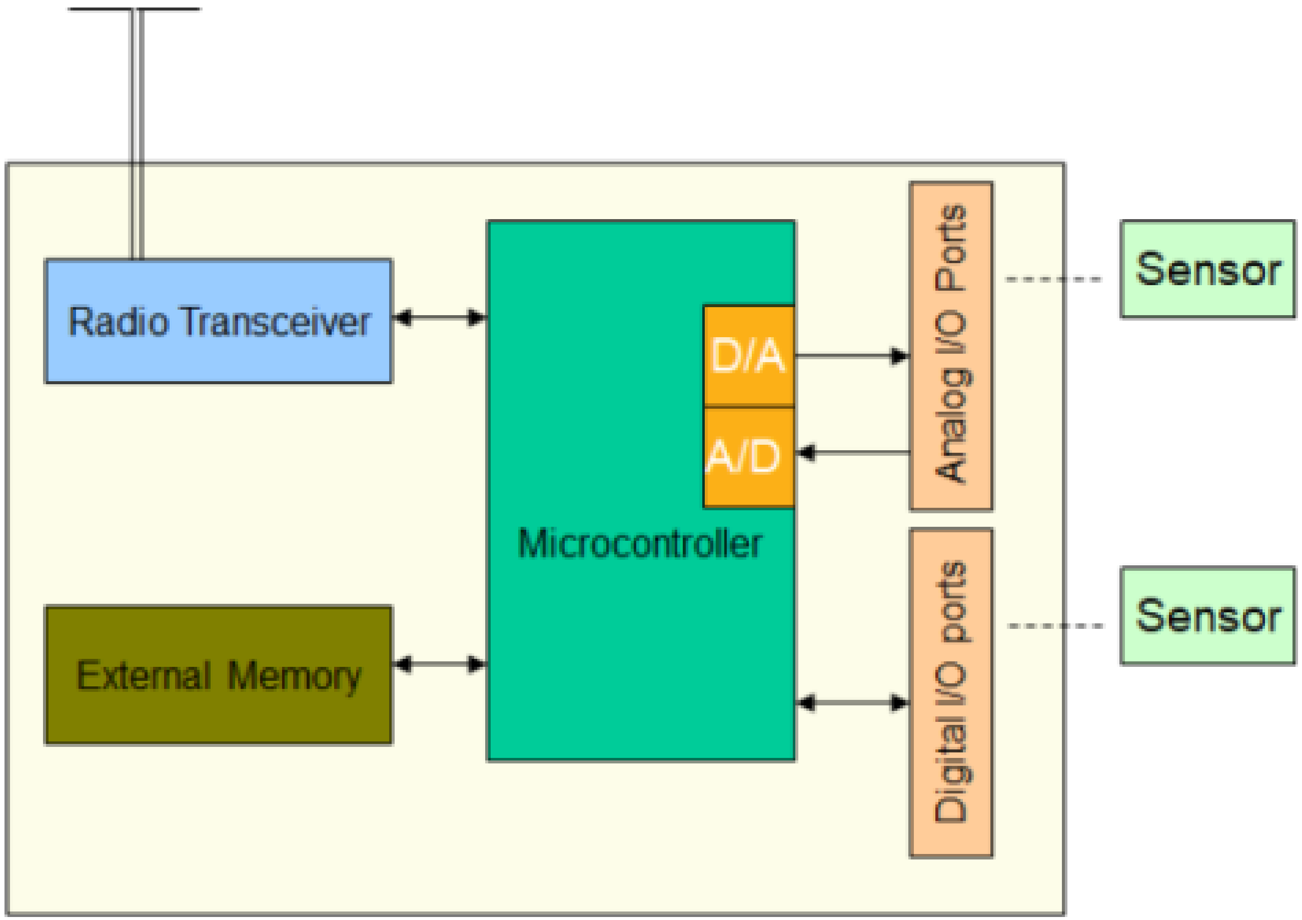
# Data acquisition, signal conditioning and Analog to Digital

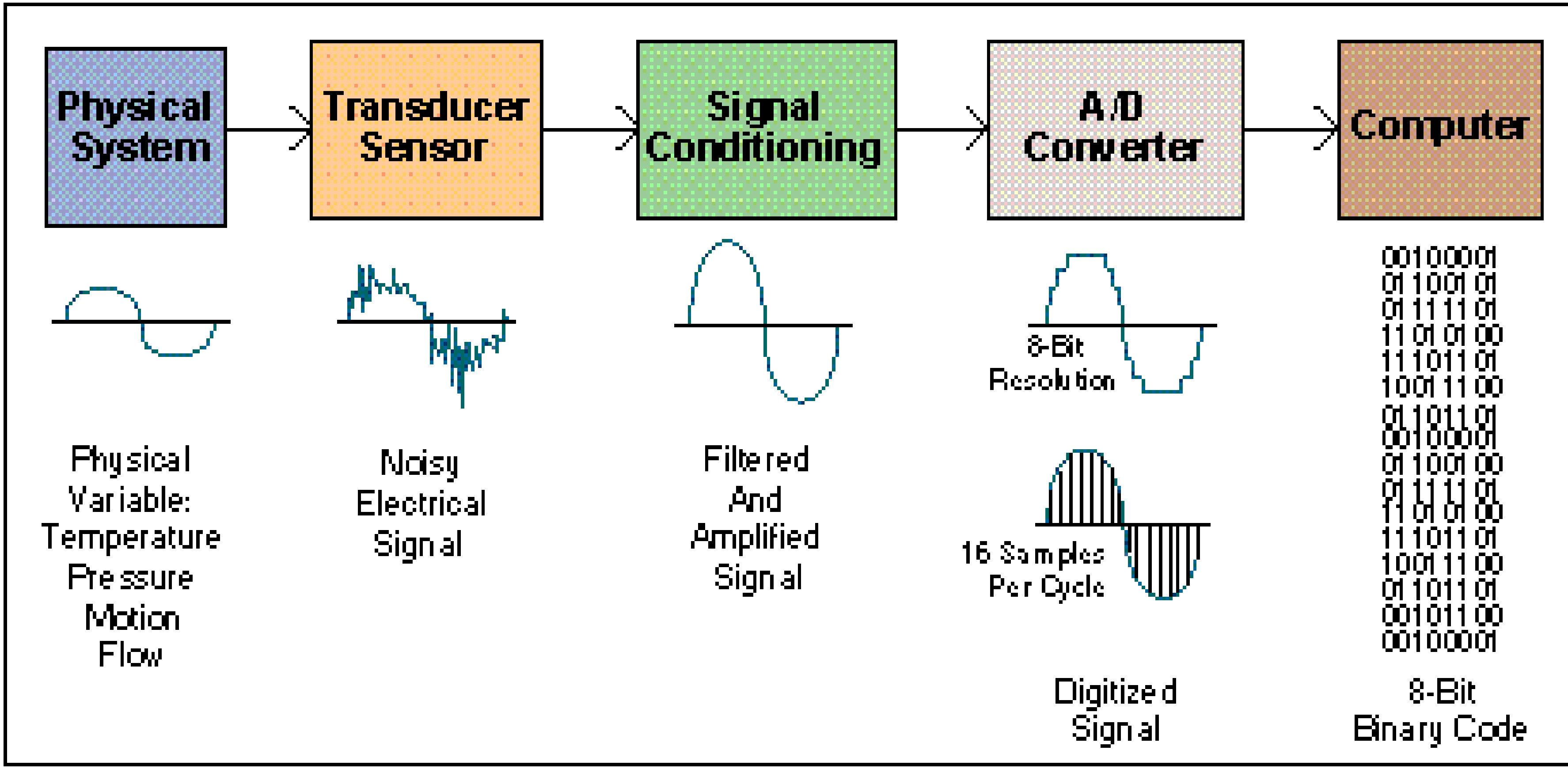
# WSN mote




Radio Transceiver
Microcontroller
D/A
A/D
Analog I/O Ports
Digital I/O ports
External Memory
Sensor
Sensor

