# Sensors for the IoT

Prof. Fouad HANNA, PhD.
fouad.hanna@heig-vd.ch

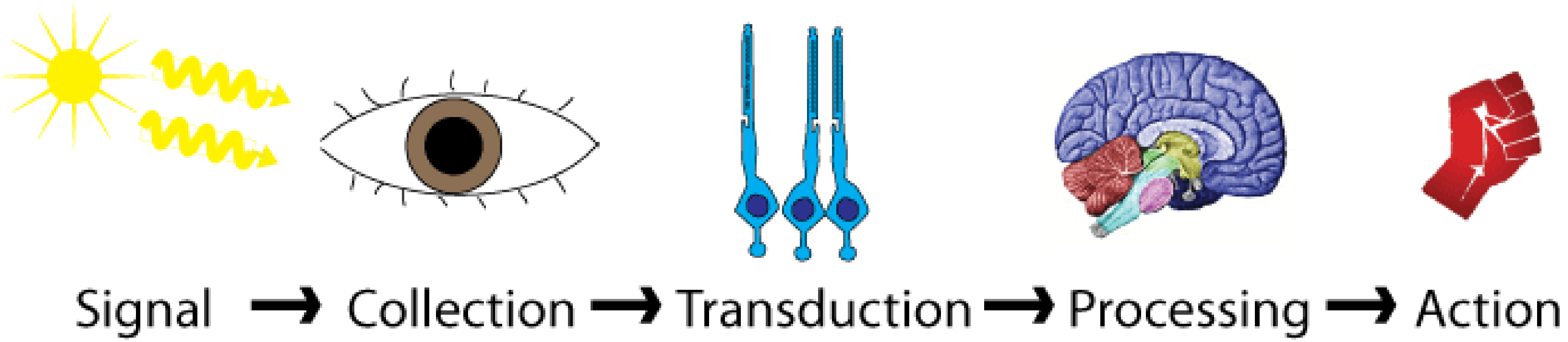
# Table of contents

# Introduction

# Our 5 senses

| Taste | Touch | Sight (vision) | Smell | Hearing - Balance |
|---|---|---|---|---|
| Chemical substance | Pressure, temperature | Light | Chemical substance | Sound - Gravity and acceleration |

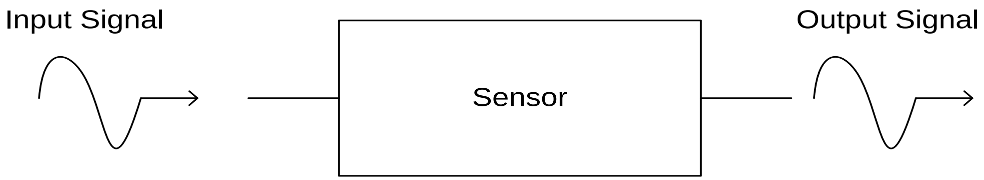
# How sensing works



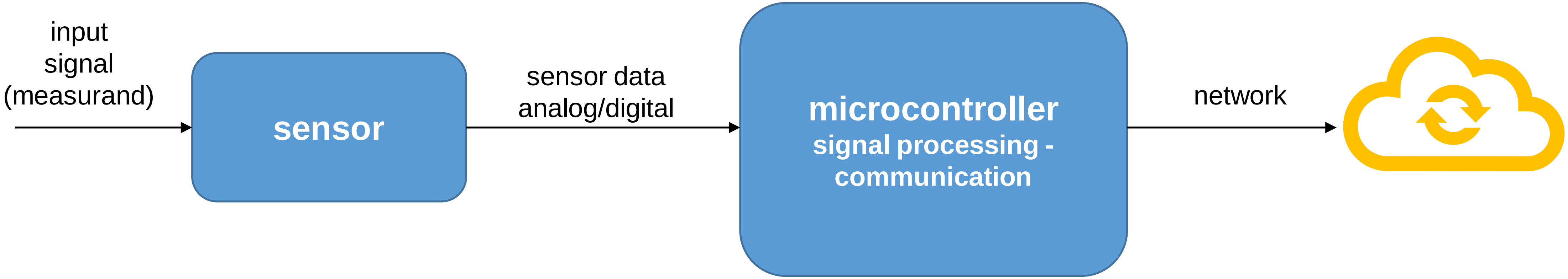
# Transducer – Sensors - Actuators

- Transducer: a device that converts energy (signal) from one form to another.
  - Examples: microphone, loud speaker, piezoelectric element.
  - Mechanical transducers : convert physical quantities into mechanical outputs or vice versa.
  - Electrical transducers: convert physical quantities into electrical outputs or signals.

- Sensor: a special type of transducers which converts the physical parameter of a quantity into corresponding electrical output.
  - i.e. device that detects a change in a physical stimulus and turns it into a signal which can be measured or recorded.
  - Output is suitable for processing.

