Introduction to Data Conversion
Part I

This lecture note was adapted from the following materials:

1) Fast and Effective Embedded System Design Applying the ARM mbed by Rob Toulson and Tim Wilmshurt
2) LPC1769 User Manual by NXP
3) Fundamental of Electric Circuits by C. Alexander and M. Sadiku

Outline

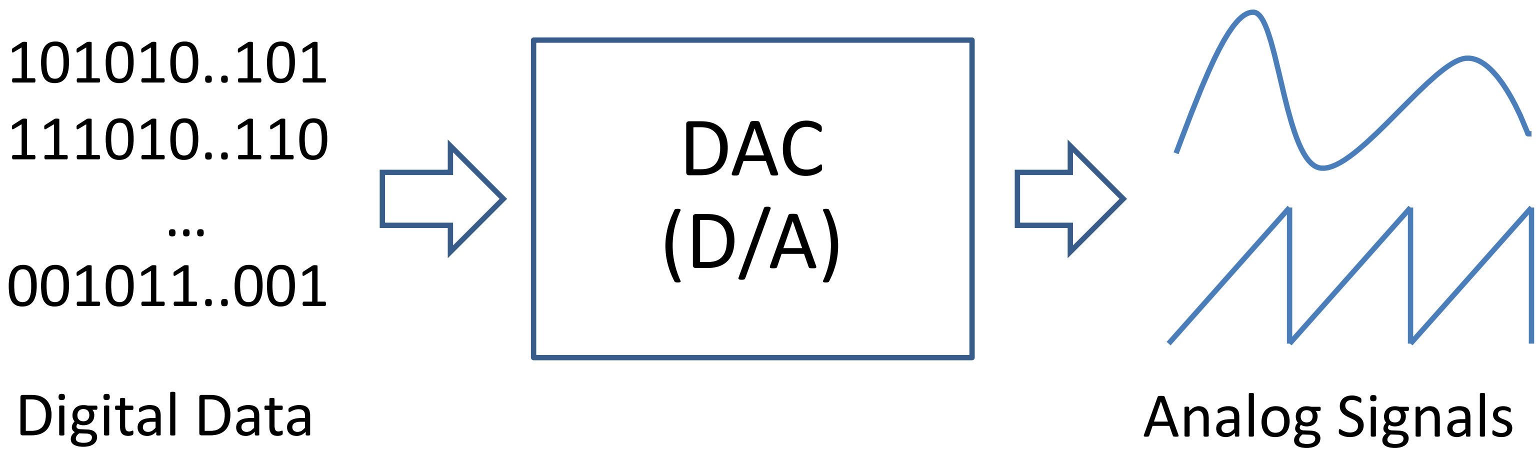
- Digital to Analog Converter
- Analog to Digital Converter
- DAC and ADC of LPC1769
- Signal Conditioning Circuit

Data Conversion

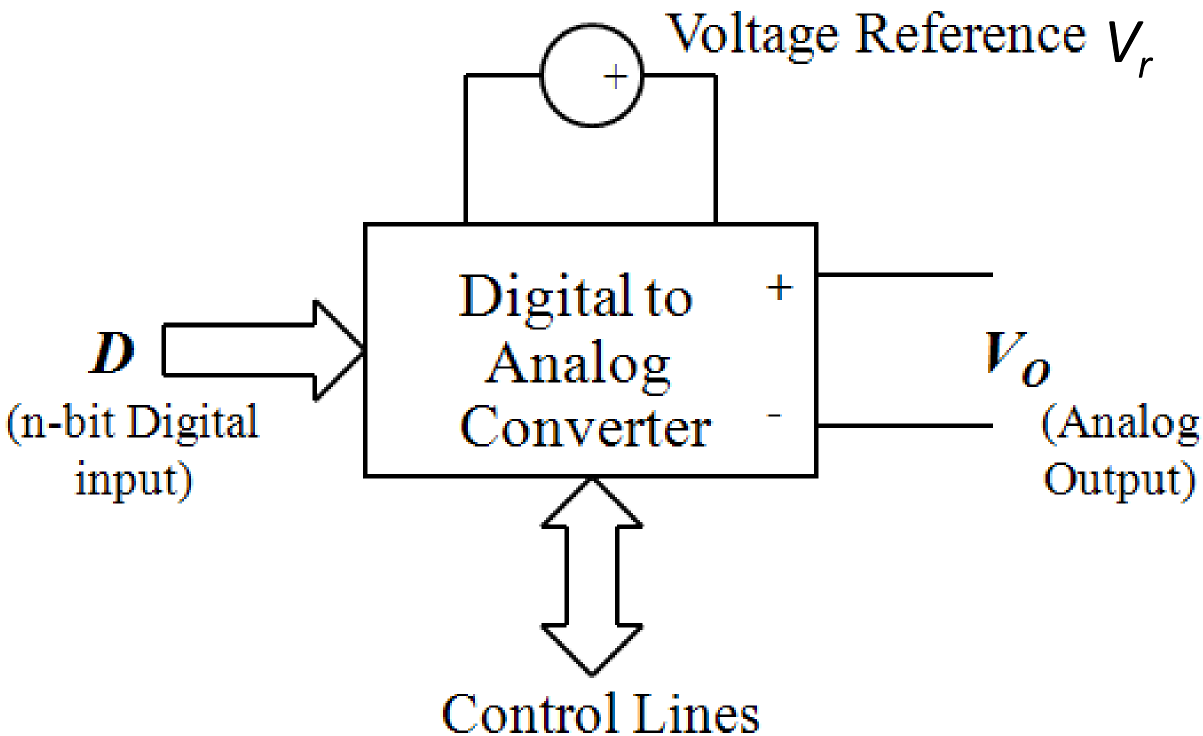
- What is data conversion ?
  - What does Analog signal look like ?
  - What does Digital signal look like ?
- Why do we need data conversion?
  - Real world is the world of analog
    - By nature, signals are in analog forms
    - We are able to perceive signals in analog form
  - Computer/Digital processor process data in digital domain
- Type of Data conversion
  - Analog to digital conversion
  - Digital to analog conversion

Digital-to-Analog Converter (DAC)

- A circuit that converts a binary input number into an analog output voltage which is proportional to that input number.
- DACs are widely used to create continuously varying signals, for example to generate analog waveforms



