The digital data is processed by a digital processor and output to the digital-to analog converter (DAC).
- The DAC reverses the ADC process, it converts a discrete time, discrete amplitude into continuous time, continuous amplitude. The DAC is usually operated at same frequency as the ADC.



1. A DAC selects and produces an analog level from a set of fixed references according to the digital input.
2. If the DAC generates large glitches during switching from one code to another, a "deglitching" circuit (usually a sample-and-hold amplifier) follows to mask the glitches.
3. The reconstruction function performed by the DAC introduces sharp edges in the waveform as well as a sine envelope in the frequency domain, an inverse-sine filter and a low-pass filter are required to suppress these effects. The resulting staircase-like signal is finally passed through a smoothing filter to ease the effects of quantization noise.



# DAC Applications

- Digital Motor Control
- Computer Printers
- Sound Equipment (e.g. CD/MP3 Players, etc.)
- Electronic Cruise Control
- Digital Thermostat

# D/A Converter Types

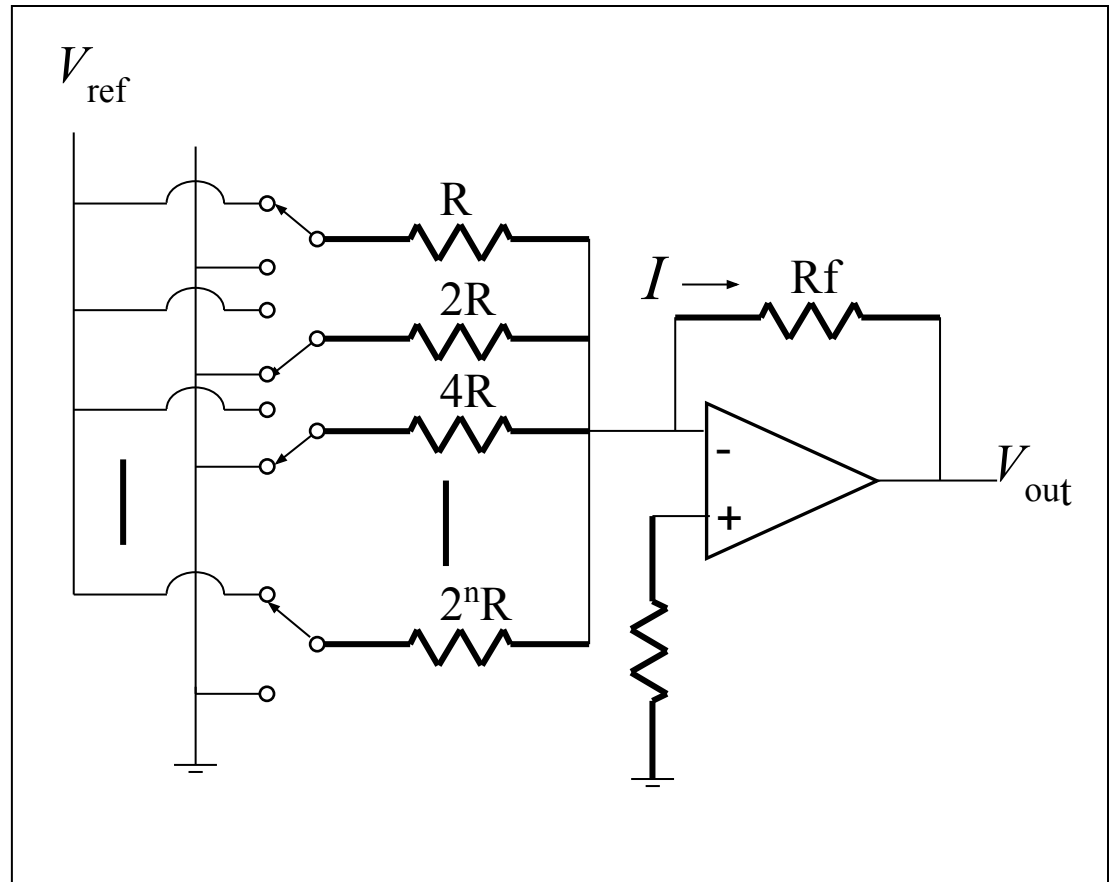
- Many types of DACs available.
- Usually switches, resistors, and op-amps used to implement conversion
- Two Types:
  - Binary Weighted Resistor
  - R-2R Ladder