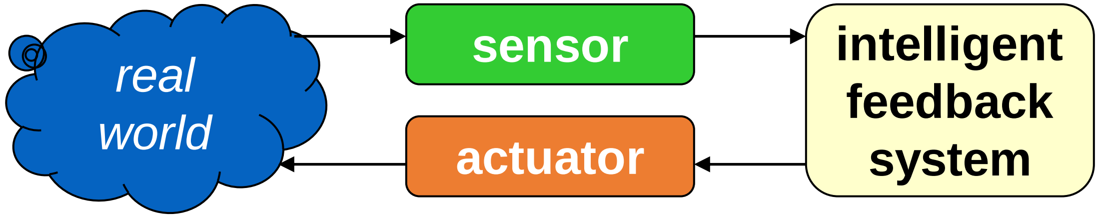


# Transducer – Sensors - Actuators

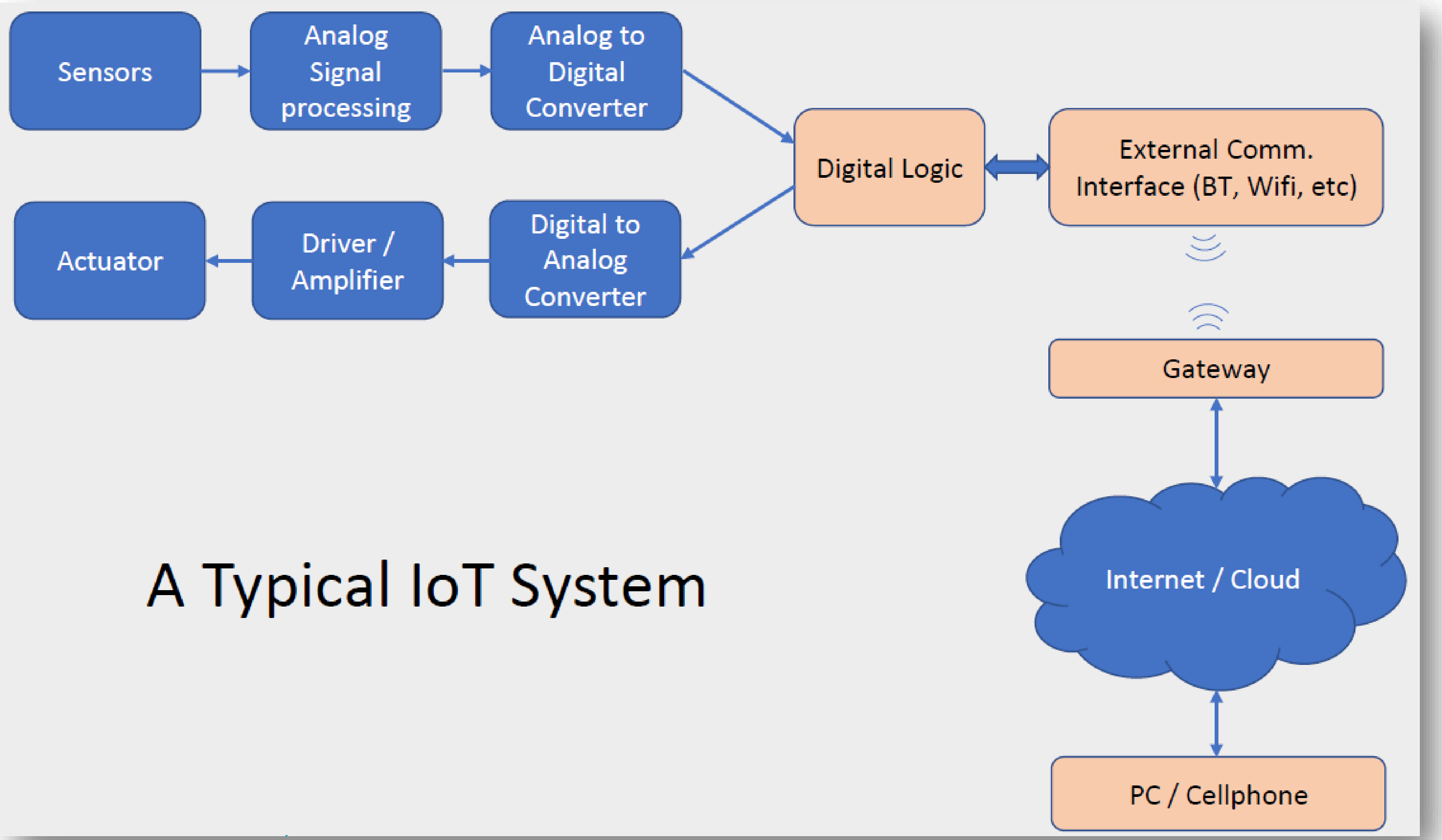


# Transducer – Sensors - Actuators

- Actuator : a device that converts electrical signal into a physical event.
  - Works opposite to sensors.
  - Used to perform output function in a system.
  - E.g. Control an external device



# Typical Electronic Sensor System



# Detectable Phenomenon

| Stimulus | Quantity |
| --- | --- |
| Acoustic | *Wave (amplitude, phase, polarization), Spectrum, Wave Velocity* |
| Biological & Chemical | *Fluid Concentrations (Gas or Liquid)* |
| Electric | *Charge, Voltage, Current, Electric Field (amplitude, phase, polarization), Conductivity, Permittivity* |
| Magnetic | *Magnetic Field (amplitude, phase, polarization), Flux, Permeability* |
| Optical | *Refractive Index, Reflectivity, Absorption* |
| Thermal | *Temperature, Flux, Specific Heat, Thermal Conductivity* |
| Mechanical | *Position, Velocity, Acceleration, Force, Strain, Stress, Pressure, Torque* |

# Need for sensors

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

- No sensors → No automation
- Imagine having to manually fill bottled water !
  - Sensor to detect when the bottle is filled

  OR

  - Sensor to measure the volume of water gone in the bottle.

# Types of sensors

- Active: uses external source of energy, such as electricity or radiation, to emit a signal and detect the response.

Examples of active sensors include radar, lidar …

- Passive: does not require an external source of energy and rely on the natural energy emitted by the object being measured.

Examples of passive sensors include thermal sensors, infrared sensors …

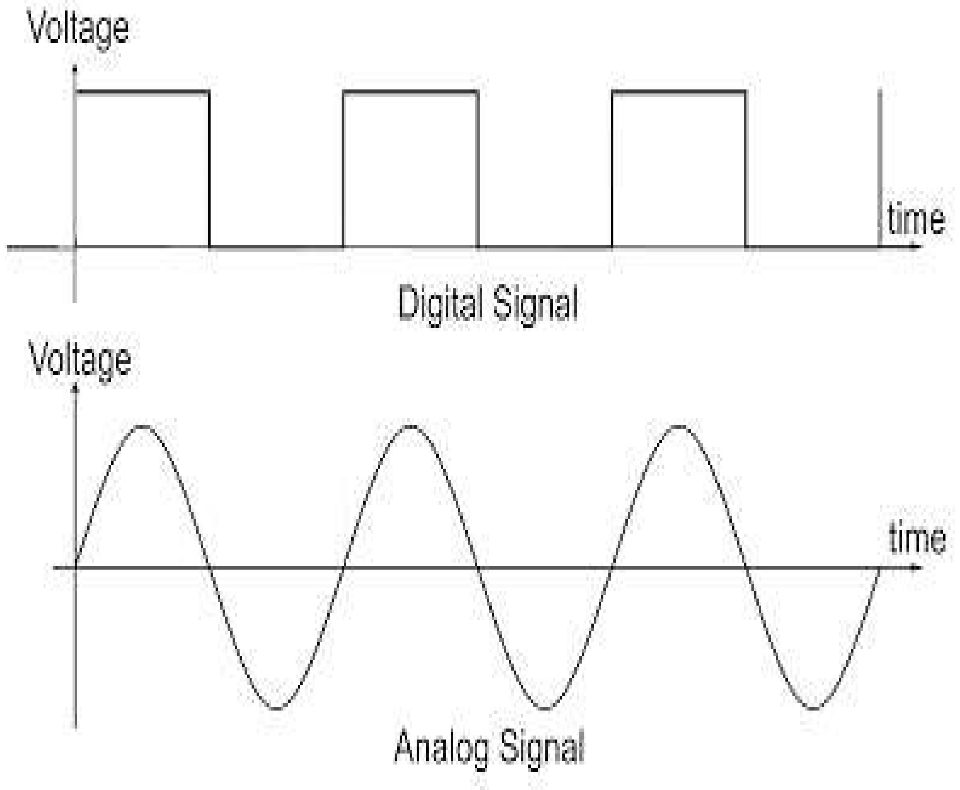
- Digital: the signal produced or reflected by the sensor is binary (digital).

- Analog: The signal produced by the sensor is continuous and proportional to the measurand.

# Digital vs Analog signal

- Digital signal: signal that is continuous in time and assumes only a limited number of values (maintains a constant level and then changes to another constant level).

- Analog signal: signal that is continuous in time and can assume an infinite number of values in a given range (continuous in time and value).