Model of DAC



$$V_o = \frac{D}{2^n} V_r$$

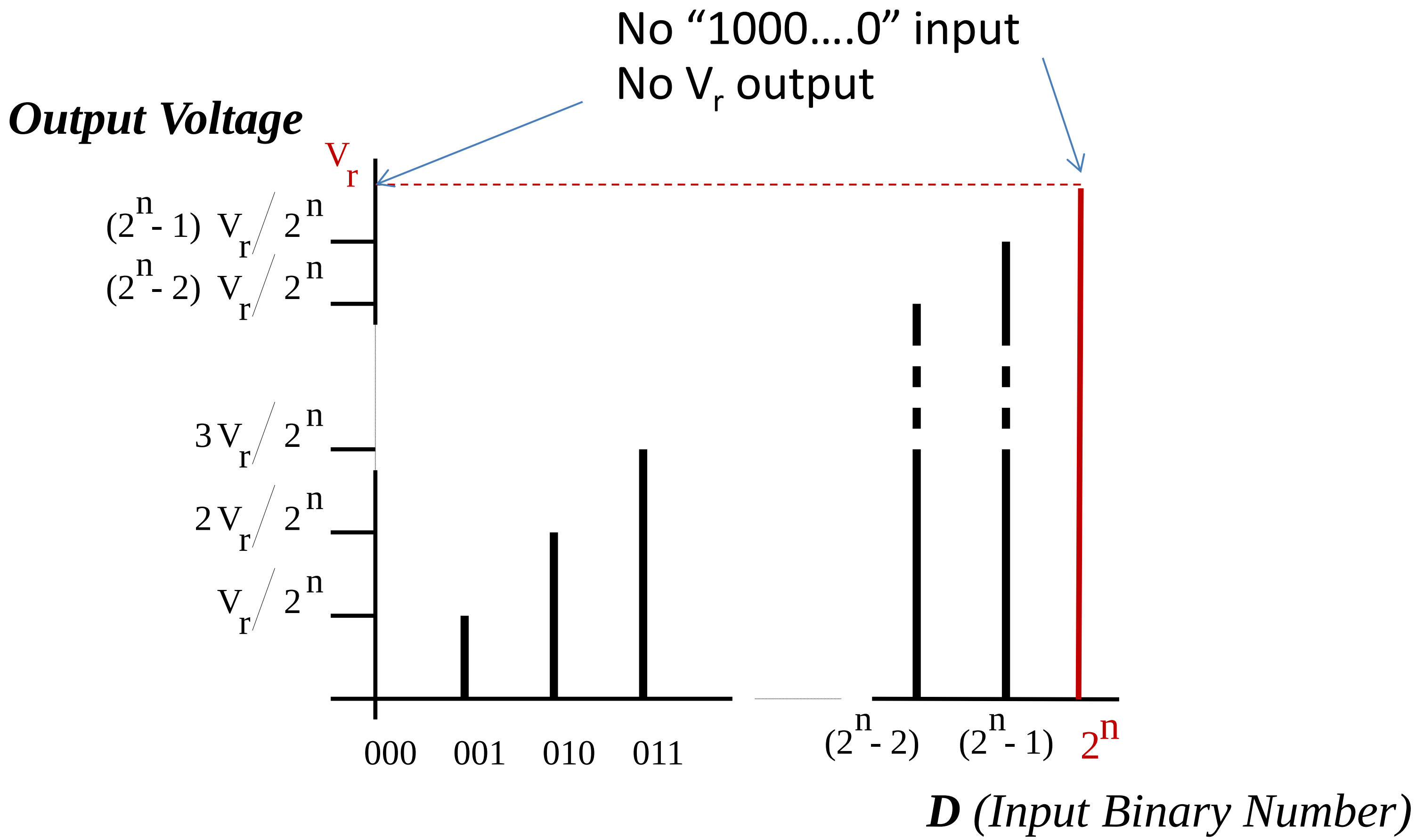
Parameters of DAC

$$\boxed{V_o = D\frac{V_r}{2^n} = D \cdot (resolution)}$$

V_o = Analog output that DAC can create
V_r = Reference voltage for using in conversion
D = Digital input word
n = Number of bit of DAC

- For each input digital value, there is a corresponding analog output.
- The number of possible output values is given by 2^n
- The step size is $V_r/2^n$ and called the ***resolution***.
- The difference between its maximum and minimum output values is called ***range***

DAC Input-Output Characteristic



DAC Input-Output Characteristic

- Example A 6-bit DAC, using reference voltage $V_r = 3.2$ Volt
  - There are 2^6 steps = 64 levels in its output analog
  - Resolution = 3.2V/ 64 = 50 mV = LSB

