~~Embedded Systems~~

HCI ✓

Sensors

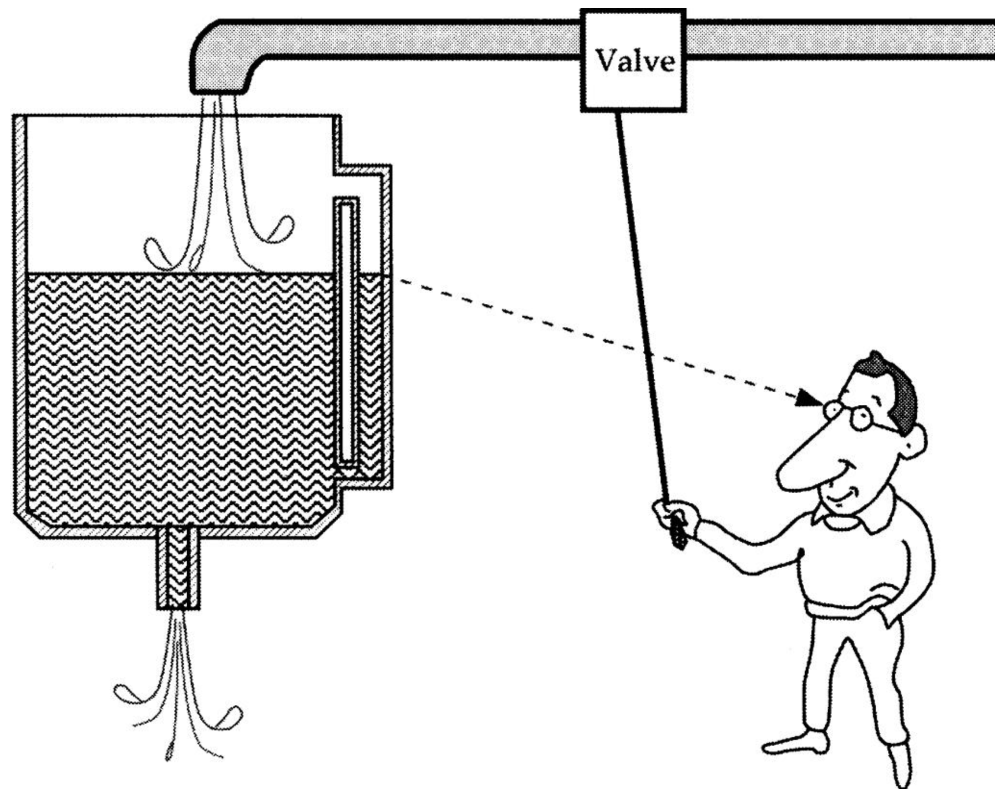
Internet-of-Things (IoT)

COCOS20

Sensors

- A sensor is a device that receives a stimulus and responds with an electrical signal.

change



Sensors

Sensors are devices that measure a particular property of the environment and convert that into an electrical output

Typically, sensors have a linear function between the physical property measured and the electrical output

The sensitivity of a sensor indicates the minimum value of the measured input that can produce a certain output signal

ADC

An analog-to-digital converter employed to turn a sensor output into signals that can be processed by MCUs

(Micro-Controller unit)

Different physical phenomena underpin the operation of sensors, e.g., the piezoelectric effect (accumulation of electric charge when mechanical pressure is applied)

Computer-Process Interface

- To implement process control, the computer must
Sense collect data from and transmit signals to the production process
- Components required to implement the interface:
  - Sensors to measure continuous and discrete process variables
  - Actuators to drive continuous and discrete process parameters
  - Devices for ADC and DAC
  - I/O devices for discrete data