# Data acquisition by mote- steps involved

BITS Pilani



# Physical System



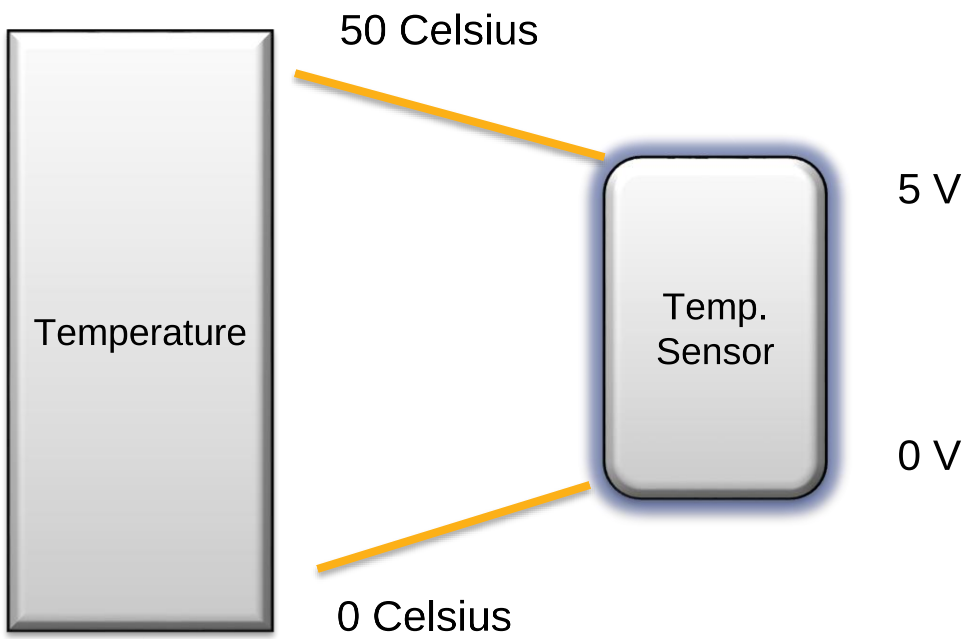
- The **physical system gives input to the DAQ System** for monitoring different system parameters like:
  - Pressure
  - Force
  - Light
  - Temperature
  - Displacement
  - ON/OFF state
  - Level
  - Electrical signals
  - Acceleration

# Transducers / Sensors

BITS Pilani

- A transducer **converts** temperature, pressure, level, length, position, etc. into voltage, current, frequency, pulses or other signals.

- A transducer thus converts the physical conditions in electrical waveform for easy signal processing




# Common Sensors and parameters measured

BITS Pilani

| Sensor used for DAQ | Physical parameter measured |
| --- | --- |
| Resistive temperature device (RTD) | Temperature |
| Gauge | Pressure |
| Microphone | Pressure |
| Accelerometer | Acceleration |
| Thermocouple | Temperature |
| Strain gauge | Force |
| Light sensor | Illuminance |
| Humidity sensor | Humidity |

# Signal Conditioning

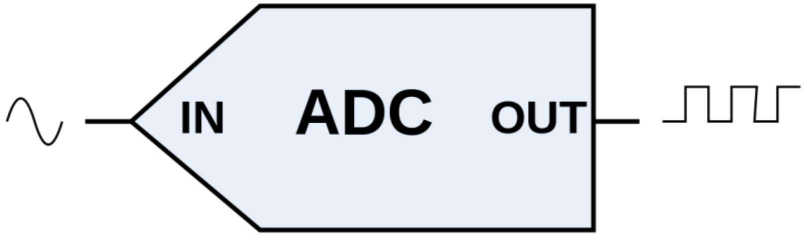


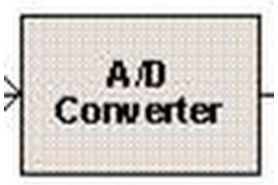
- Signal conditioning circuits **improve the quality of signals** generated by transducers before they are converted into digital signals by the PC's data- acquisition hardware.

- Most common signal conditioning functions are amplification, linearization, cold-junction compensation, filtering, attenuation, excitation, common-mode rejection, and so on.

# Analog to Digital convertor

- Analog to digital (A/D) conversion **changes analog voltage or current levels into digital information**. The conversion is necessary to enable the microcontroller to process or store the signals.



# Microcontroller in Mote

BITS Pilani

- This processes the digital signal.

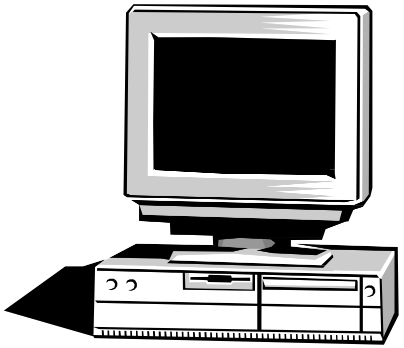
# Base station (PC)

BITS Pilani

- Can install Data Acquisition Software

- Data Acquisition Software can be the most critical factor in obtaining reliable, high performance operation

- Different alternatives:
  - ➢ Programmable software.
  - ➢ Data acquisition software packages



# Programmable Software

BITS Pilani

- Involves the use of a programming language, such as:
  - C++
  - Visual C++
  - BASIC,
  - Visual Basic + Add-on tools (such as VisuaLab with VTX)

- **Advantage**: flexibility
- **Disadvantages**: complexity and steep learning curve

# Data Acquisition Software Packages

BITS Pilani

- These do not require programming.
- They enable the developers to design the custom instrument best suited to their application.

- Examples: TestPoint, SnapMaster, LabView, MatLab, DADISP, DASYLAB, etc.

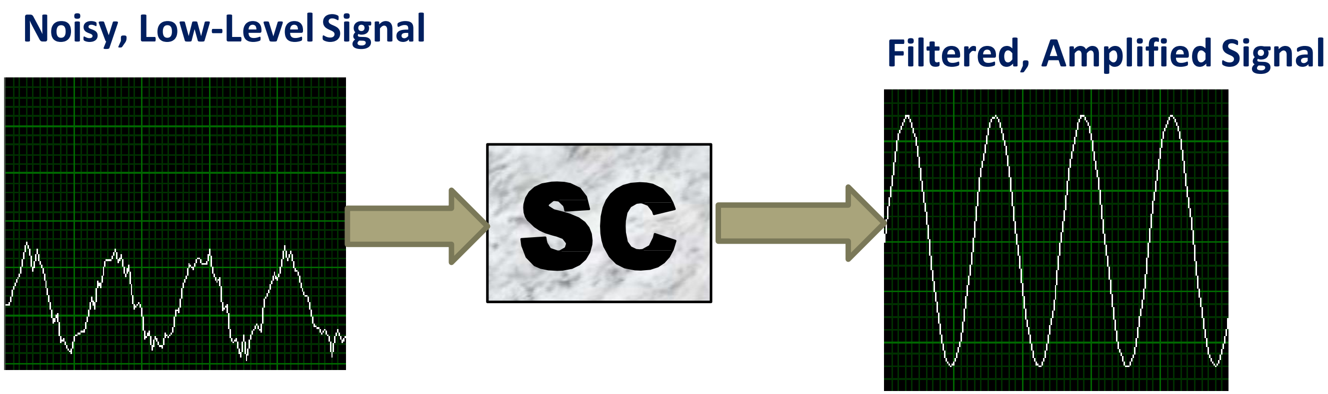
# Signal conditioning

BITS Pilani

- To **remove noise** from the measured sensor data or to improve it for usage
- Common types of signal conditioning include amplification, linearization etc.
- Sensor signals are often incompatible with data acquisition hardware. To overcome this incompatibility, the sensor signal must be conditioned.