# Binary Weighted Resistor

- Utilizes a summing op-amp circuit
- Weighted resistors are used to distinguish each bit from the most significant to the least significant
- Transistors are used to switch between  $V_{\text{ref}}$  and ground (bit high or low)

# Binary Weighted Resistor (Cont.)

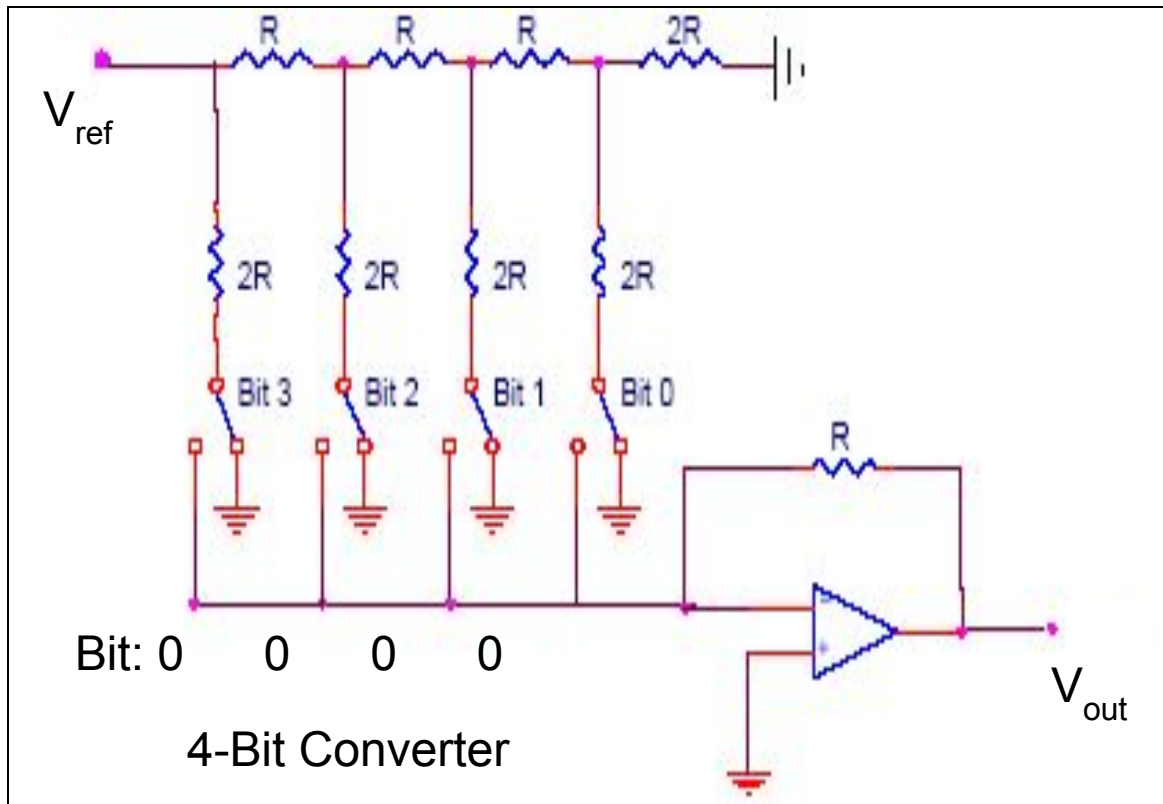
- Assume Ideal Op-amp
- No current into op-amp
- Virtual ground at inverting input
- $V_{\text{out}} = -IR_f$



# Binary Weighted Resistor (Cont.)

- Advantages
  - Simple Construction/Analysis
  - Fast Conversion
- Disadvantages
  - Requires large range of resistors (2000:1 for 12-bit DAC) with necessary high precision for low resistors
  - Requires low switch resistances in transistors
  - Can be expensive. Therefore, usually limited to 8-bit resolution.