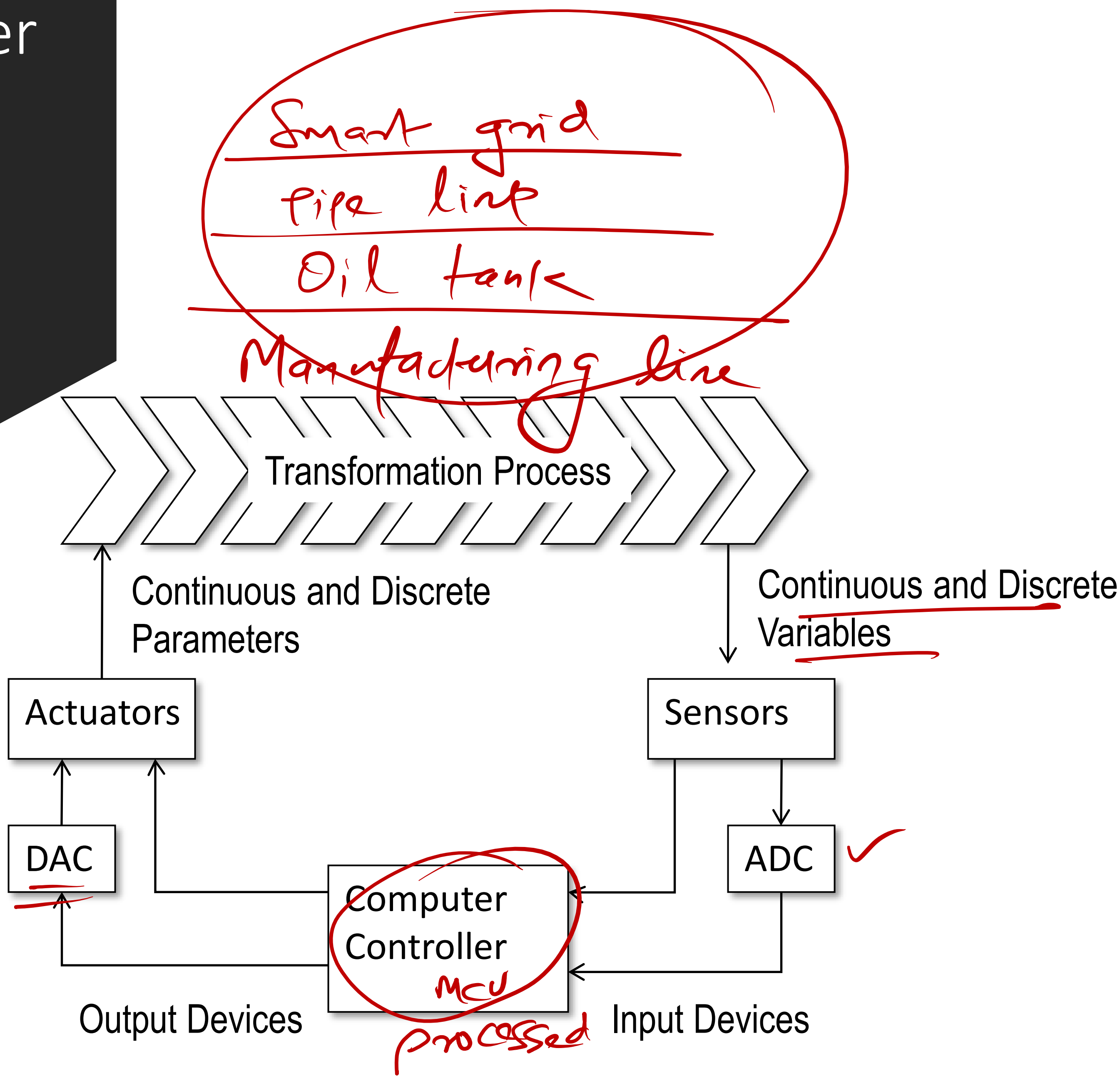
Need for Sensors

- Sensors are omnipresent. They embedded in our bodies, automobiles, airplanes, cellular telephones, radios, chemical plants, industrial plants and countless other applications.

- Without the use of sensors, there would be no automation!!

if you don't have input, then how you will decide.

Computer Process Control System

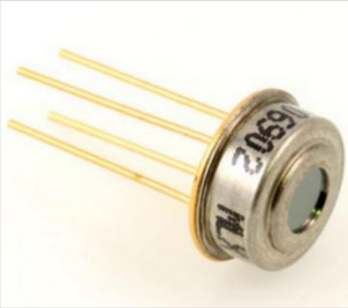


What is a Stimulus?

- Motion, position, displacement
- Velocity and acceleration
- Force, strain
- Pressure
- Flow
- Sound
- Moisture
- Light
- Radiation
- Temperature
- Chemical presence

Visual Sensor

Ultrasound Sensor



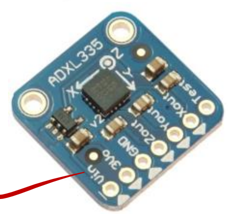
Infrared Sensor

Types of sensors

- Push-button/switch

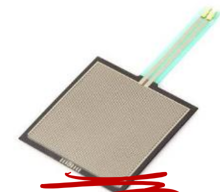
- Temperature

- Acceleration

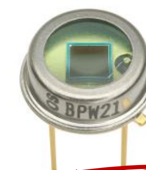


Accelerometer

- Pressure



- Optical/photodiode



- Humidity

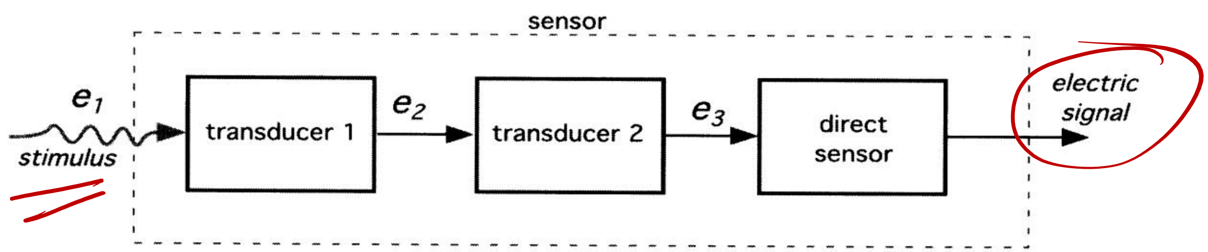
What is a Response?