    000000 = 0 Volt = (0)(50 mV) This is the ***minimum***

    000001 = 0.050 V

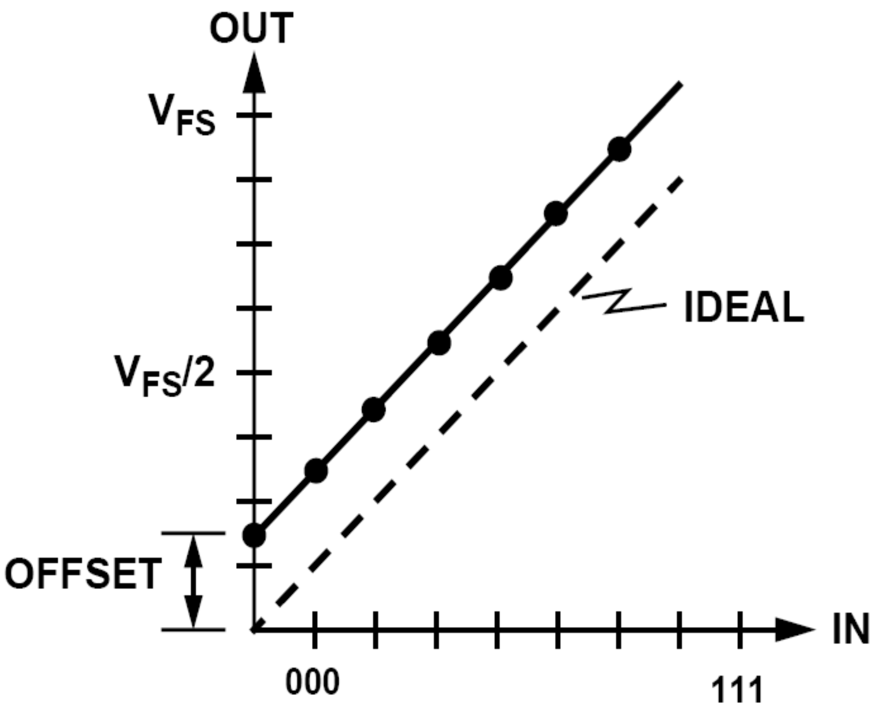
    000010 = 0.100 V

    …

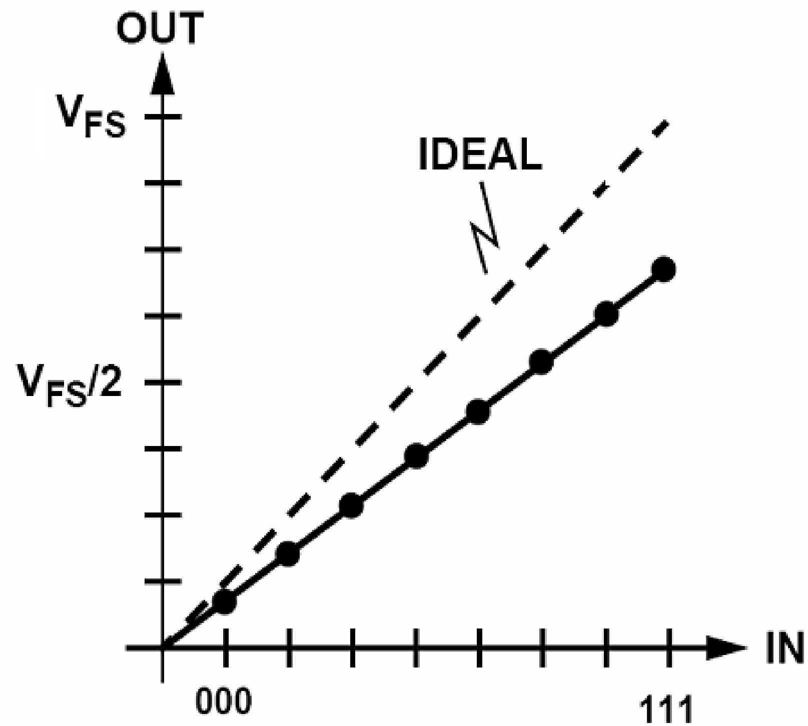
    111111 = 3.150 V = $(2^6 - 1)$(50 mV) This is the ***maximum***
  - Range is 3.15 – 0 = 3.15 Volt