# Why conditioning

BITS Pilani




Improving the quality of signals
(e.g. by Amplification  Filtering  Isolation)

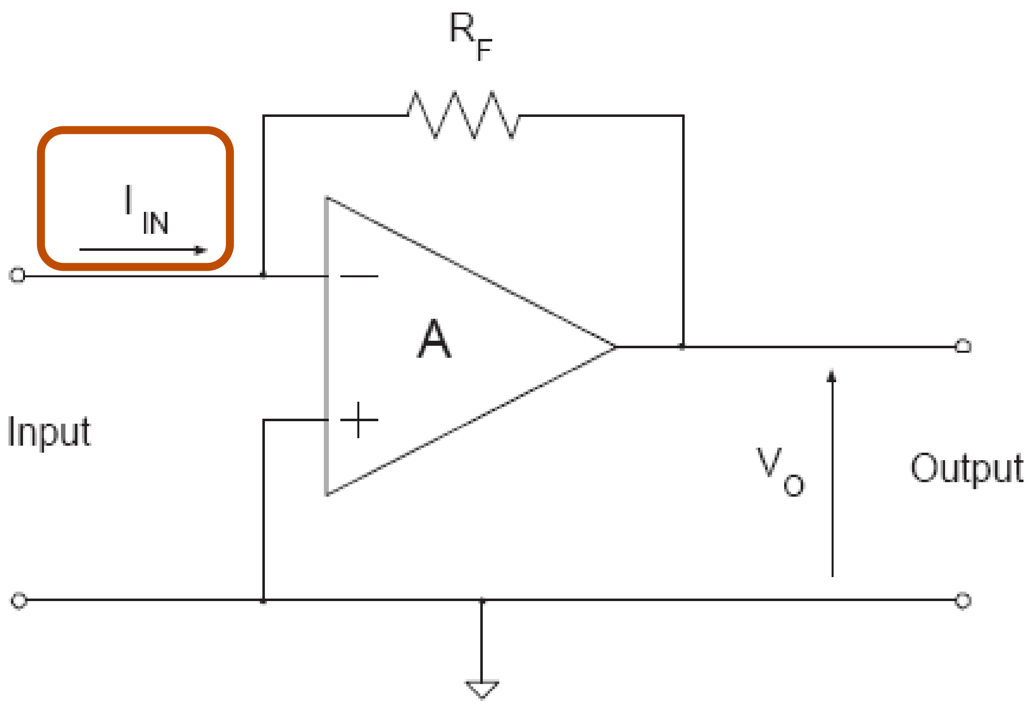
# Common ways to condition signals include



- Signal Conversion
- Amplification
- Attenuation
- Filtering
- Electrical isolation
- Multiplexing
- Excitation source

# Signal Conversion: Ex: Current-to-voltage converter

- Transimpedance amplifier (Feedback Ammeter)
- Recommended connection for **small currents**
- Sensitivity determined by Rf




$$\boxed{V_O = -I_{IN} R_F}$$

# Amplification

BITS Pilani

- **When the input signal is as small** as microvolts, electrical noise can drown out the signal itself, leading to meaningless data.

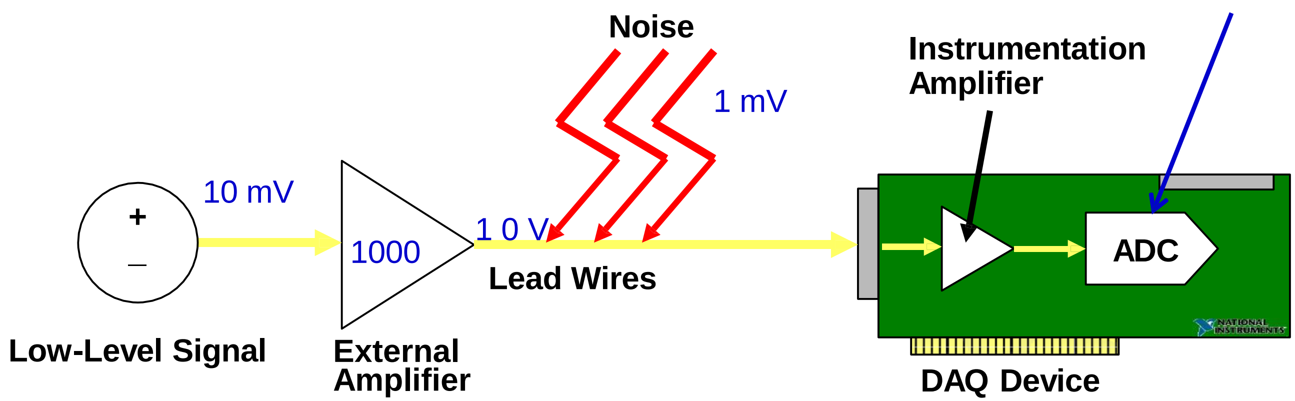
- For reducing the effects of noise on your signal is to amplify the signal as close to the source as possible. This increases Signal to Noise Ratio (SNR)

- e.g. A J-type thermocouple outputs a very low-level voltage signal that varies by about 50 µV/°C.

# Amplification

BITS Pilani

- Used on low-level signals (less than around 100 mV )
- Maximizes use of Analog-to-Digital Converter (ADC) range and increases accuracy
- Increases Signal to Noise Ratio (SNR)



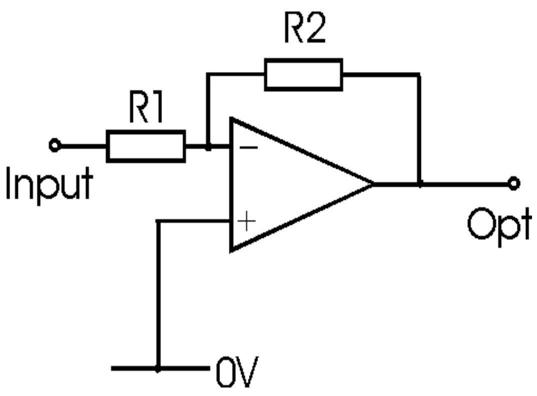

SNR = $V_{signal}/V_{noise}$ = (10 mV * 1000) /1 mV = 10 000

$$SNR = 20\log\left(\frac{V_{signal}}{V_{noise}}\right)$$

# Example of amplifier: Operational amplifier (Op-amp)

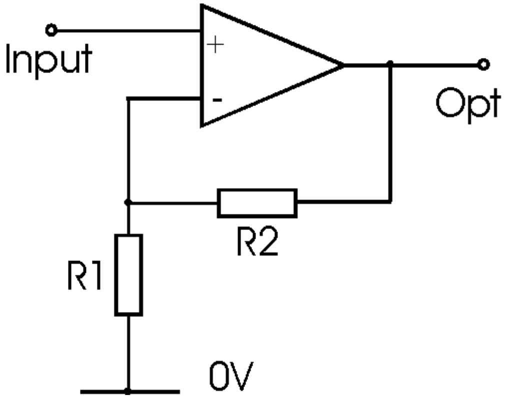
BITS Pilani

- Inverting op-amp amplifier
  - Vo = -R2/R1 * Vi




Inverting Amplifier

- Non-inverting op-amp amplifier
  - Vo = (1+R2/R1) * Vi

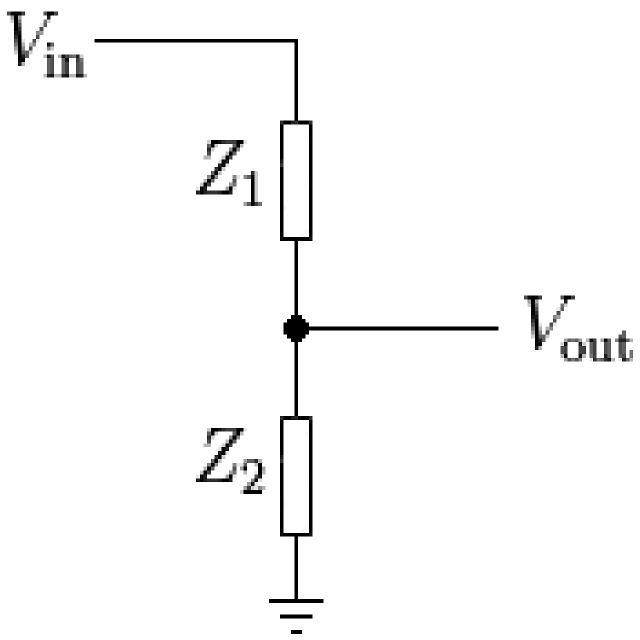



Non-Inverting Amplifier

# Attenuation



- Voltage divider: A circuit that produces an output voltage (Vout) that is a fraction of its input voltage (Vin)
- Can be needed to get a high-level signal down to the acceptable DAQ-card range