When we say electrical, we mean a signal which can be channeled, amplified, and modified by electronic devices:

- **Voltage**
- **Current**
- **Charge**

Sensor as Energy Converter

- This conversion can be direct or it may require transducers



- Example:

  A chemical sensor may have a part which converts the energy of a chemical reaction into heat (transducer) and another part, a thermopile, which converts heat into an electrical signal.

  Microphone, Loud Speaker, Biological Senses (e.g. touch, sight,..., etc)

transduction

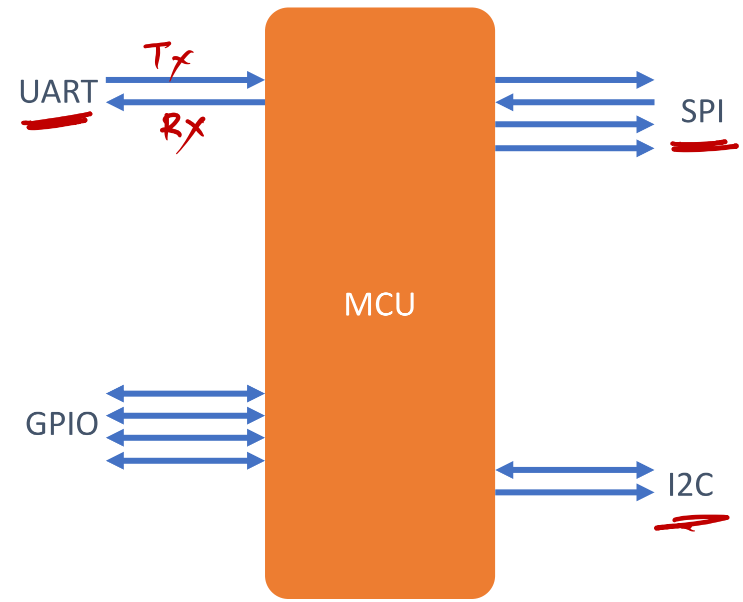
Physical Principles of Sensing

- Charges, fields, & potentials
- Capacitance ⇒ pressure - water level
- Magnetism
- Induction
- Resistance - light ?
- Piezoelectric effect ⇒ Pressure - crack
- Seebeck and Peltier effects
- Thermal properties of materials
- Heat transfer ⇒ Temperature
- Light

Input/Output

I/O defines how an MCU can interact with the environment

- Different I/O protocols are available
- Universal Asynchronous Receiver/Transmitter (UART) – two wires to send/receive data between devices
- Serial Peripheral Interface (SPI) – any number of bits can be sent/received without interruption
- Inter-Integrated Circuit (I^2C) – combines features of UART and SPI
- General purpose I/O (GPIO) – controllable by user at run time. → Nodemcu | RPI



GPIO

- Can be set up to accept or source different logic voltage levels, through which MCU can control peripherals or receive external input/interrupts.
- If pins are configured for interrupts, they can be used to move wake-up from low-power/sleep modes.
- Can be grouped into a GPIO port and controlled as such.
- Pulse-width modulation (PWM) employed when the linear processes must be controlled (e.g., fans)
- Limited to low-current applications → transistors and relays used to help drive higher current loads.

Pulse-width modulation

PWM

Reducing average power output by switching on/off at high rates.

- Duty Cycle – the fraction of one period T during which a signal is active:

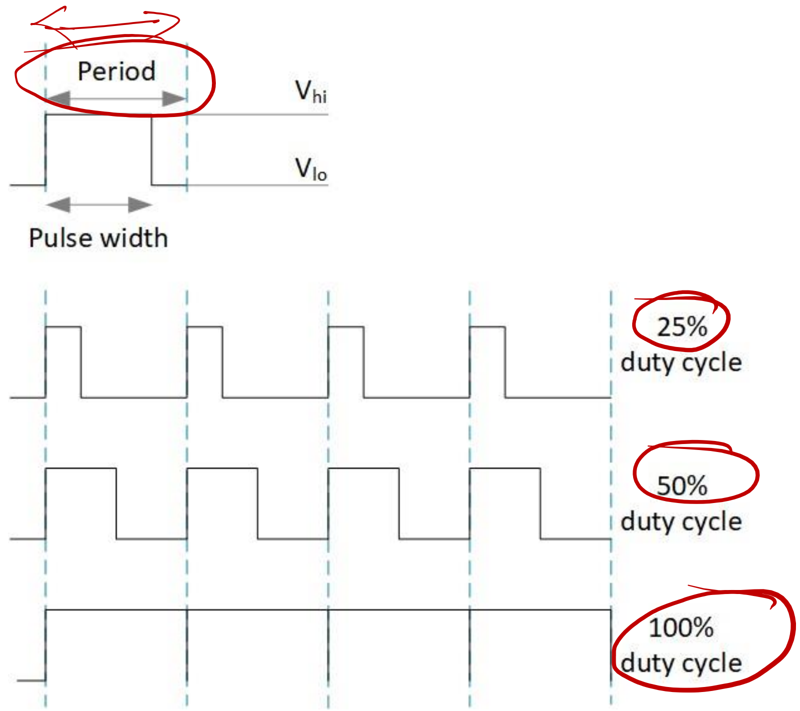
$$D = \frac{PW}{T} \times 100\ [\%]$$

- Frequency – rate of periods:

$$f = \frac{1}{T}\,[\mathrm{Hz}]$$

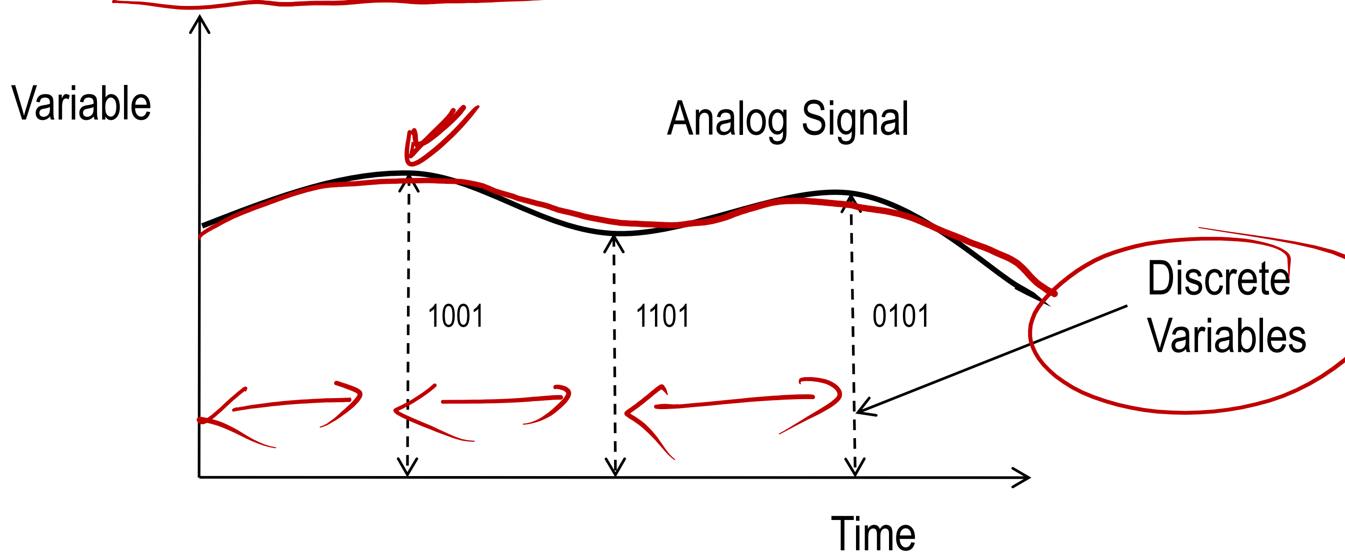
- Average voltage:

$$V_{avg} = V_{hi} \times \frac{D}{100}$$

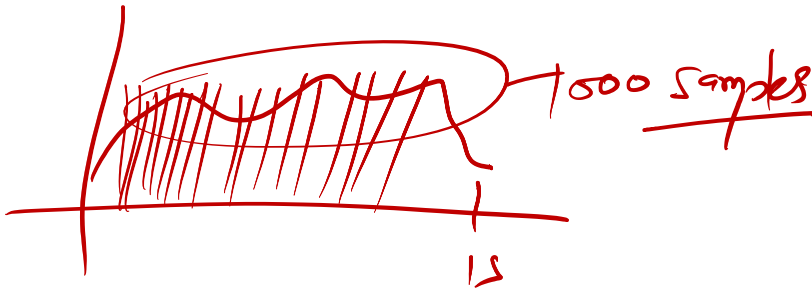


Analog-to-Digital Conversion for MCU

- **Sampling** – converts the continuous signal into a series of discrete analog signals at periodic intervals
- **Quantization** – each discrete analog is converted into one of a finite number of (previously defined) discrete amplitude levels
- **Encoding** – discrete amplitude levels are converted into digital code



Features of an ADC



- **Sampling rate** – rate at which continuous analog signal is polled (e.g., 1000 samples/sec)
- **Quantization** – divide analog signal into discrete levels
- **Resolution** – depends on number of quantization levels
- **Conversion time** – how long it takes to convert the sampled signal to digital code
- **Conversion method** – means by which analog signal is encoded into digital equivalent
    - Example: Successive approximation method

Successive Approximation Method

- A successive approximation ADC is a type of ADC that converts a continuous analog waveform into a discrete digital representation via a binary search through all possible quantization levels before finally converging upon a digital output for each conversion