Non ideal DAC

Examples of non-ideal DAC



Offset error

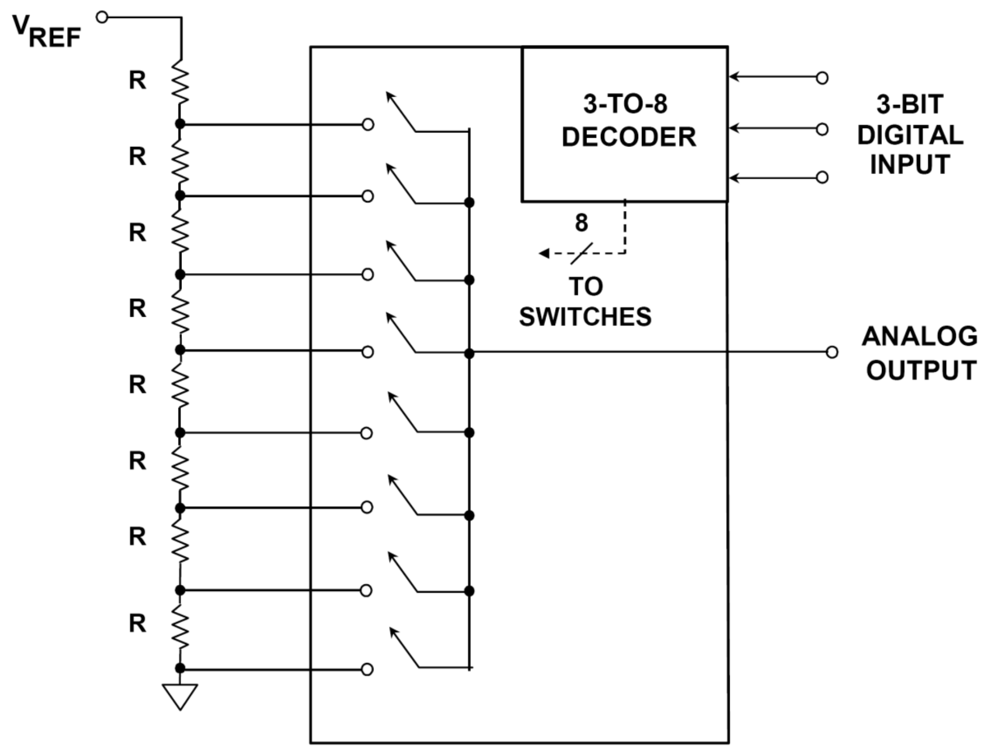


Gain error

LPC1769: DAC Architecture

- Using 10-bit Register String

  Example 3-bit Resister String DAC

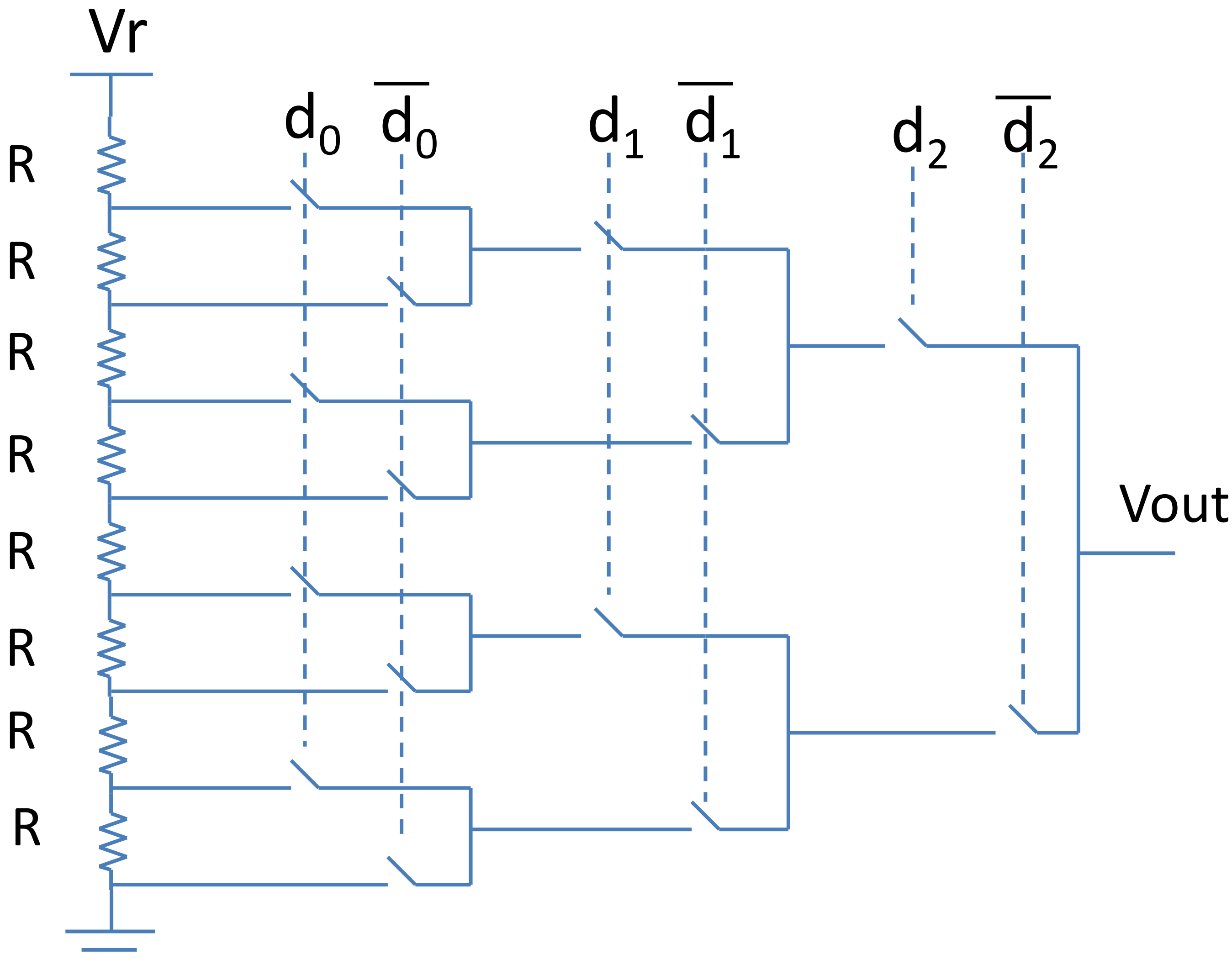


Simplest Voltage-Output Thermometer DAC:
The Kelvin Divider ("String DAC")

Source: Analog Devices

LPC1769 DAC Architecture

- Another 3-bit Resister String DAC



Analog-to-Digital Converter (ADC)

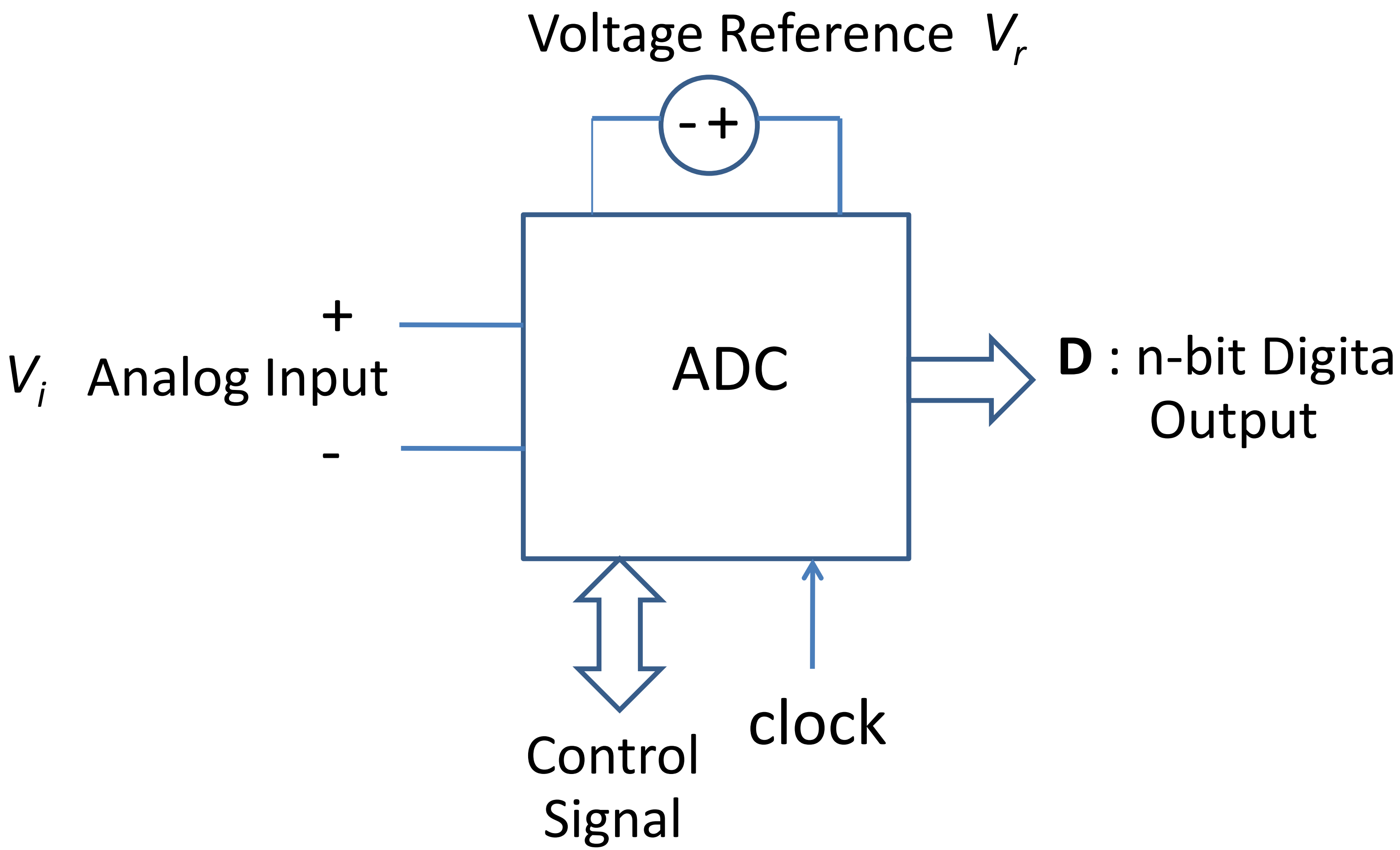
Why ADC ?

- The world around the embedded system is a largely analog one
  - Temperature, sound (audio), image (video), acceleration, light, magnetic field, electric field, etc
- When the embedded system want to use this quantities, it needs sensors:
  - Sensors change these physical quantities in an <u>electrical signal</u> and some in digital data but mostly in the form of <u>analog voltage</u>
- Because MCUs process data in digital form, we need ADC to convert those analog voltage into digital data

Analog-to-Digital Converter (ADC)

- An ADC is an electronic circuit whose digital output is proportional to its analog input.
  - It 'measures' the input voltage and gives a binary output number proportional to its size.
- There are many types of analog inputs depending on its sources and its applications
  - From different applications such as medical, industrial, entertainment, etc.
  - Difference in their frequency and amplitude
- As a result, many types of ADC have been developed, with characteristics optimized for these differing applications.