A simple attenuation circuit

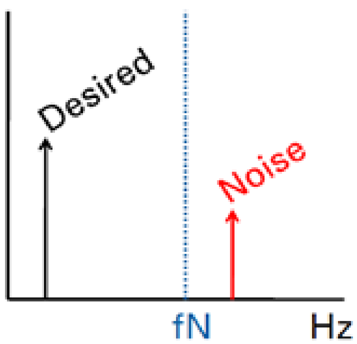
# Filtering

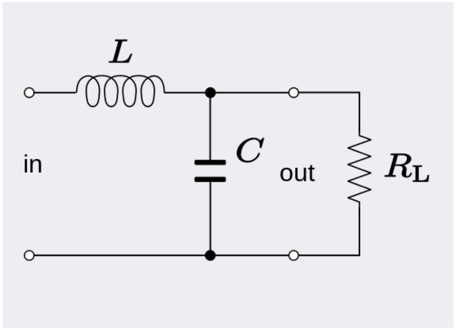
BITS Pilani

- To reject unwanted noise within a certain frequency range.
- Many systems will exhibit periodic noise from sources such as power supplies or machinery.
- Examples of filters:
  - Butterworth Filter
  - Bessel Filter
  - Chebyshev Filter
  - Simple RC Filter
  - Passive & Active Filters

# Hardware Filtering

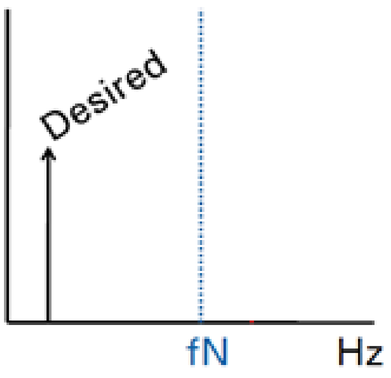
- Purpose:
  - To remove unwanted signals from the signal that you are trying to measure




LPF

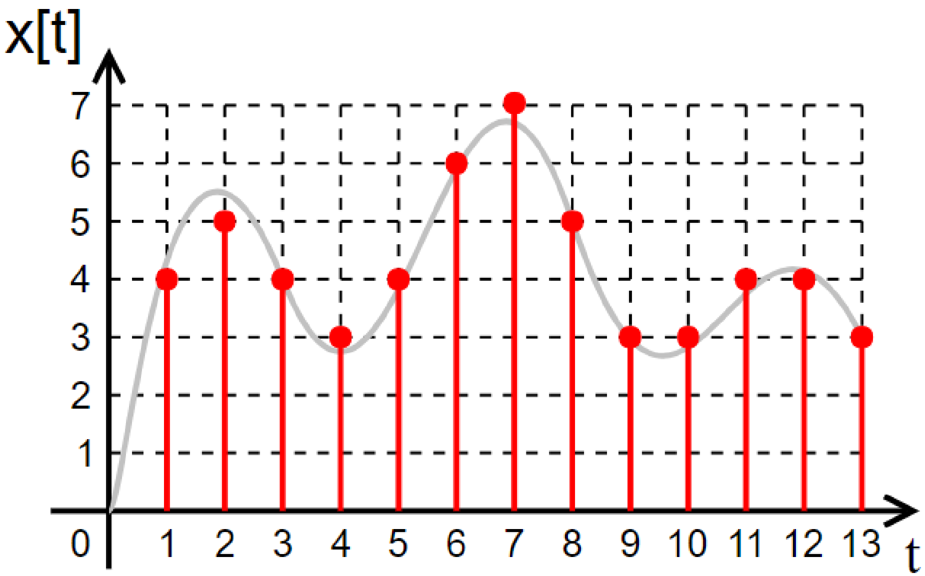
Low pass filter



# A/D conversion

BITS Pilani

- Digital data is more convenient to process/store by the microcontroller as compared to the analog signal.
- Resort to Analog to Digital conversion



# A/D conversion



- After the signal conditioning, the signal from the sensor is passed to the analog to digital (A/D) board.

- The **A/D board converts the conditioned analog voltage or current signal into a digital format** which is readable by the microcontroller.

- A/D conversion is a **ratio operation**, where the input signal is compared to a reference and converted into a fraction which is then represented as a coded digital number.

# A/D conversion



- To optimize measurement accuracy, **there is a minimum and maximum number of data points that need to be acquired**.

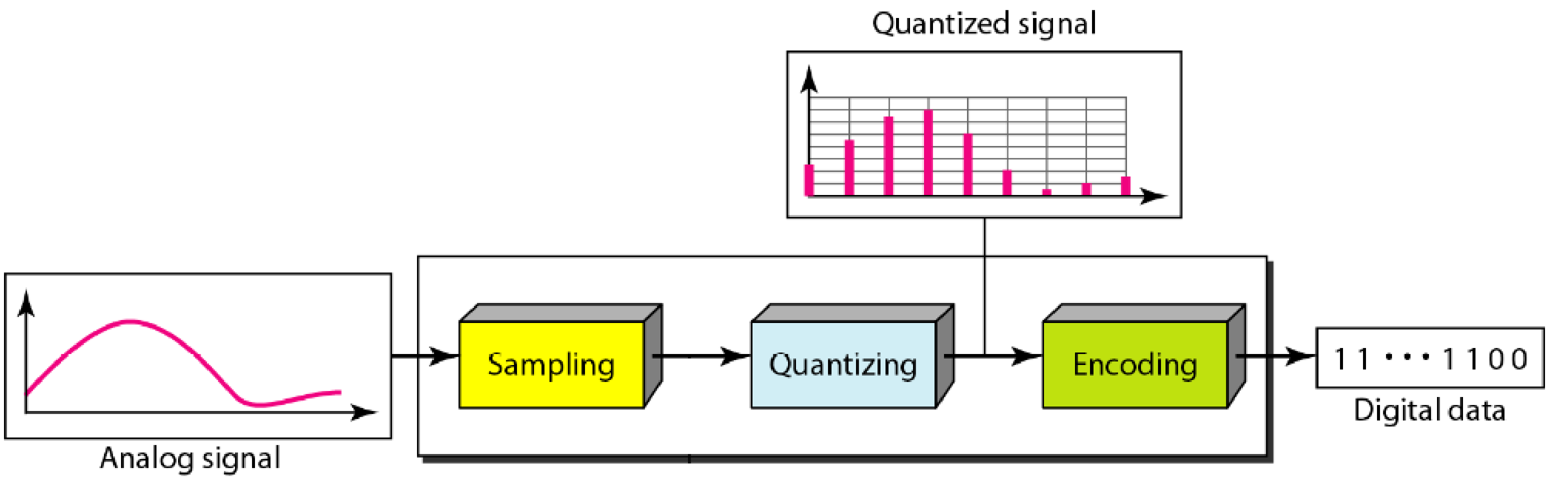
- One of the most critical factors when selecting an A/D board is **sampling rate** (speed).

- The sampling rate is a measure of **how rapidly the A/D board can scan the input channel** and identify the discrete value of the signal present with respect to a reference signal.