# Choosing a Sensor

| Environmental Factors | Sensor Characteristics | Economic Factors |
|---|---|---|
| Temperature range | Sensitivity | Cost |
| Humidity effects | Range | Availability |
| Corrosion | Stability | Lifetime |
| Size | Repeatability | |
| Ruggedness | Resolution | |
| Power consumption | Error | |
| | Response time | |

# Choosing a Sensor

- Type of Sensing: The parameter that is being sensed like temperature or pressure.

- Operating Principle: The principle of operation of the sensor

- Power Consumption: The power consumed by the sensor will play an important role in defining the total power of the system → IoT important criteria.

- Accuracy: The accuracy of the sensor is a key factor in selecting a sensor.

- Environmental Conditions: The conditions in which the sensor is being used will be a factor in choosing the quality of a sensor → IoT important criteria.

# Choosing a Sensor

- ❑ Cost: Depending on the cost of application, a low cost sensor or high cost sensor can be used → IoT important criteria (scalability).

- ❑ Resolution and Range: The smallest value that can be sensed and the limit of measurement are important.

- ❑ Calibration and Repeatability: Change of values with time and ability to repeat measurements under similar conditions.

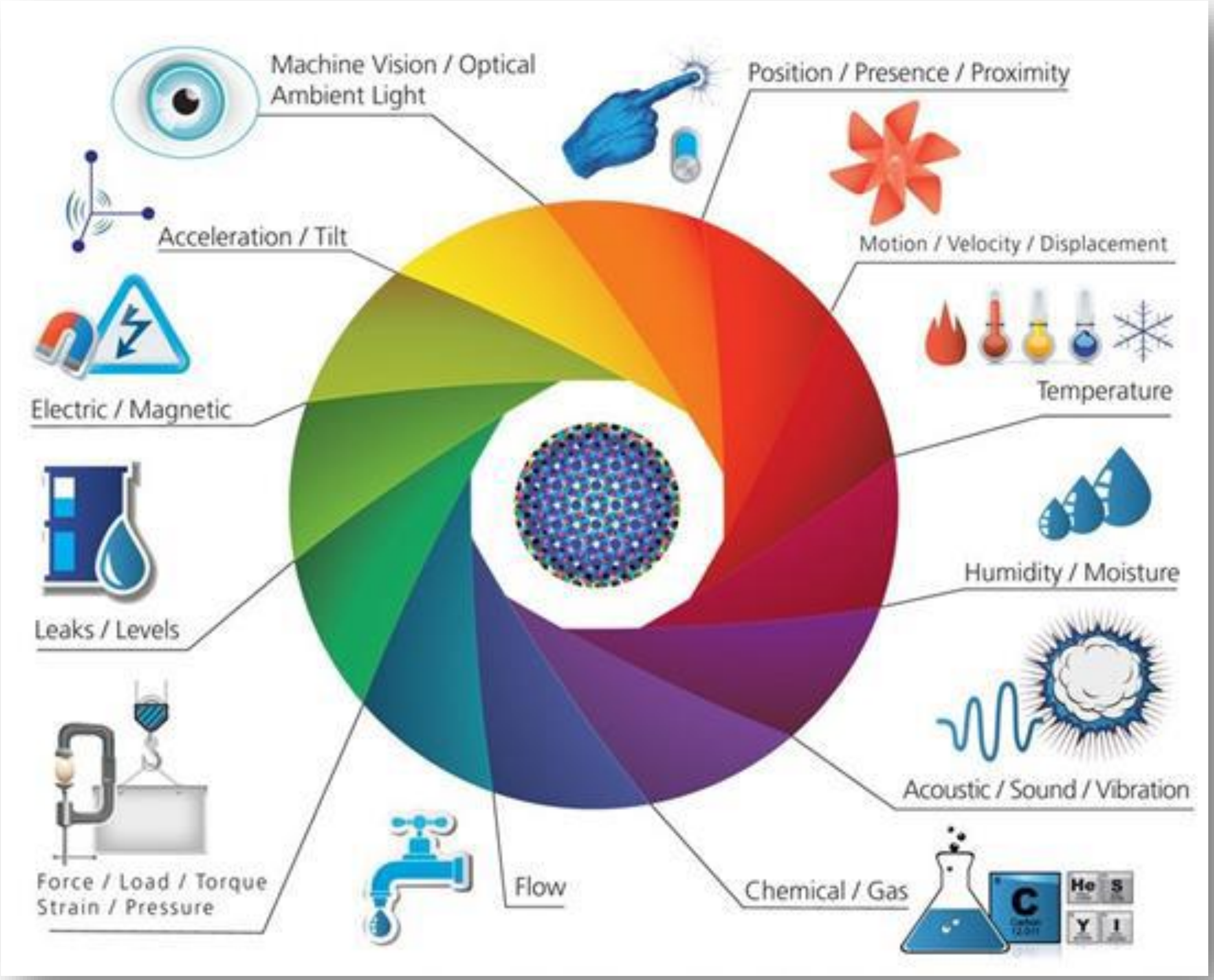
→ In some applications, a single sensor may not be sufficient to do the job.

→ Multiple sensors can be combined.

For example, temperature sensor and vibration sensor data can be used to detect the onset of mechanical failure.

# Basic Sensors

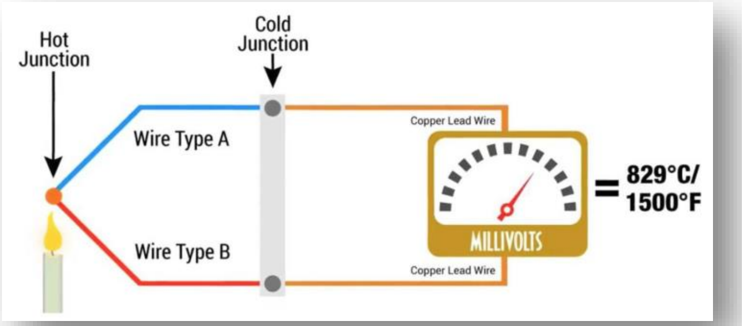
# Basic sensors



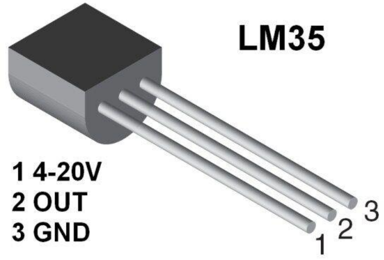

*© 2018 Capgemini - Sensors in IoT System | Pradeep P | 6*

# Temperature Sensors

- ❑ Before talking IoT, where do you see temperature sensors around you ?
- ❑ Two categories:
  - ❑ Contact based.
  - ❑ Non-contact based.
- ❑ Types:
  - ❑ Thermocouples: temperature ←→ voltage.
  - ❑ Resistor Temperature Detectors (RTD) and Thermistors: temperature ←→ resistance.
  - ❑ IC (Integrated circuit): temperature ←→ resistance.
  - ❑ Infrared sensors: temperature ←→ emitted infrared energy intensity.