Model of ADC


Voltage Reference V_r
- +
+
V_i Analog Input
-
ADC
D : n-bit Digital Output
Control Signal
clock

Parameters of ADC

$$\boxed{D = \frac{V_i}{V_r} 2^n}$$

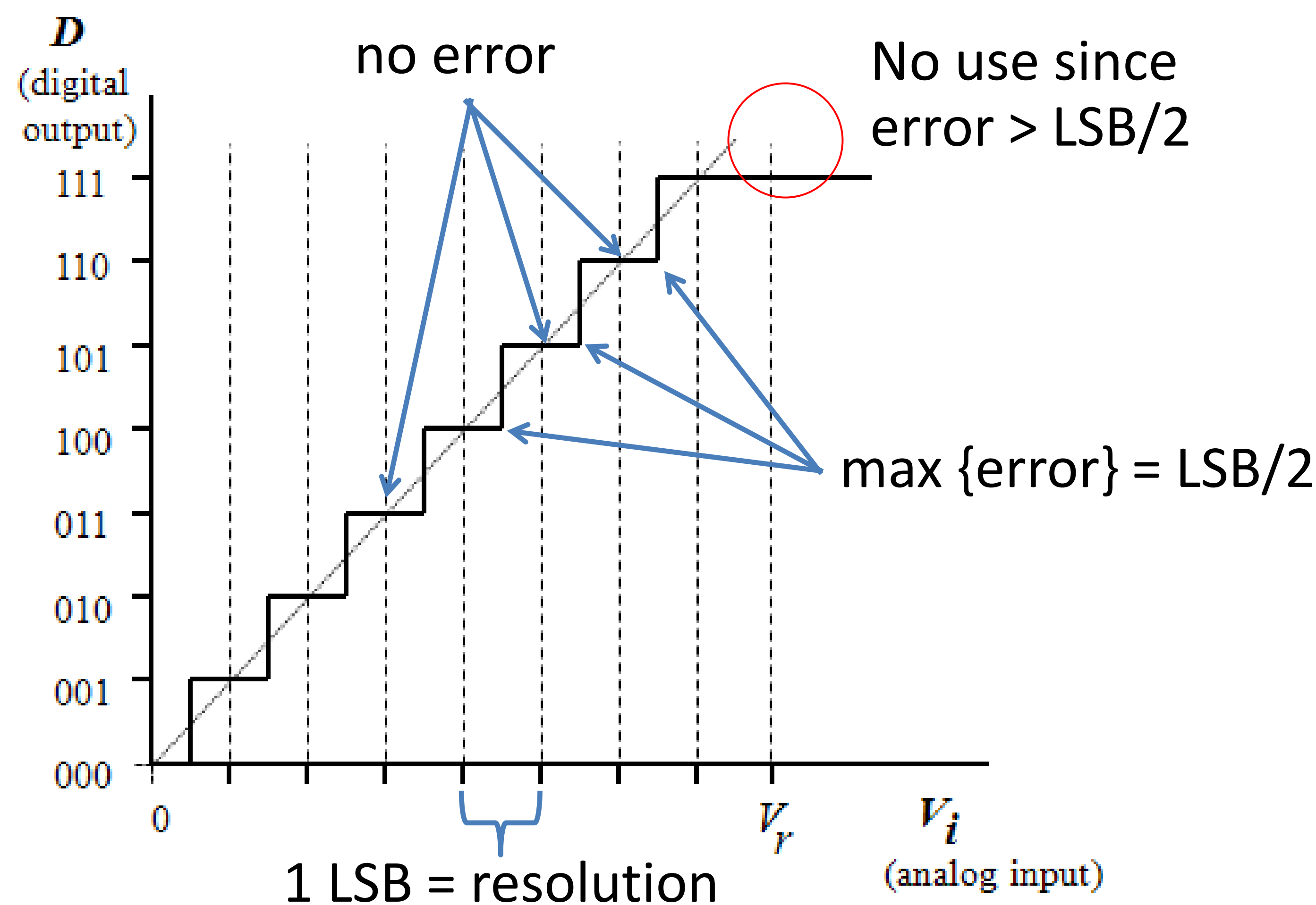
D = Digital output word: integer 0 to (2^n-1)
n = Number of bit of conversion output
V_i = Analog input of ADC
V_r = Reference voltage for using in conversion

Ex: V_r = 3.2 V, n=3, V_i = 0.8 D= $(0.8/3.2)(2^3)$= 2 or 010_2
V_r = 3.2 V, n=3, V_i = 0.9 D= $(0.9/3.2)(2^3)$= 2.25 => 2 or 010_2
V_r = 3.2 V, n=3, V_i = 1.9 D= $(1.9/3.2)(2^3)$= 4.75 => 5 or 101_2
V_r = 3.2 V, n=3, V_i = 2.0 D= $(2.0/3.2)(2^3)$= 5 or 101_2

Parameters of ADC

- ***Range***: The difference between maximum and minimum permissible input values
  - The input range of the ADC is directly linked to the value of the voltage reference; in many ADC circuits the range is actually equal to the reference voltage.
  - Analog inputs that exceed the maximum or minimum permissible input values are likely to be digitized as maximum and minimum values respectively, i.e. a limiting (or 'clipping') action takes place.
- ***Resolution***: The resolution is $V_r/2^n$ which is the size of a single step on the staircase characteristic (next slide).
- ***Quantization error***: The error due to mapping an interval of analog value to an integer digital word. It is equal to value of digital output – value analog input

ACD Input-Output Characteristic



3-bit ADC characteristic

ADC Input-Output Characteristic

- The output value is precisely correct for the input voltage at the middle of the step.
- The greatest quantization error occurs at either end of the step and is equal to one half of the step width, or half of one least significant bit (LSB) equivalent of the voltage scale.
- The more steps there are representing the range, the narrower they will be, and hence the quantization error is reduced.
- More steps are obtained by increasing the number of bits in the ADC process but this will
  - increase the complexity and cost of the ADC
  - Increase the time it takes to complete a conversion