# A/D steps

BITS Pilani

- Sampling
- Quantization
- Encoding

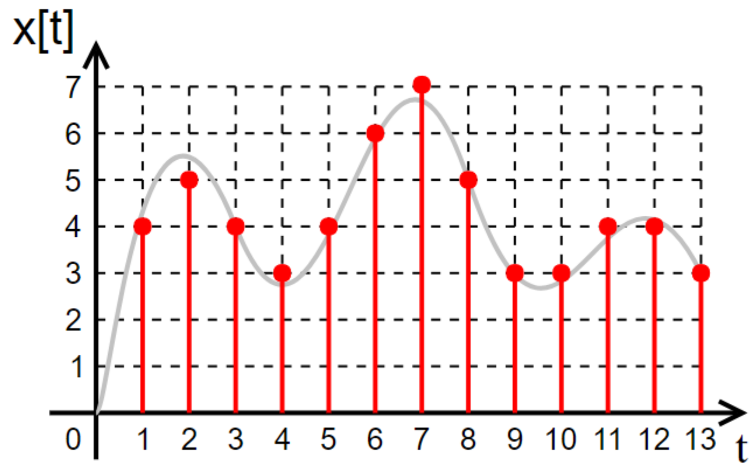



https://sipdtdevelopers.files.wordpress.com/2013/11/quantization.png

## Sampling Considerations



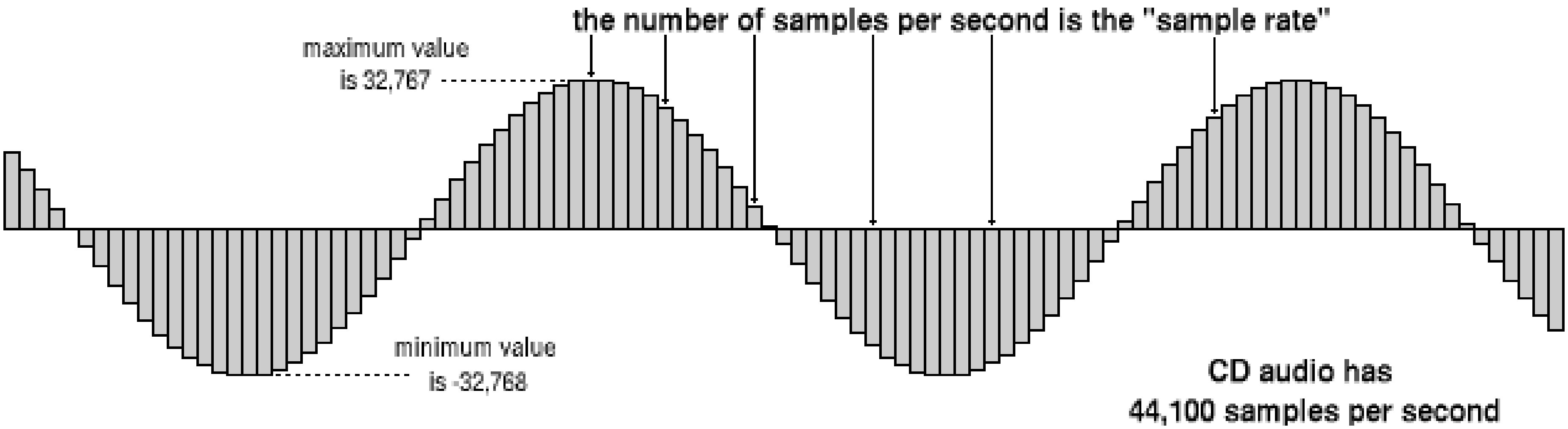
- An analog signal is continuous.
- A sampled signal is a series of discrete samples acquired at a specified sampling rate.
- The faster we sample the more our sampled signal will look like our actual signal
- If not sampled fast enough a problem known as aliasing will occur



# Sampling rate

BITS Pilani

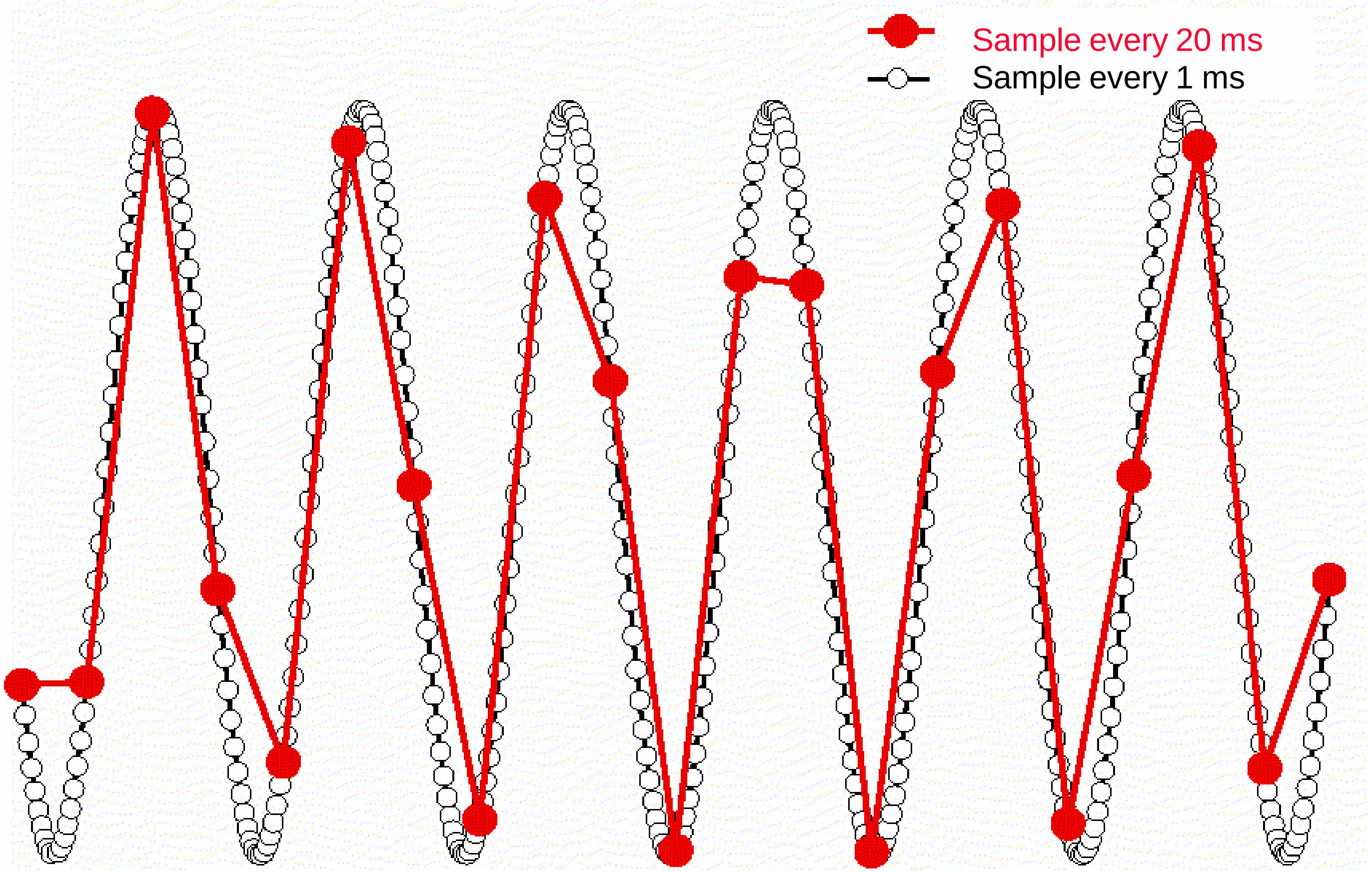
- Sampling an analog signal occurs at discrete time intervals.
- The rate at which the signal is sampled is known as the **sampling frequency.**