Thermocouples



LM35 - integrated-circuit temperature sensors

# Humidity Sensors

- ❑ Usually used with temperature sensors → measuring working conditions.

- ❑ Humidity results from the presence of water in the air.

- ❑ Detect changes in electrical currents or air temperature.

- ❑ Types:
  - ❑ Capacitive: electric capacitance of some materials ←→ humidity.
  - ❑ Resistive: electric resistance ←→ humidity.
  - ❑ Thermal: thermal conductivity of air ←→ humidity.

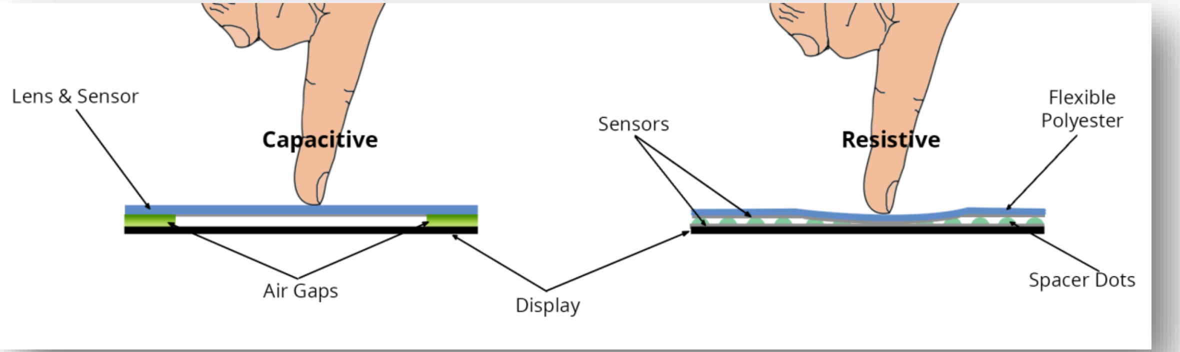
# Touch Sensors

- ❑ Detect touch, force or pressure and convert it into an electric signal.

- ❑ General concept: When there is contact with the surface of the touch sensor, the circuit is closed inside the sensor and there is a flow of current. When the contact is released, the circuit is opened and no current flows.

- ❑ Many applications like mobile phones, remote controls, control panels

- ❑ Types: Capacitive and Resistive.



# Touch Sensors

| | Capacitive | Resistive |
|---|---|---|
| **Activation type** | Light touch (conductive) | Pressure touch |
| **Cost** | Higher manufacturing cost | Lower manufacturing cost |
| **Touch sensitivity** | More sensitive | Less sensitive |
| **Brightness and contrast** | Best | Good |
| **Durability** | Best | Good |
| **Capable of multi-touch** | Yes | No |
| **Can gloves, pen or stylus be used?** | Yes (but Limited) | Yes |
| **Uses** | Multi-touch, accurate applications. | Rugged environments with simple touch features. |

# Accelerometer Sensors

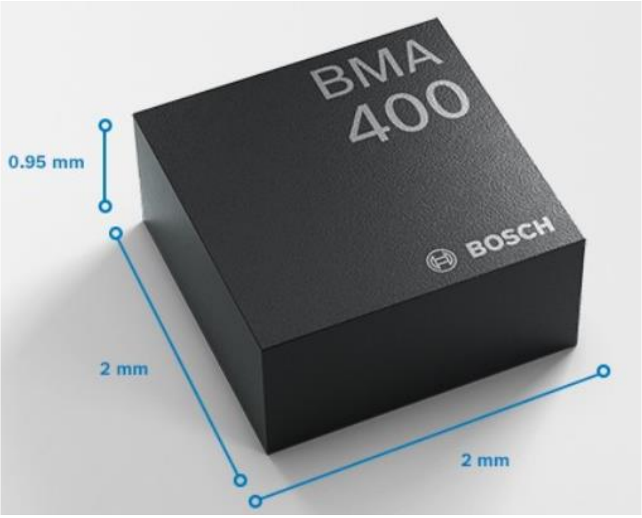
- Accelerometer measure the physical or measurable acceleration experienced by an object and converts the mechanical motion into an electrical output.

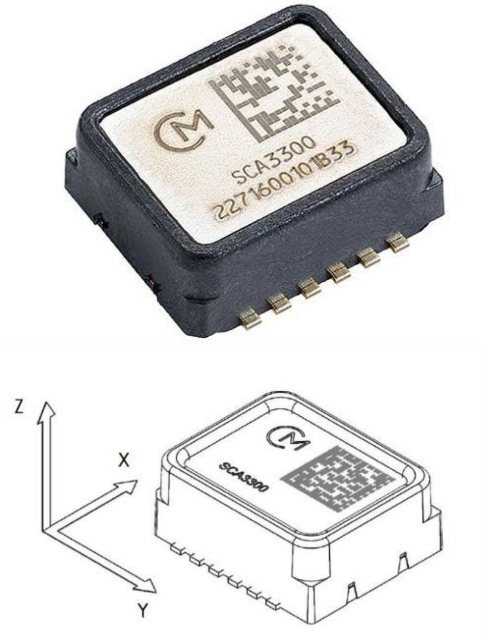
- Unit is g-force: 0g, 0.5g, 1g (earth), 2g, 3g etc … g ~= 9.81 m/s

- Detect vibration, monitor driving fleet, anti-theft (stationary object is moved).

- Different technologies:
  - Hall effect accelerometers
  - Capacitive accelerometers
  - Piezoelectric accelerometers

- Examples: Bosch BMA 400, Murata SCA3300-D01-1

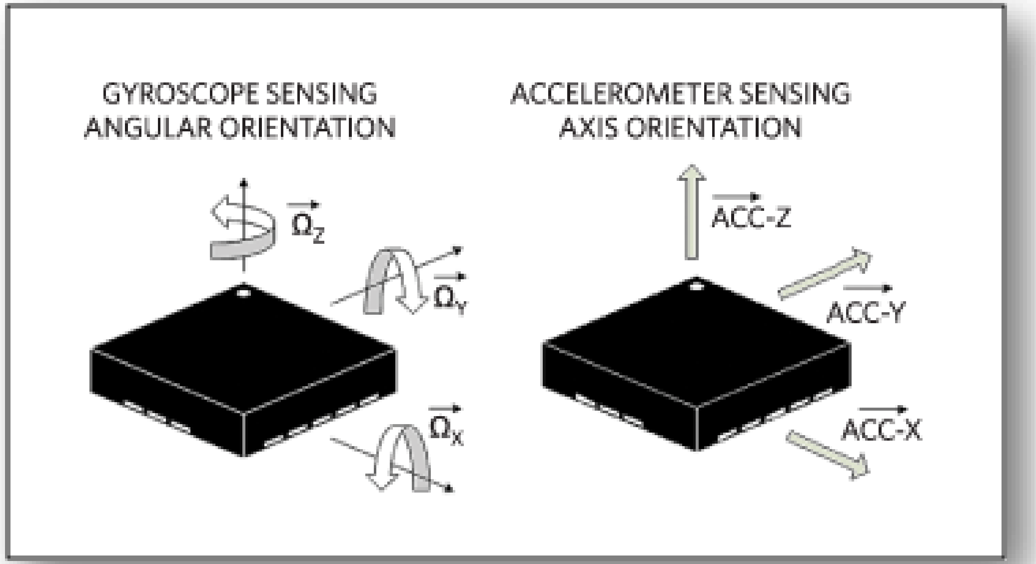



Murata SCA3300-D01-1

# Gyroscope Sensors

- ❑ Orientation, tilt and angular velocity.
- ❑ Main applications: Car navigation systems, Game controllers, Robotics control, Drone & Vehicle control and many more.
- ❑ Most usually, gyroscopes and accelerometers are used together.