# R-2R Ladder



Each bit corresponds to a switch:

If the bit is high, the corresponding switch is connected to the inverting input of the op-amp.

If the bit is low, the corresponding switch is connected to ground.



# R-2R Ladder (Cont.)

- Advantages
  - Only two resistor values ( $R$  and  $2R$ )
  - Does not require high precision resistors
- Disadvantage
  - Lower conversion speed than binary weighted DAC

# AD 7522

- The AD 7522 is a systems-compatible 10-bit multiplying D/A converter, fabricated on a single 3 x 2.2mm (118 x 89 mil) silicon die, and packaged in a 28-pin plastic or ceramic dual in-line package.

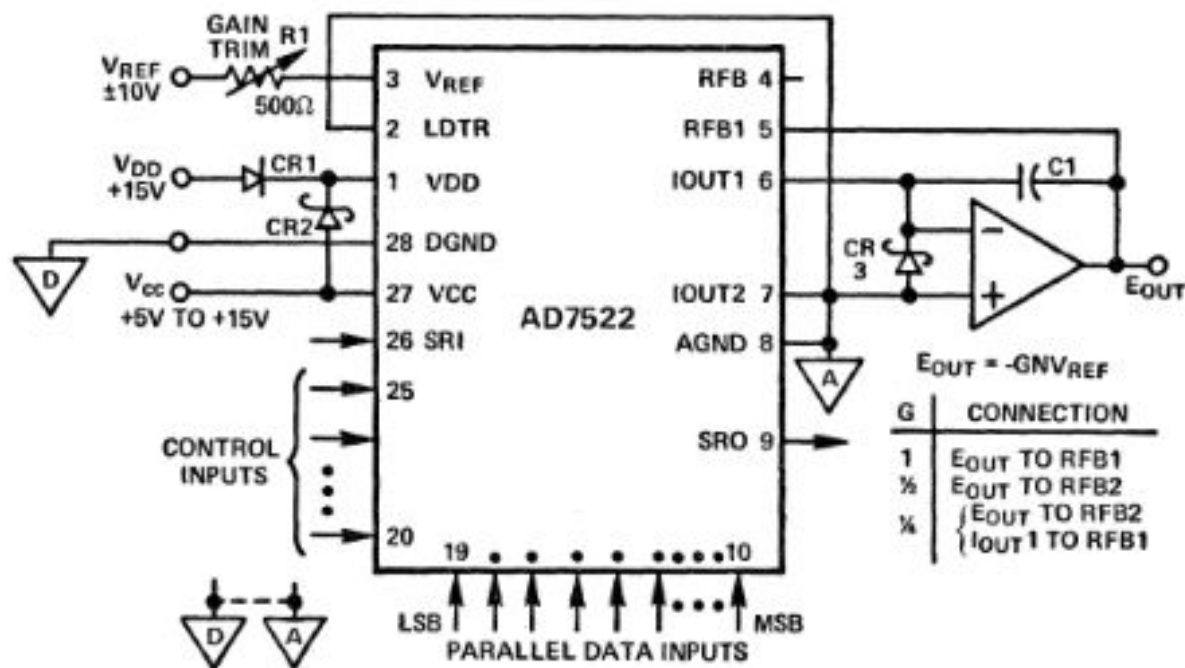


Figure 24. Connecting the AD7522 for unipolar D/A conversion.  $N$  is a fractional 10-bit binary number from 0 to  $(1 - 2^{-10})$ .