http://wiki.hydrogenaud.io/images/3/37/Digital_wave.png

# Effect on sampling rate on information captured/lost

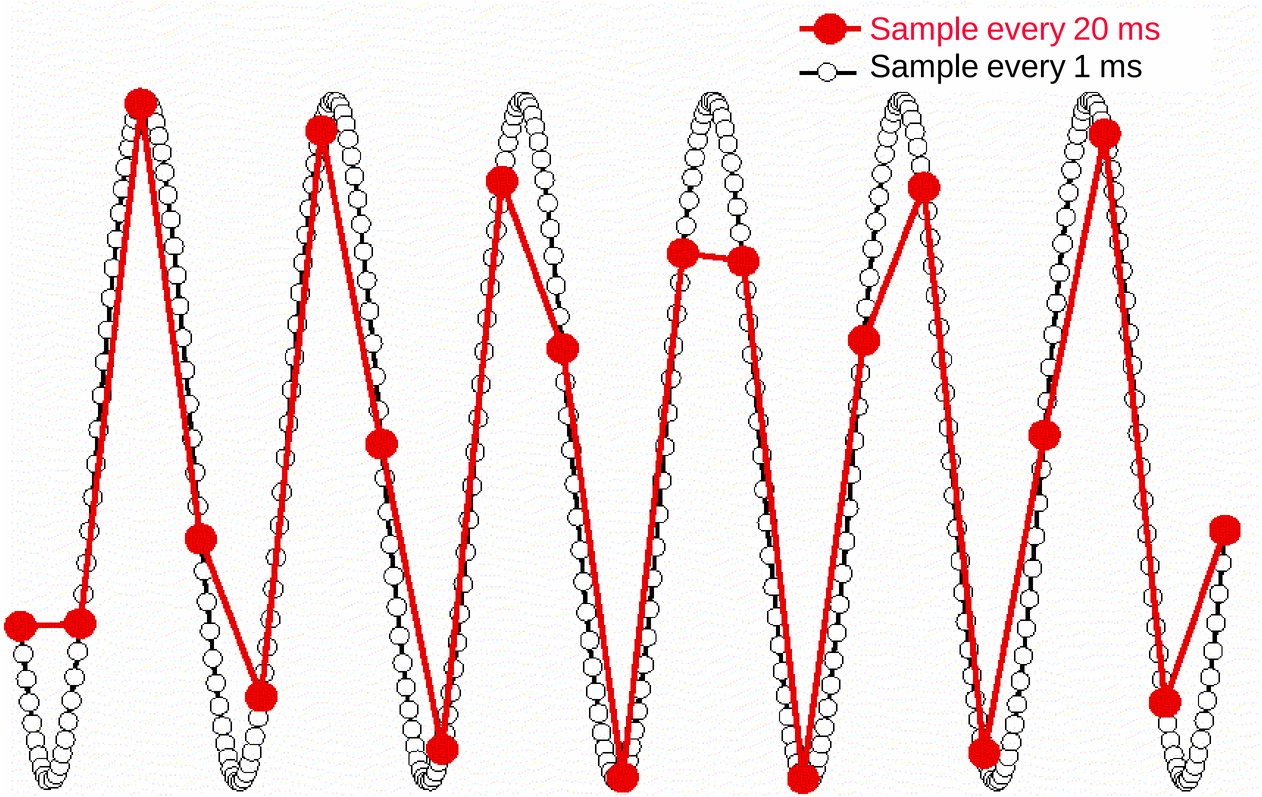
BITS Pilani



# Aliasing

BITS Pilani

- **If the sampling rate is too slow, then a completely different waveform of a lower frequency is constructed from the data acquired**. This effect is called aliasing.



# Aliasing



- It has the effect of increasing the variance in the recorded signal, i.e. it **adds noise to the signa**l, basically by missing the peaks and troughs of the rapidly changing signal.

- So, even if the signal has the same peak all the time, the board will catch the rising and falling phase but miss the peak giving the appearance that the peak (i.e. the maximum value recorded in each cycle) is changing.

# How to avoid aliasing



- To avoid aliasing, it is necessary that the sample rate be at least twice the highest expected frequency input.

# Nyquist Sampling Theorem



- ***Nyquist sampling theorem tells us that we must sample the signal at more than twice the rate of the maximum frequency component in the analog input signal to accurately represent the frequency of your signal.***

- According to the Nyquist Theorem, the sampling rate
- must be at least $2f_{max}$, where , highest frequency component = $f_{max}$

# Choosing the sampling rate

BITS Pilani

- Application specific
- Eg. Recording the ambient temperature during the day manually at intervals of 30 minutes.

| Time | Temperature | Time | Temperature |
|---|---|---|---|
| 6.00 | 10 | 10.30 | 15 |
| 6.30 | 11 | 11.00 | 15 |
| 7.00 | 11 | 11.30 | 17 |
| 7.30 | 12 | 12.00 | 18 |
| 8.00 | 13 | 12.30 | 19 |
| 8.30 | 13 | 13.00 | 19 |
| 9.00 | 14 | 13.30 | 18 |
| 9.30 | 14 | 14 | 17 |
| 10.00 | 15 | 14.30 | 16 |

# Example: Temperature measurement



- A platinum resistance thermometer can be used as the transducer which is a temperature-dependent resistor, and by using an appropriate circuit, a continuous measurement of the actual temperature in the form of a proportional voltage is obtained.

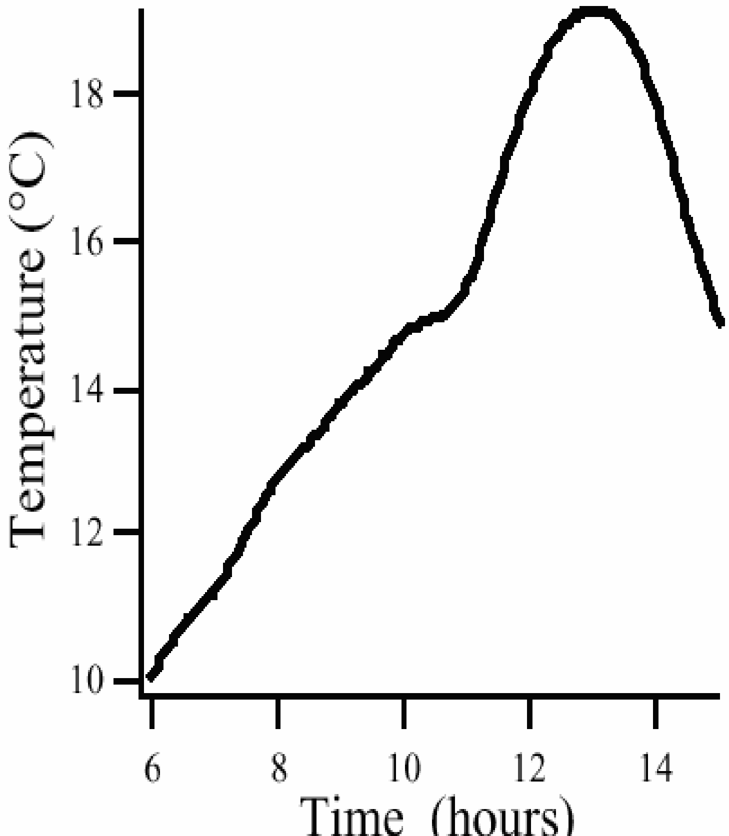
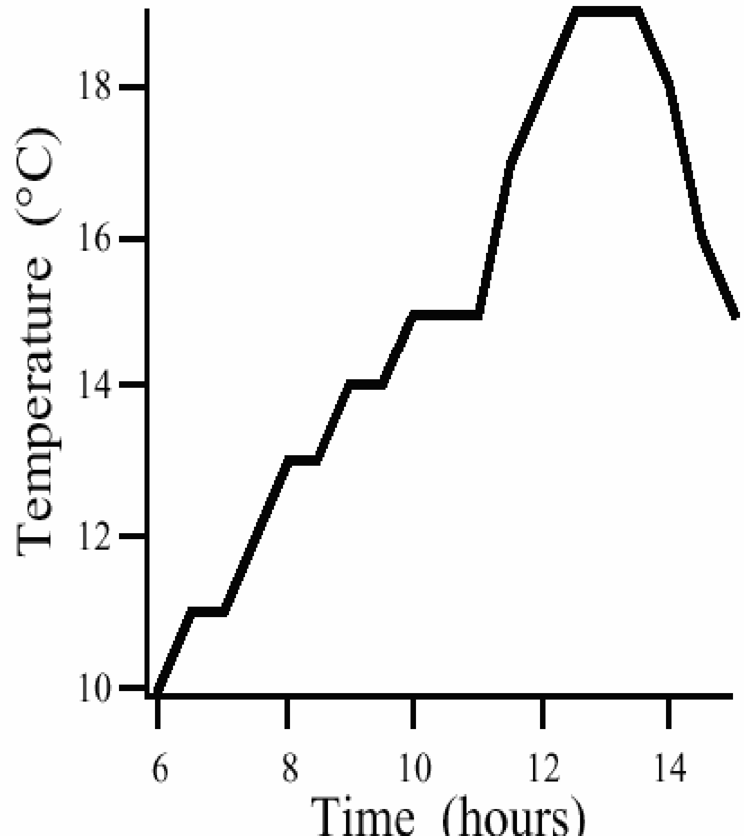
# Example: Temperature measurement (cont.)