# Motion Detection Sensors

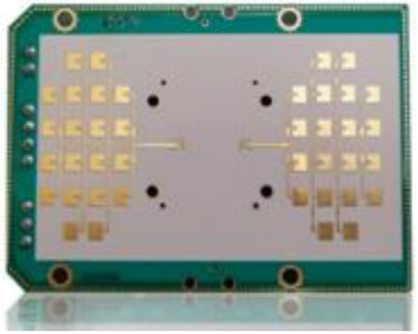
- ❑ Used to detect the physical movement (motion) in a given area.
- ❑ Transforms motion into an electric signal
- ❑ Motion of any object or motion of human beings.
- ❑ Important role in the security industry.

- ❑ Types:
  - ❑ **Passive Infrared (PIR):** It Detects body heat (infrared energy) and the most widely used motion sensor in home security systems.
  - ❑ **Ultrasonic**: Sends out pulses of ultrasonic waves and measures the reflection off a moving object by tracking the speed of sound waves.
  - ❑ **Microwave**: Sends out radio wave pulses and measures the reflection off a moving object. They cover a larger area than infrared & ultrasonic sensors, but they are vulnerable to electrical interference and more expensive.

# Motion Detection Sensors

*Passive Infrared*
*HC-SR501*

*Microwave  AMG061*



*Ultrasonic HC-SR04*


# Other Sensors

- ❑ Gaz and liquid pressure sensors.

- ❑ Chemical Sensors.

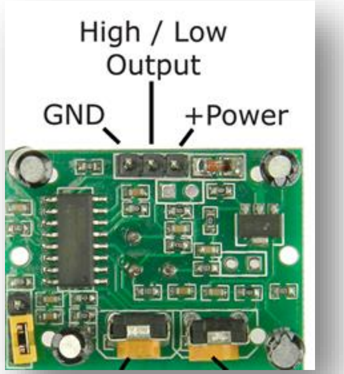
- ❑ Gas Sensors
  - ❑ Gas sensors are similar to the chemical ones but are specifically used to monitor changes of the air quality and detect the presence of various gases.

- ❑ Light sensors.

# Serial communication protocols

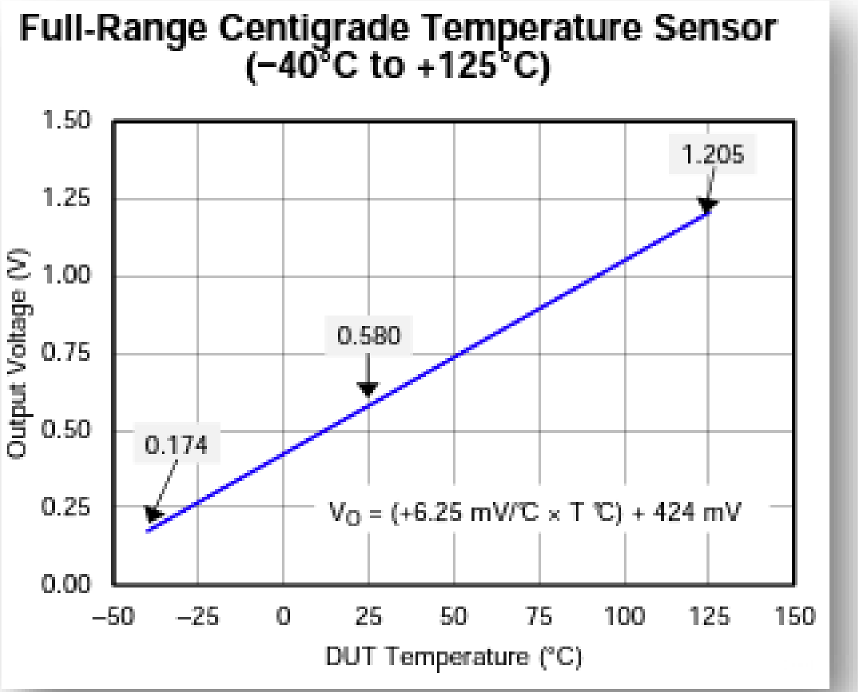
# Sensor Output

- Simple output pin:
  - Digital (all-or-nothing): HC-SR501 PIR motion sensor.
  - Analog: LM60CIZ/NOPB