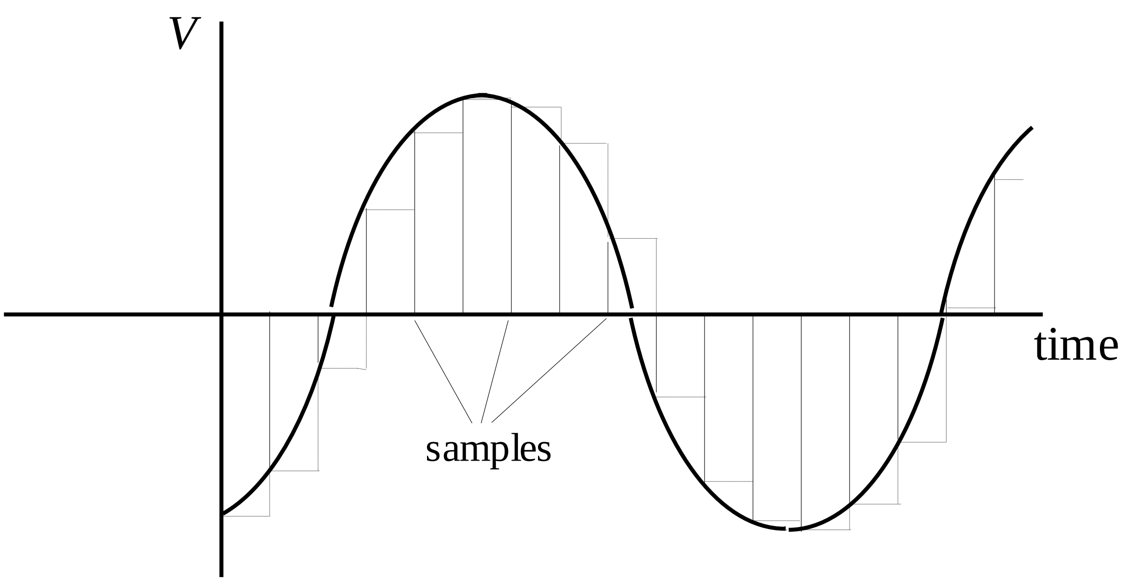
ADC Input-Output Characteristic

- Example using 8-bit converter to convert 0-3.3 Volt
  - n = 8; there are 2^8 = 256 words of digital output to represents 256 intervals of analog data.
  - Each step has a width of 3.3/256 = 12.89 mV = 1 LSB or resolution
  - The worst case quantization error is 6.45 mV = ½ LSB

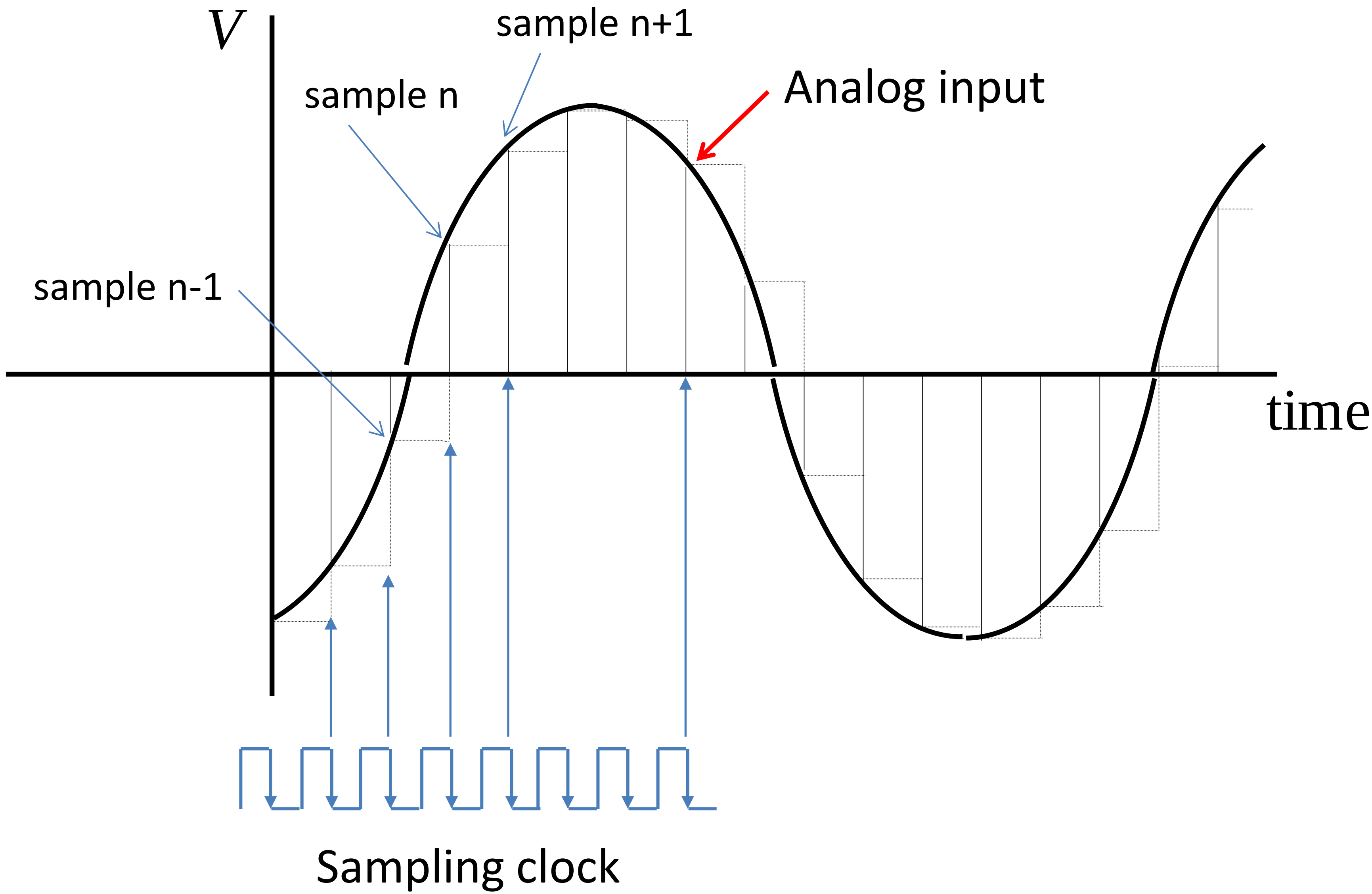
    $-6.45 \text{ mV} \leq$ quantization error $\leq 6.45 \text{ mV}$

How ADC Operates

- ADC converts analog signal to digital bit by
  - Repeatedly samples analog input signal using a fixed frequency called ***sampling frequency***.
  - A "sample" is hold and then quantized to the accuracy defined by the resolution of the ADC
  - Since analog signal keeps changing, <u>more often</u> the sample we take <u>more accurate</u> the data will be