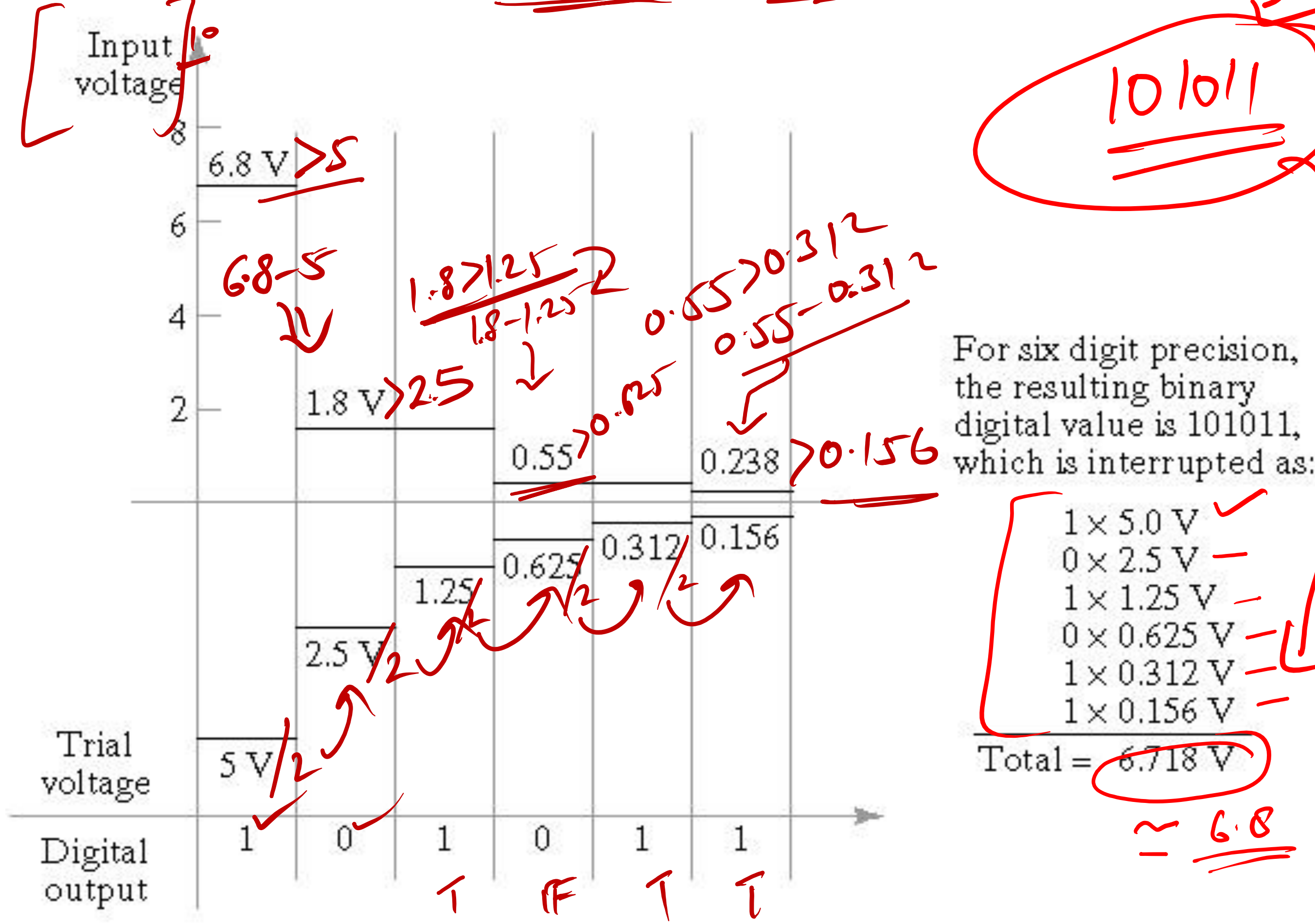
Algorithm

- A series of trial voltages are successively compared to the input signal whose value is unknown
- Number of trial voltages = number of bits used to encode the signal
- First trial voltage is 1/2 the full scale range of the ADC
- If the remainder of the input voltage exceeds the trial voltage, then a bit value of 1 is entered, if less than trial voltage then a bit value of zero is entered
- The successive bit values, multiplied by their respective trial voltages and added, becomes the encoded value of the input signal

Example

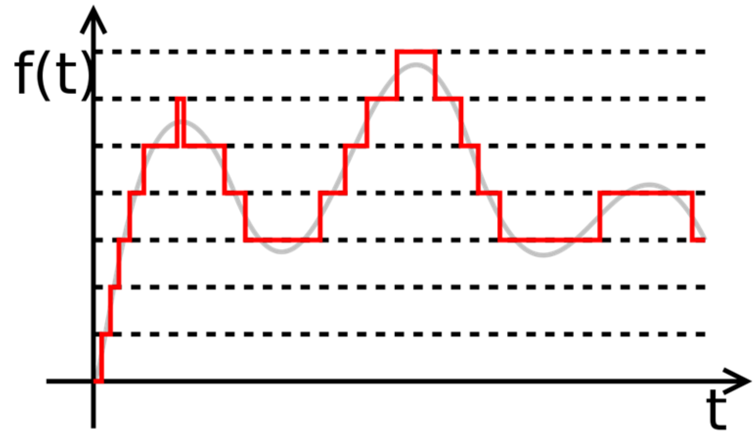
- Analogue signal is 6.8 volts. Encode, using SAM, the signal for a 6 bit register with a full scale range of 10 volts.