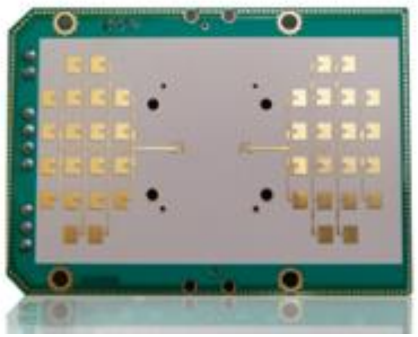

*HC-SR501*

LM60CIZ/NOPB

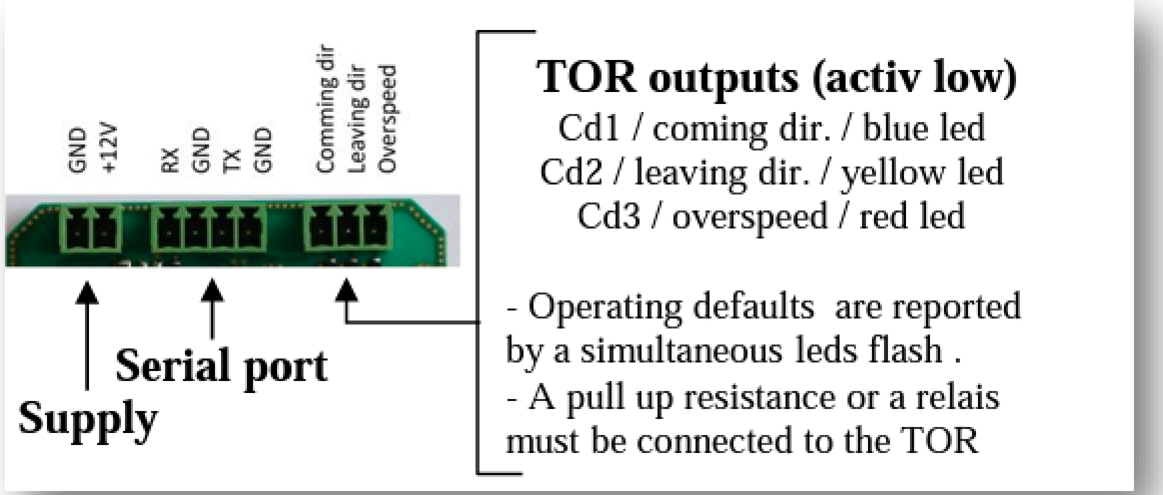


# Sensor Output

*Microwave AMG061*



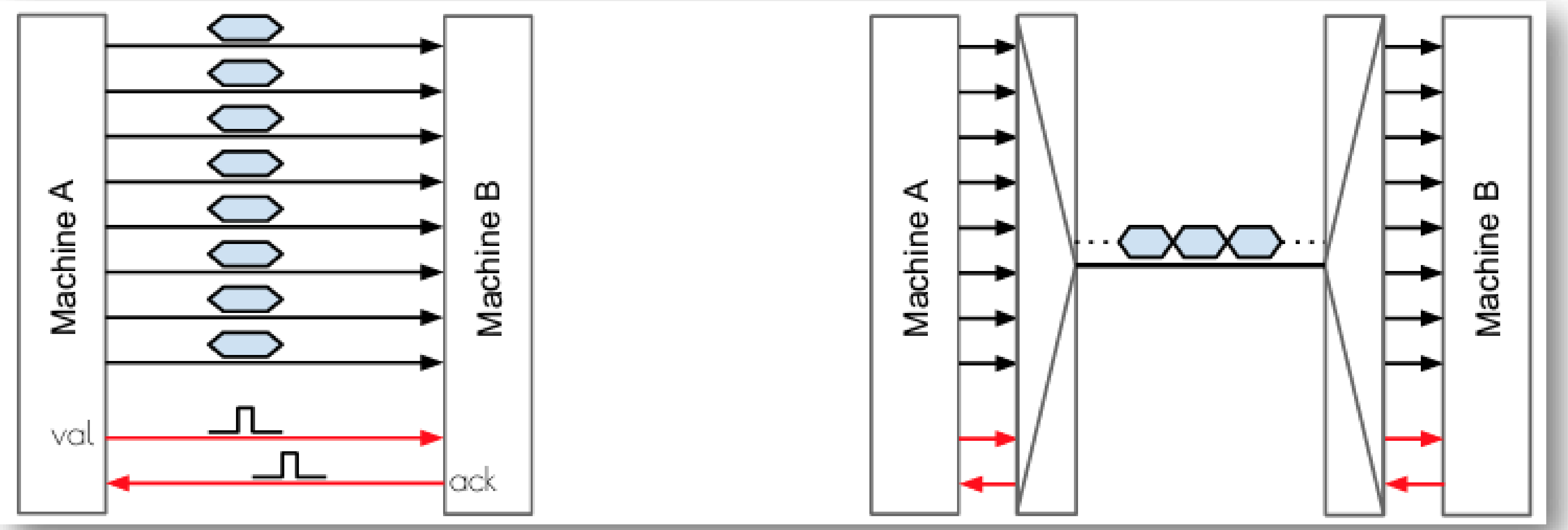
- More sophisticated sensors use serial communication protocols:
  - Microwave AMG061: data frame, UART.




| Communication bus : frame transmitted by the radar | |
|---|---|
| Byte 1 | 0xAA |
| Byte 2 | 0x55 |
| Byte 3 | **NTR : frame N°** |
| Byte 4 | **Cf/St**<br>o Bit 0 : 0 no filtering / 1 filtering<br>o Bit 1 : 0 = coming dir. – 1 = leaving dir.<br>o Bits2 à 7 : free. |
| Byte 5 | **VPE :** Velocity integer part |
| Byte 6 | **VPD** : Velocity Decimal Part |
| Byte 7 | **Signal power** (dBlsb) |
| Byte 8 | **Distance** (m) |
| Byte 9 | **Distance limitation :** programmed value copy |
| Byte 10 | **SV :** speed threshold, programmed value copy |

- What is a communication protocol ?

# Serial vs Parallel Communication




*Parallel Communication*

*Serial Communication*

# Serial vs Parallel Communication

- Parallel communication:
  - In parallel communication, all the bits of data are transmitted simultaneously on separate communication lines.
  - Used for shorter distance.
  - In order to transmit n bit, n wires or lines are used.
  - More costly.
  - Faster than serial transmission.
- Serial communication:
  - In serial communication the data bits are transmitted serially one by one i.e. bit by bit on single communication line.
  - It requires only one communication line rather than n lines to transmit data from sender to receiver.
  - Less costly.
  - Long distance transmission.
  - Asynchronous / Synchronous.