BITS Pilani

- The data acquisition software converts the analog voltage corresponding to the temperature into binary numbers (digital format) in every 30 minutes.

- Assume that the A/D board rounds off all numbers within its operating range to the nearest 1°C. That is although the ambient temperature changes continuously, the A/D board only indicates a change in it when a difference greater than 1°C is observed. The data thus changes in 1°C steps.

# Temperature record

| Continuous, analog measurements of temperature | Discrete digital measurements of temperature |
| --- | --- |
|  |  |

# Applications with high sampling rate requirement

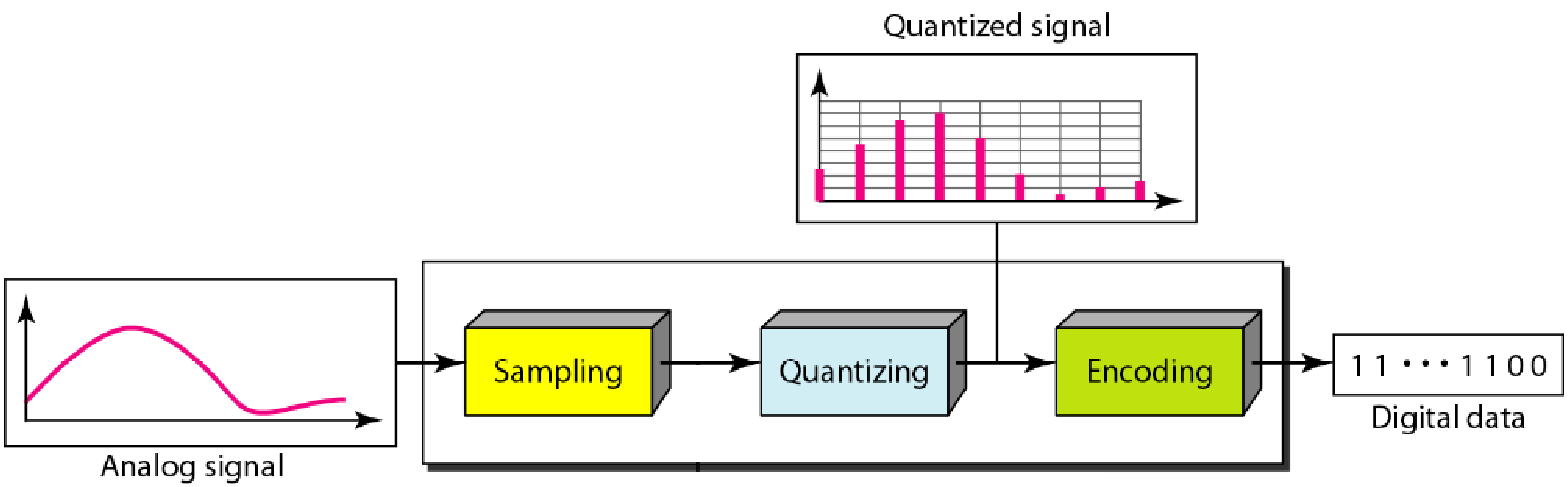
BITS Pilani

- You can’t have this high a sampling period when you are having a system which is giving real time images of the patient’s heart for a doctor doing a crucial surgery.

# A/D steps



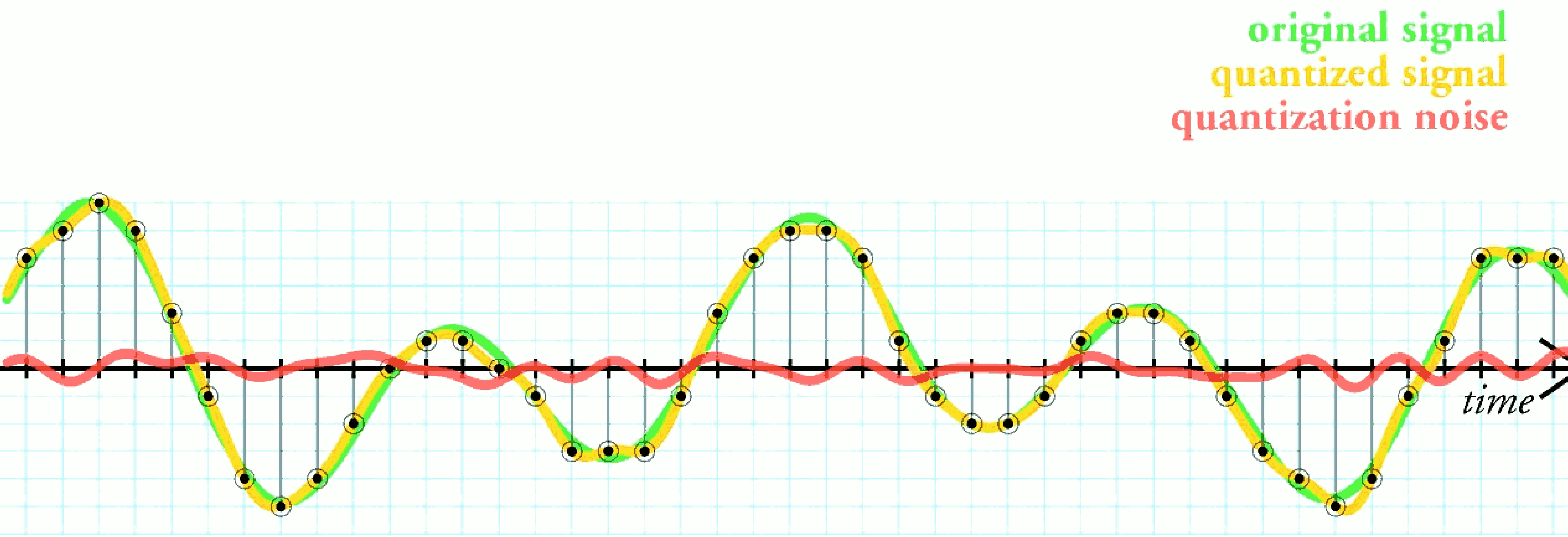
- Sampling
- Quantization
- Encoding




https://sipdtdevelopers.files.wordpress.com/2013/11/quantization.png

# Quantization- Introduction

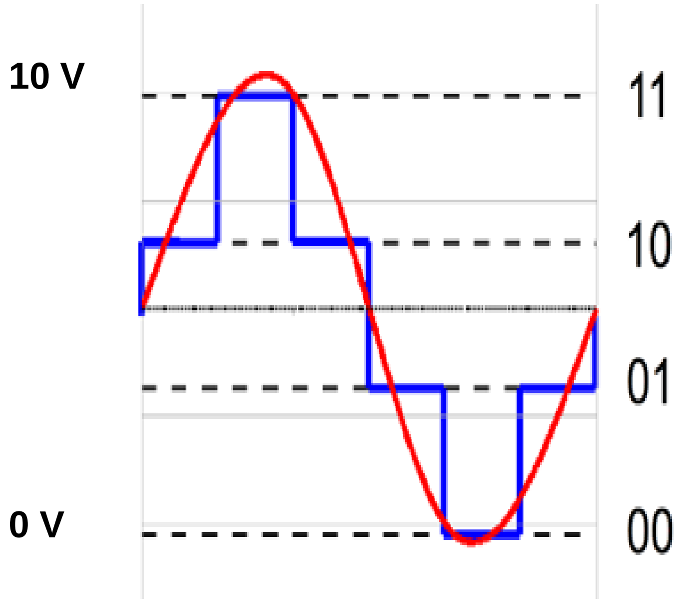




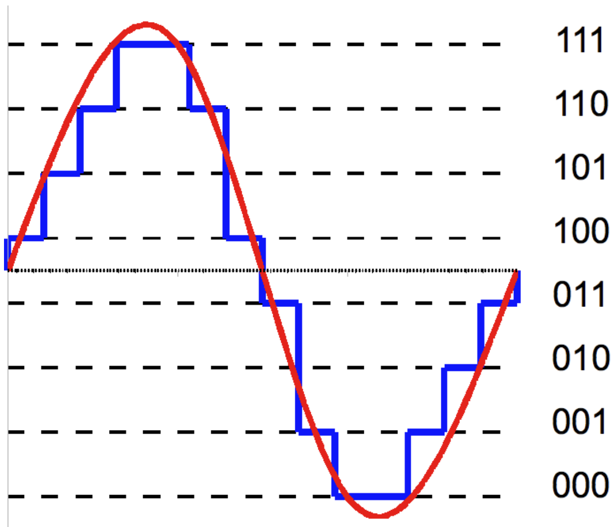
# Resolution

BITS Pilani

- Precision of the analog to digital conversion process is dependent upon the number (n) of bits the ADC of the DAQ uses.
- More the bits used for ADC, more is the number of divisions the voltage range is broken into (2^n), and thus more is the accuracy.
- An 2 bit ADC gives 4 levels (2^2) compared to a 3 bit ADC that has 8 levels (2^3).