$\frac{10}{2} = 5$



Example

$$\boxed{Q = \frac{E_{FSR}}{2^M} = \frac{E_{FSR}}{N}}$$



- Q = resolution in volts per step
- M = resolution in bits
- N = Number of intervals (steps, quantization levels)
- E_{FSR} = Full scale voltage range
- Quantization error = ½ of interval

- Voltage range 0 – 10V; M = 12 bits
- N = 4096 intervals (steps) $= 2^M = 2^{12} = 4K = 4096$
- Q = 2.44 mV/code

$$= \frac{10V}{4096} = \frac{10*1000}{4096} = 2.44 \text{ mV/code}$$

Example

- Using an analogue-to-digital converter, a continuous voltage signal is to be converted into its digital counterpart. The ADC has a 16-bit capacity, and full scale range of 60 V. Determine:
  - number of quantization levels
  - resolution
  - quantization error

N

$\Rightarrow 2^{16} = 64K = 65{,}536$

$\Rightarrow Q = \frac{E}{N} = \frac{60 \times 1000}{65{,}536} =$

$\Rightarrow \frac{R}{2} =$

Solution

(1) Number of quantization levels:

$= 2^{16} = 65{,}536$

(2) Resolution:

$= 60 / 65{,}536 = \sim 0.00092$ Volts

(3) Quantization error:

$= 0.00092/2 = 0.00046$ Volts

Digital-to-Analog Conversion

- Convert digital values into continuous analogue signal
  - Decoding digital value to an analogue value at discrete moments in time based on value within register

$$E_0 = E_{ref}\left\{0.5B_1 + 0.25B_2 + \cdots + \left(2^n\right)^{-1}B_n\right\}$$

Where E_0 is output voltage; E_{ref} is reference voltage; B_n is status of successive bits in the binary register

Example

- A DAC has a reference voltage of 100 V and has 6-bit precision. Three successive sampling instances 0.5 sec apart have the following data in the data register:

Instant	Binary Data
1	101000
2	101010
3	101101

- Output Values:

$$E_{01} = 100\{0.5(1)+0.25(0)+0.125(1)+0.0625(0)+0.03125(0)+0.015625(0)\}$$
$$E_{01} = 62.50V$$
$$E_{02} = 100\{0.5(1)+0.25(0)+0.125(1)+0.0625(0)+0.03125(0)+0.015625(0)\}$$
$$E_{02} = 65.63V$$
$$E_{03} = 100\{0.5(1)+0.25(0)+0.125(1)+0.0625(0)+0.03125(0)+0.015625(0)\}$$
$$E_{03} = 70.31V$$

Sensor Types: HW & SW

- **Hardware**-based sensors
  - Physical components built into a device
  - They derive their data by directly measuring specific environmental properties
- **Software**-based sensors
  - Not physical devices, although they mimic hardware-based sensors
  - They derive their data from one or more hardware-based sensors