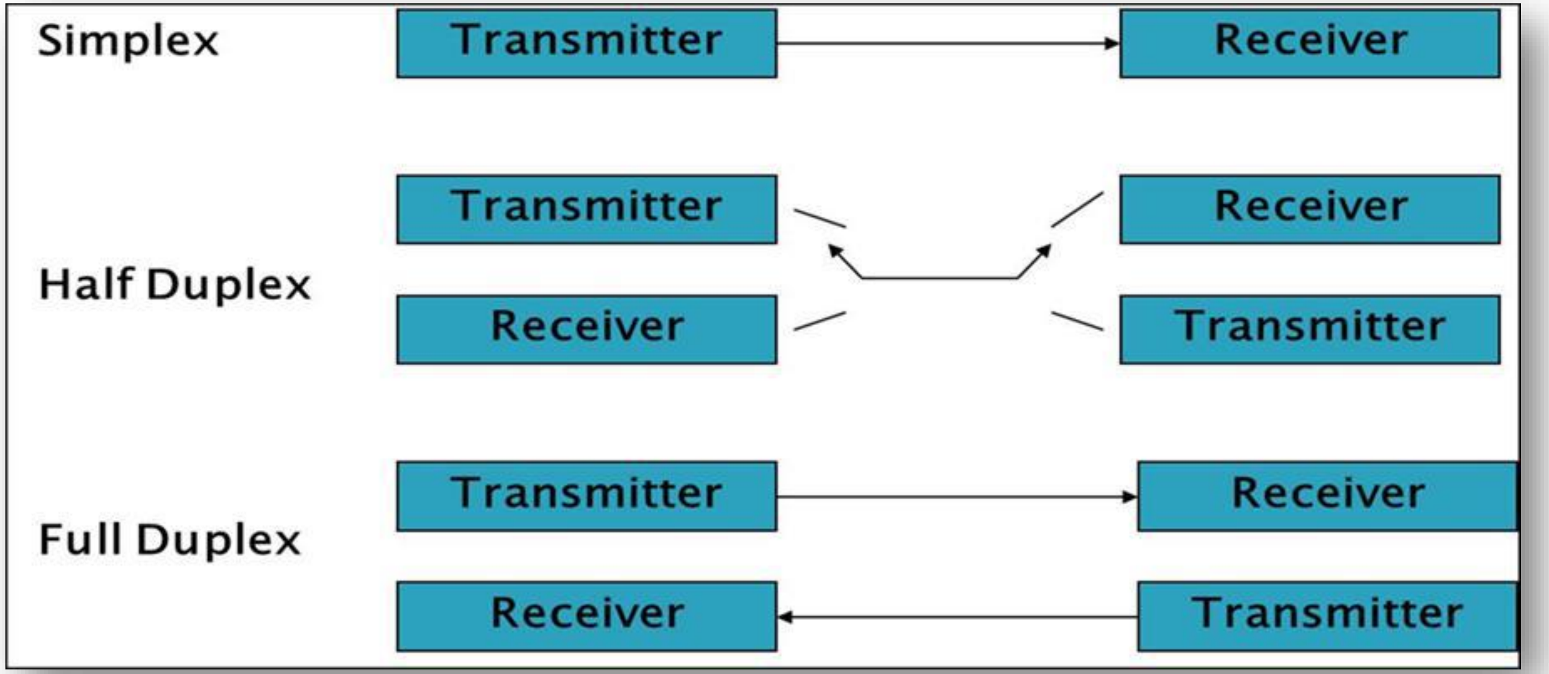
# Serial Communication

- Asynchronous:
  - Transfers single byte at a time
  - No need of clock signal
  - Example: UART (universal asynchronous receiver transmitter)

- Synchronous:
  - Transfers a block of data (characters) at a time
  - Requires clock signal
  - Example: SPI (serial peripheral interface),
    I2C (inter integrated circuit).

- Simplex, Half duplex and full duplex.

# Simplex, Half duplex and full duplex




- Simplex: Data is transmitted in only one direction i.e. from TX to RX only one TX and one RX only
- Half duplex: Data is transmitted in two directions but only one way at a time i.e. two TX's, two RX's and one line
- Full duplex: Data is transmitted both ways at the same time i.e. two TX's, two RX's and two lines

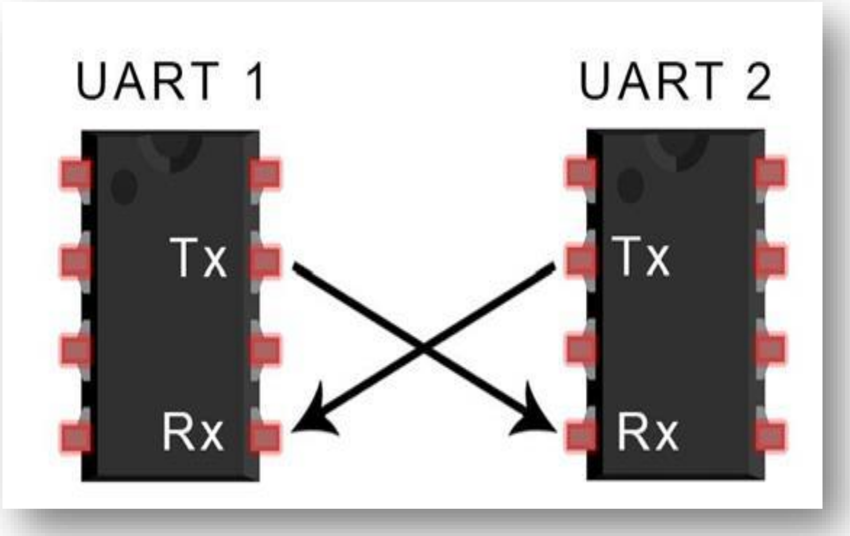
# Comparing serial communication protocols

| Serial protocol | Asynchronous / Synchronous | Type | Duplex | Data transfer rate (kbps) |
|---|---|---|---|---|
| UART | Asynchronous | Peer to peer | Full duplex | < 115.2, 9.6 |
| SPI | Synchronous | Multi-Peripheral | Full duplex | >1,000 |
| I2C | Synchronous | Multi-Peripheral<br>Multi-Controller | Half duplex | 100, 400, 3400, 5000 |

*Multi – Controller: you can connect multiple Peripherals to a single Controller and you can have multiple Controllers controlling single, or multiple Peripherals.*

# UART

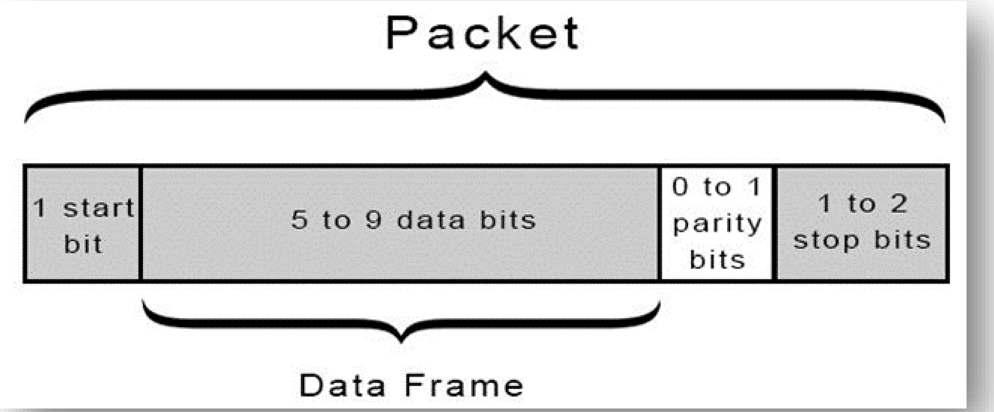
- ❑ In UART communication, two UARTs communicate directly with each other.
- ❑ Only two wires are needed to transmit data between two UARTs.




- ❑ Asynchronous:
  - ❑ no clock signal to synchronize the output of bits
  - ❑ start and stop bits are added to the data packet
  - ❑ Both UARTs must operate at the same baud rate
- ❑ Data rate: < 115200 baud (bps), 9600 baud.
- ❑ Both UARTs must use the same data packet structure