How ADC Operates



Sampling Frequency

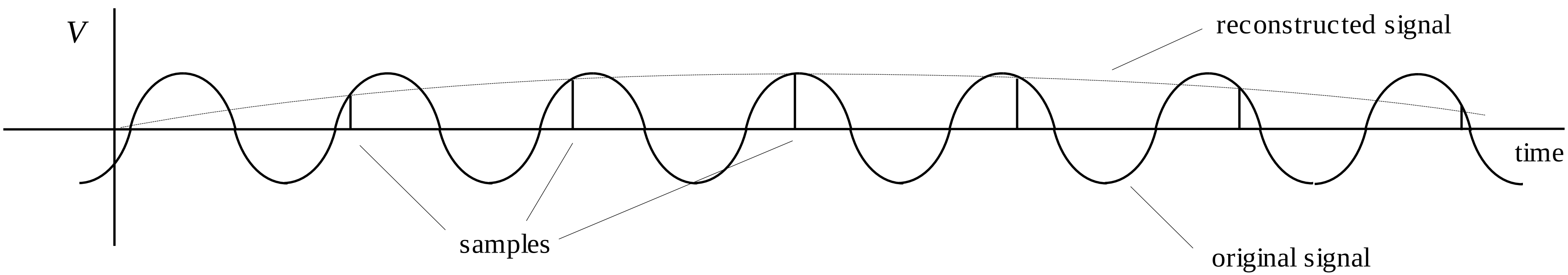
- Requirement
  - The *Nyquist* sampling criterion states that the sampling frequency f_{clk} must be at least double that of the highest signal frequency f_{signal}

$$\boxed{f_{clk} \geq 2f_{signal}}$$

  - For example, the human auditory system is known to extend up to approximately 20 kHz, so standard audio CDs are sampled and played back at 44.1 kHz in order to adhere to the *Nyquist* sampling

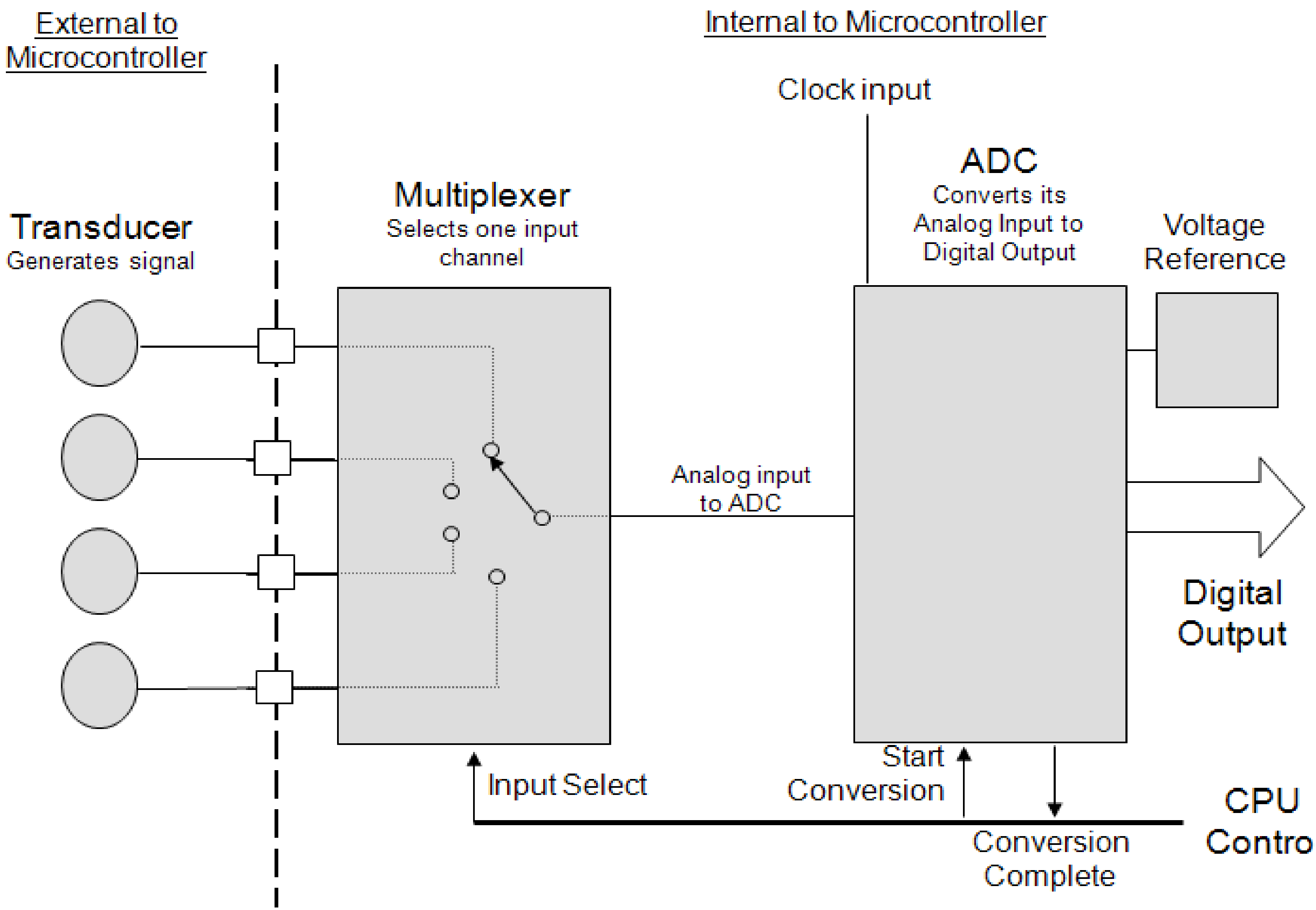
Effect of Aliasing

- If the sampling criterion is not satisfied, then a phenomenon called ***aliasing*** occurs i.e. a new lower frequency is generated



- To prevent this effect, a common approach is to use an anti-aliasing filter, which limits all signal components to those that satisfy the sampling criterion.