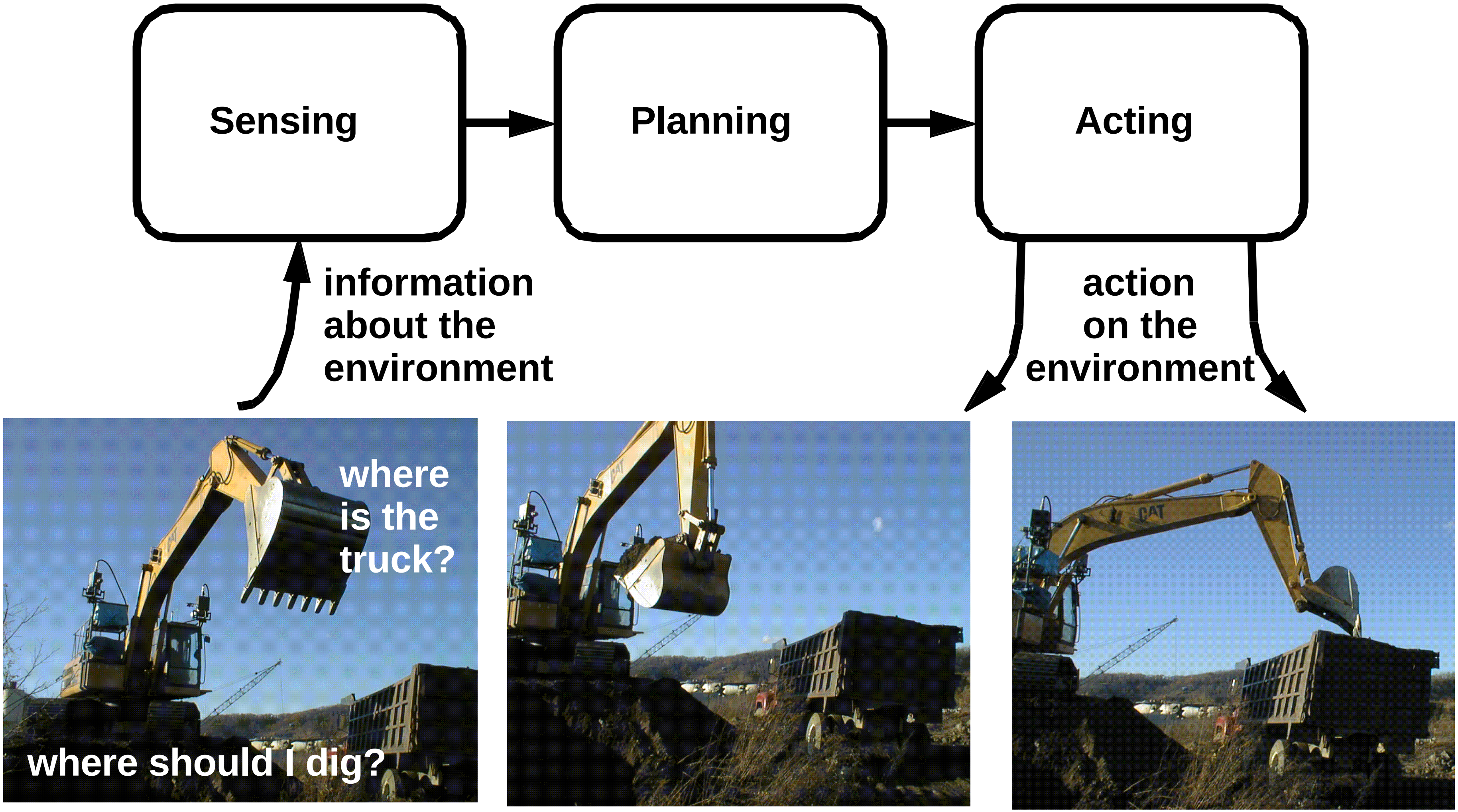
Sensor List of Smartphone

Sensorhub - App

Sensor	Function Type	Software-based or Hardware-based
Accelerometer	Motion Sensor	Hardware-based
Gyroscope	Motion Sensor	Hardware-based
Gravity	Motion Sensor	Software-based
Rotation Vector	Motion Sensor	Software-based
Magnetic Field	Position Sensor	Hardware-based
Proximity	Position Sensor	Hardware-based
GPS	Position Sensor	Hardware-based
Orientation	Position Sensor	Software-based
Light	Environmental Sensor	Hardware-based
Thermometer	Environmental Sensor	Hardware-based
Barometer	Environmental Sensor	Hardware-based
Humidity	Environmental Sensor	Hardware-based

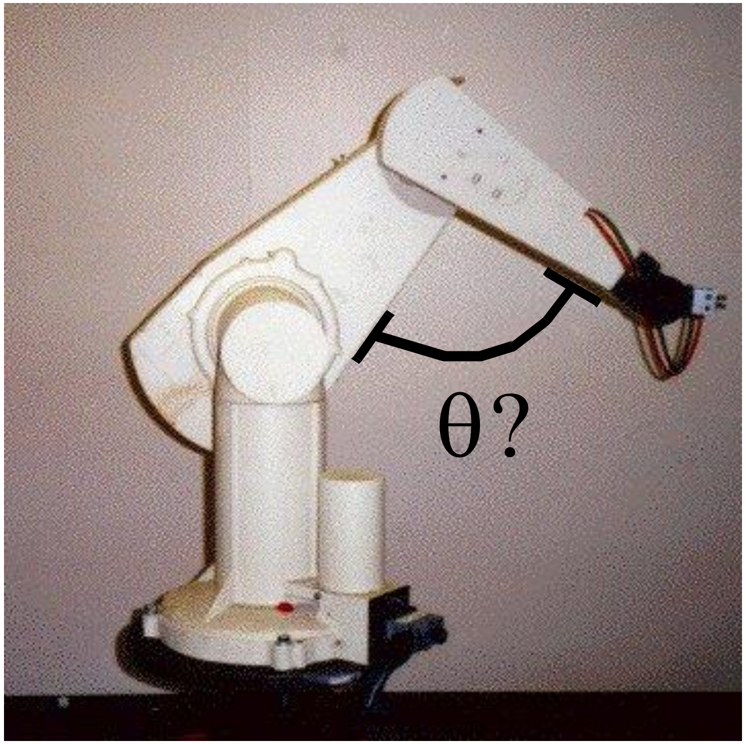
Overview of Our Sensors For Robotics

What makes a machine a robot?



Why do robots need sensors?