2-bit ADC




3-bit ADC

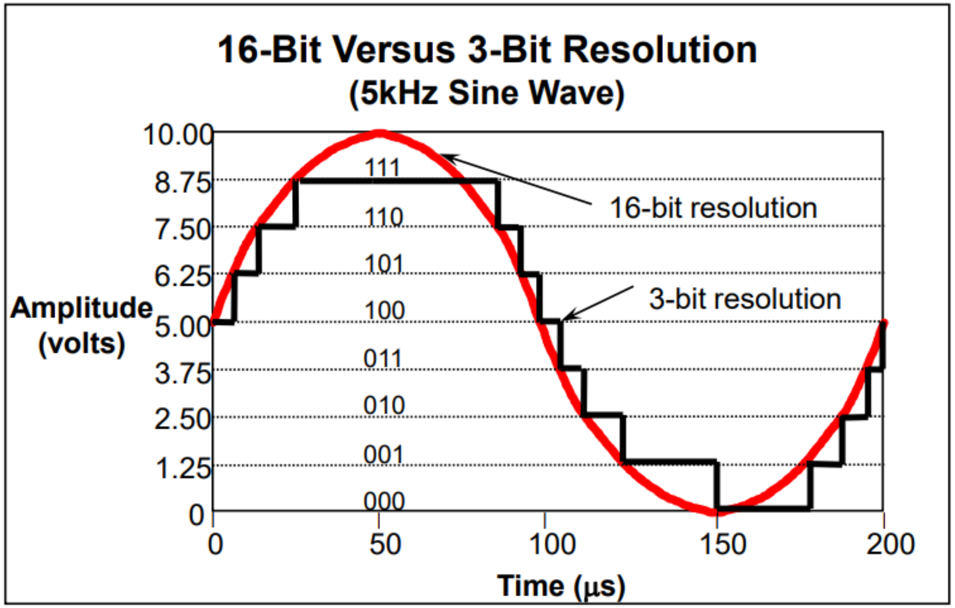
# Resolution

- The number of bits used to represent an analog signal determines the resolution of the ADC
- **Larger resolution = more precise representation of your signal**
- The resolution determines the **smallest detectable change in the input signal, referred to as code width** or LSB (least significant bit)

$$code\ width = \frac{device\ range}{2^{resolution}}$$

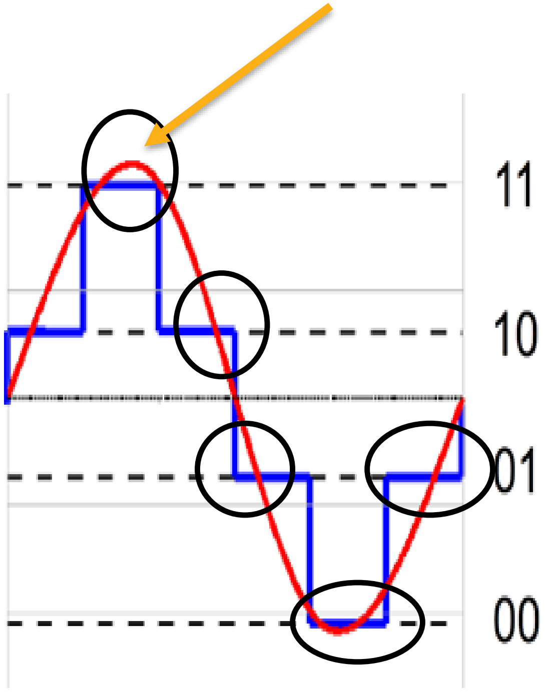
Example:

$$\frac{device\ range}{2^{resolution}} = \frac{10}{2^{16}} = .15\ mV$$



# Quantization error

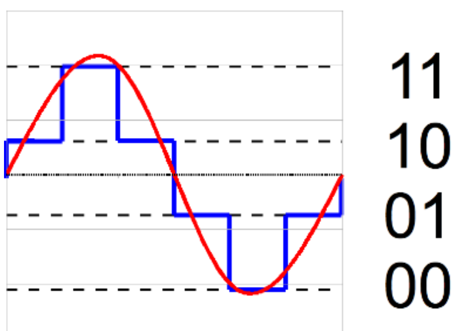
- Error incurred during the process of quantization




**2-bit ADC**

# Encoding

- Encoding is based on sampling and quantization.
- A simple encoding scheme is shown below.



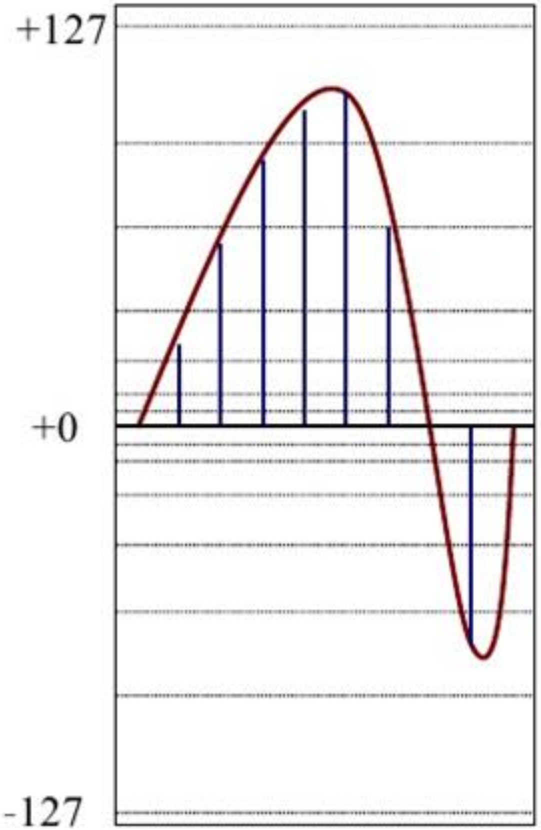

2-bit ADC

Encoder

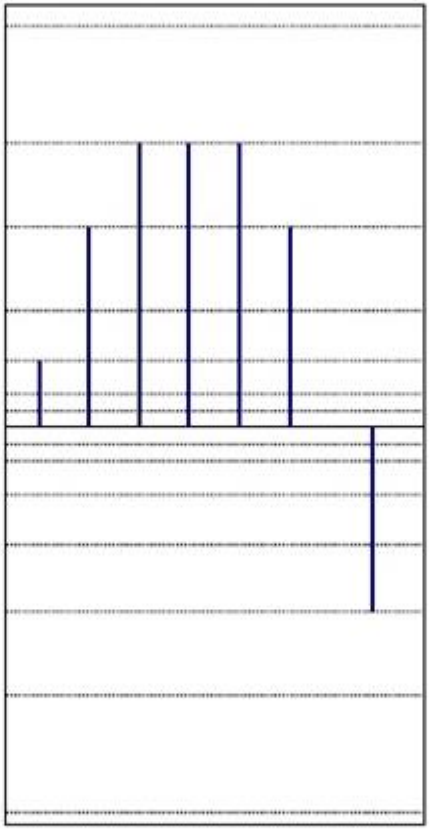
**10 11 10 01 00 01**

# Sampling – Quantization - Encoding

Sampling




Quantization



Encoding

10101111...01101101