Data Acquisition System

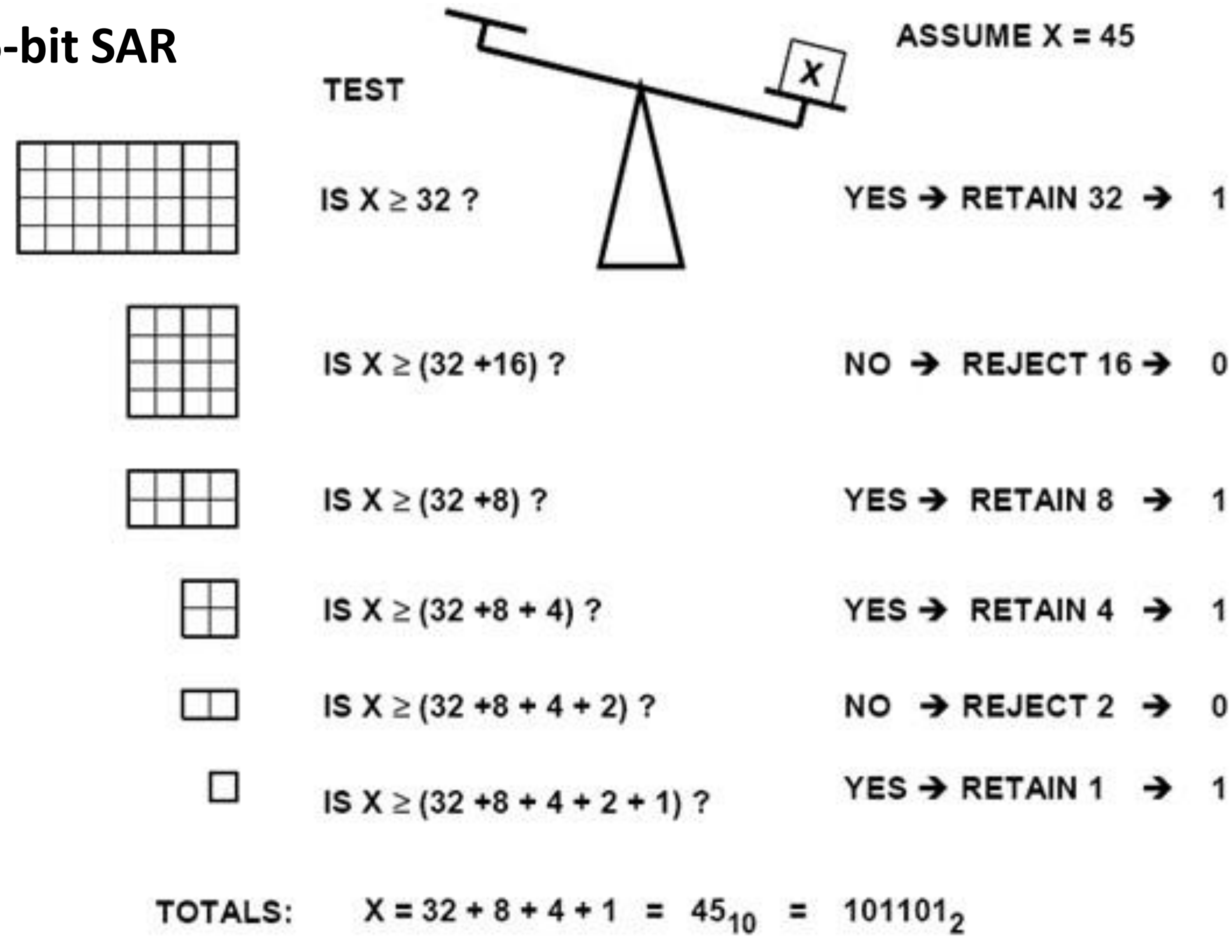


ADC as a part of data acquisition system

LPC1769: ADC Architecture

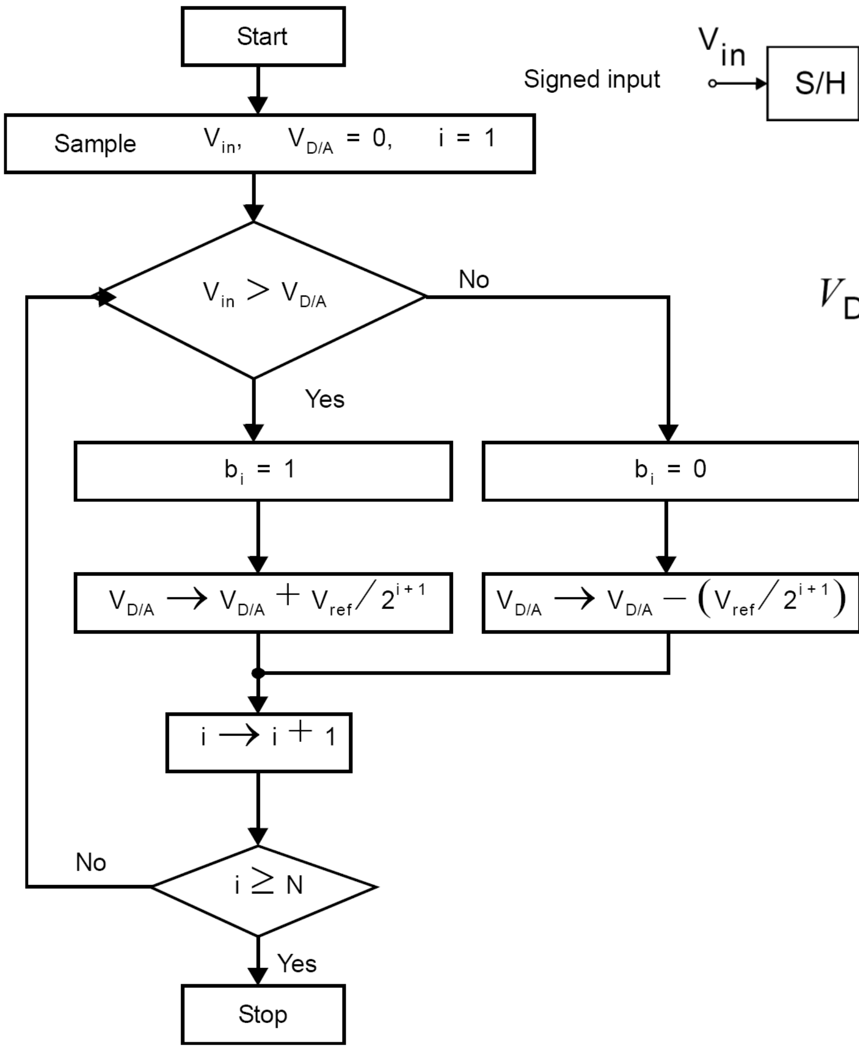
- Using 12-bit Successive Approximation DAC

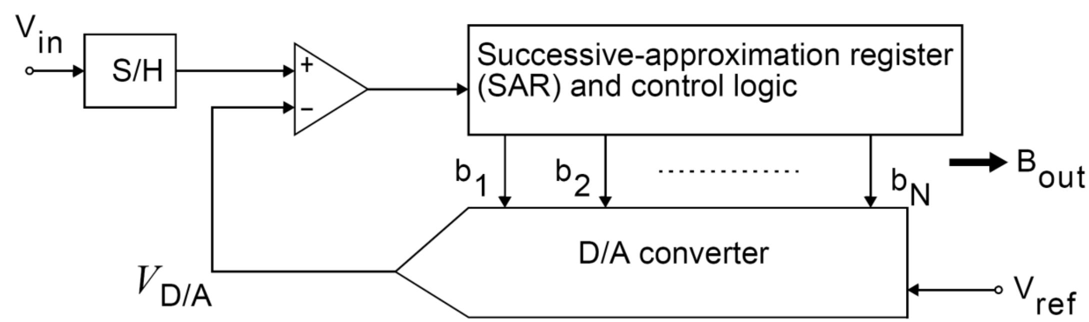
Ex: 6-bit SAR



Source: Texas Instrument

How Successive Approximation works





- Using binary search algorithm
- Adjust $V_{D/A}$ until within 1 LSB of V_{in}
- Requires ***n*** steps for ***n*-bit** converter

(Credit: Prof. Kent Martin, U of Toronto)

How Successive Approximation works

- Example