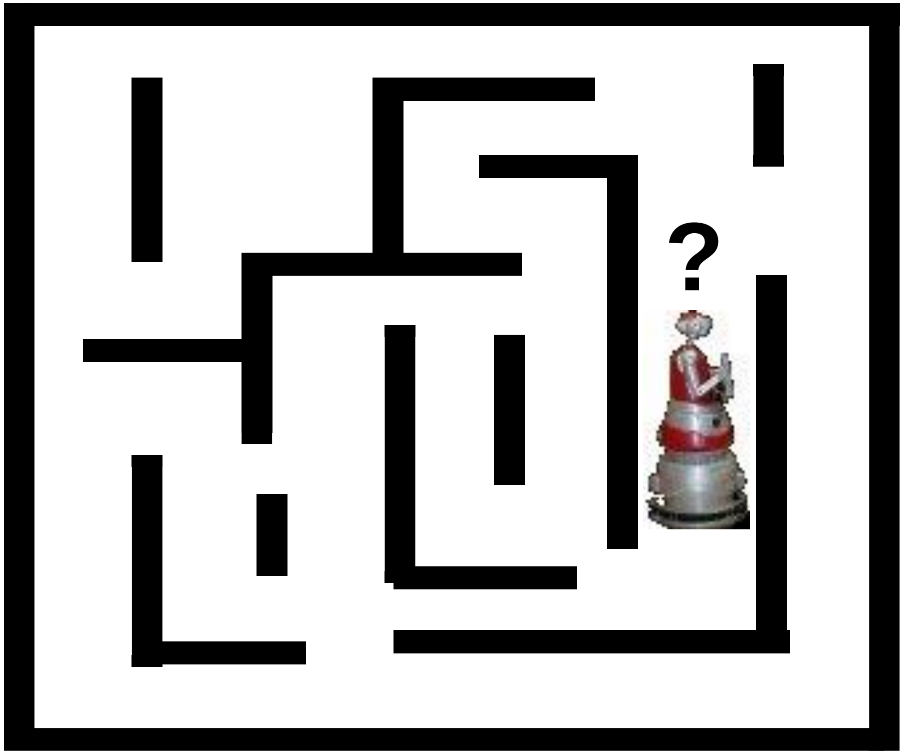
What is the angle of my arm?



internal information

Why do robots need sensors?

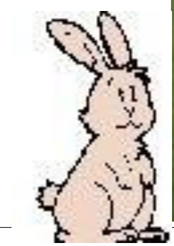
Where am I?



localization

Why do robots need sensors?

Will I hit anything?



obstacle detection

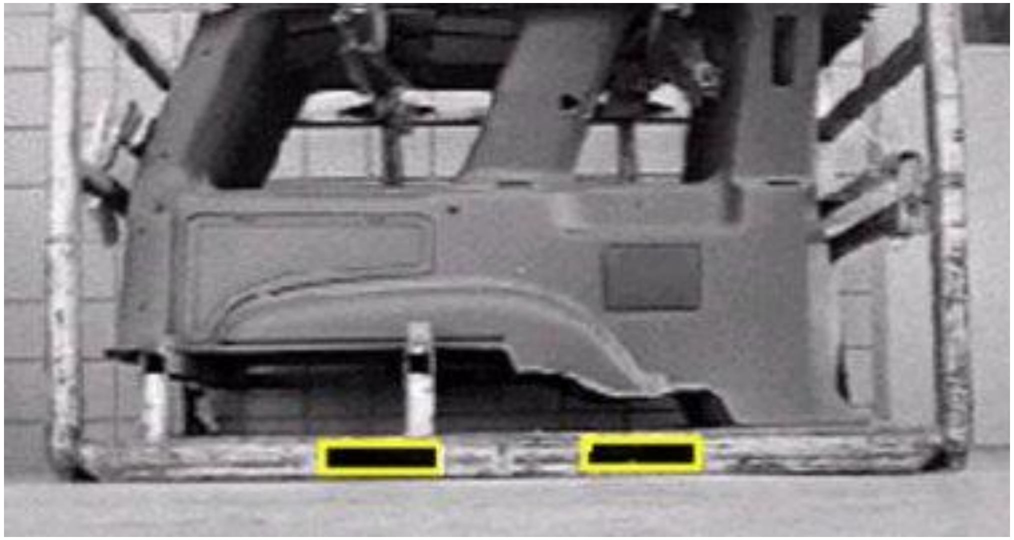
Sensing for specific tasks

Where is the cropline?

Autonomous harvesting

Sensing for specific tasks

Where are the forkholes?



Autonomous material handling

Sensing for specific tasks

Where is the face?

Face detection & tracking

Types of Sensors

- Active
  - send signal into environment and measure interaction of signal w/ environment
  - e.g. radar, sonar
- Passive
  - record signals already present in environment
  - e.g. video cameras

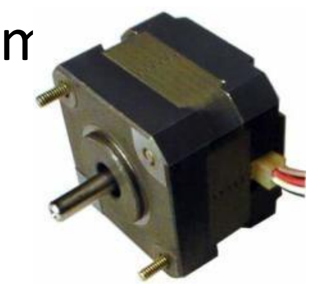
Actuators

- Hardware devices that convert a controller command signal into a change in a physical parameter
  - The change is usually mechanical (e.g., position or velocity)
  - An actuator is also a transducer because it changes one type of physical quantity into some alternative form
  - An actuator is usually activated by a low-level command signal, so an amplifier may be required to provide sufficient power to drive the actuator

Types of Actuators

1. Electrical actuators
    - Electric motors
        - DC servomotors
        - AC motors
        - Stepper motors
    - Solenoids

2. Hydraulic actuators
    - Use hydraulic fluid to amplify the controller comm signal



3. Pneumatic actuators
    - Use compressed air as the driving force

Contact me:

gauravsingal789@gmail.com

Gaurav.singal@nsut.ac.in

www.gauravsingal.in

LinkedIn:

https://www.linkedin.com/in/gauravsingal789/

Twitter: https://twitter.com/gaurav_singal