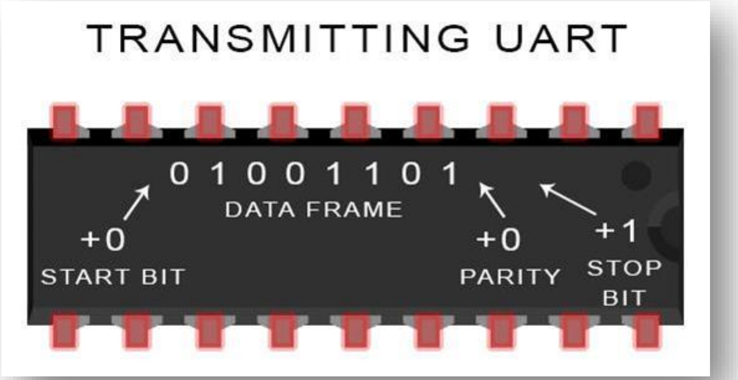
# UART – Packet frame

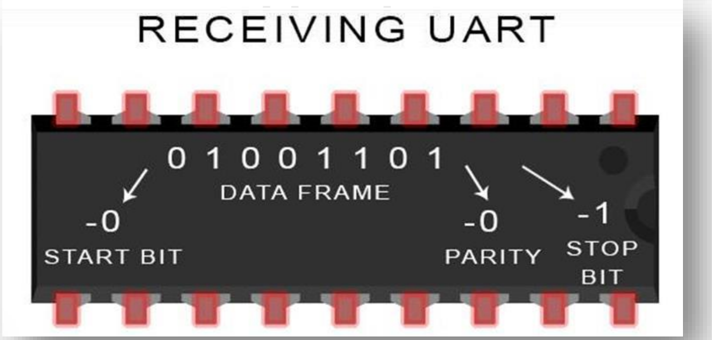
- START BIT:
  - No transmission → line held at a high (1)
  - Transmission start → line goes from high to low (1→0)
- DATA FRAME:
  - actual data being transferred
  - In most cases, data is sent with the least significant bit first.
- PARITY BIT:
  - Optional, detect transmission errors.
  - Parity mode is set at the beginning of communication.
    - even parity: total number of 1 bits, including the parity bit, is even
    - odd parity: total number of 1 bits , including the parity bit, is odd
- STOP BITS:
  - marks the end of the data packet, often 1 bit
  - end transmission → maintain the data line at high (1)

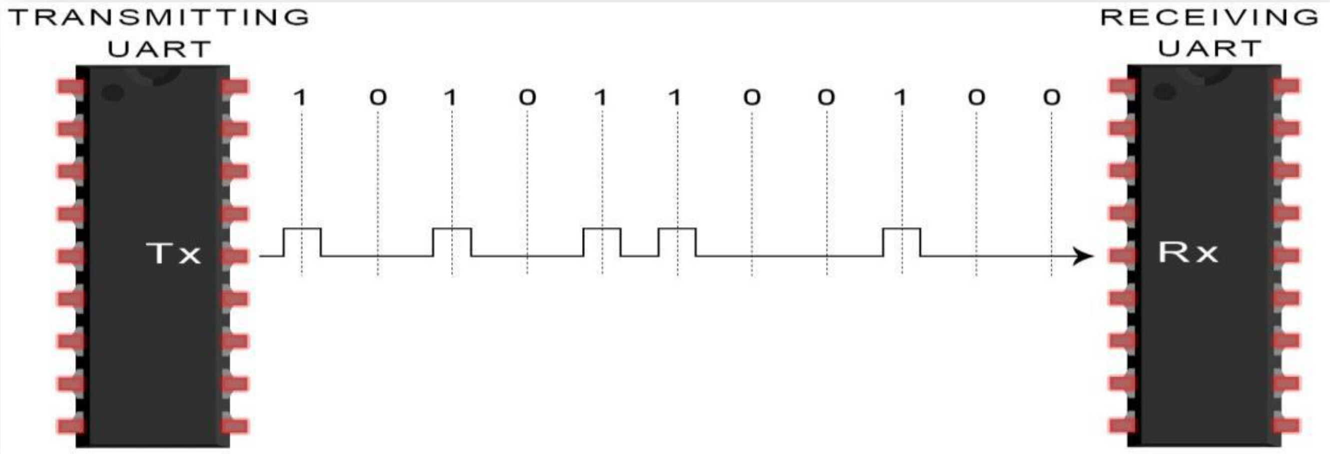


# UART – Transmission

- ❑ The transmitting UART adds the start bit, parity bit, and the stop bit(s) to the data frame.
- ❑ The entire packet is sent serially from the transmitting UART to the receiving UART. The receiving UART samples the data line at the pre-configured baud rate
- ❑ The receiving UART discards the start bit, parity bit, and stop bit from the data frame

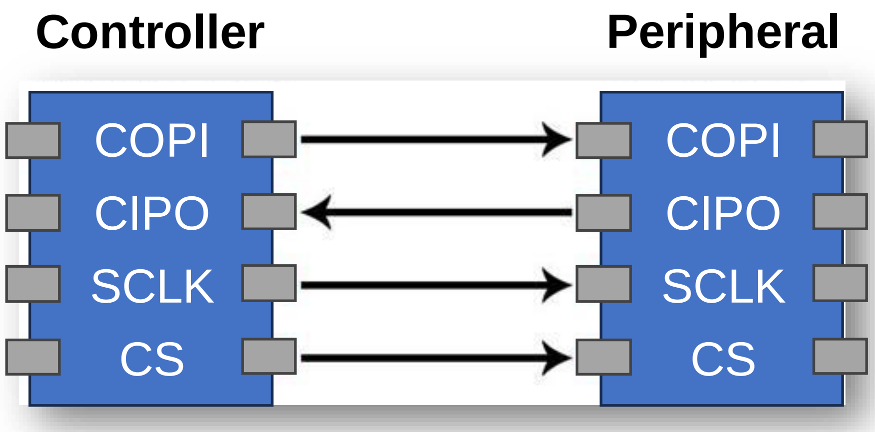


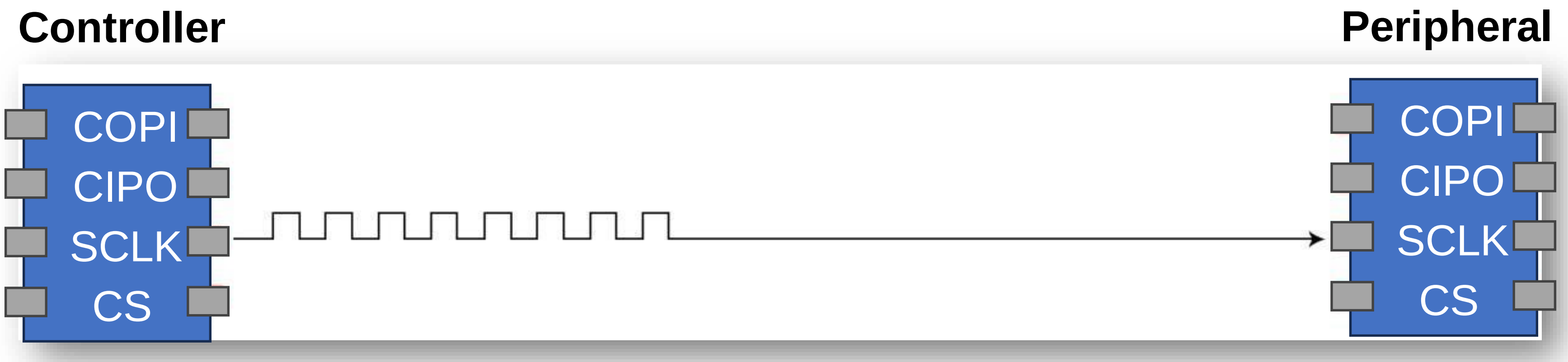
# SPI

- Controller-Peripheral relationship:
  - Controller, usually a microcontroller
  - Peripheral, usually a sensor, display or memory chip
  - A controller can control several peripherals
- Up to 10 Mbps
- Synchronous.
- Data transfer is continuous, no stop.
- 4 wires used:
  - **COPI (Controller Output/Peripheral Input)** – Line for the controller to send data to the peripheral (MSB bit first).
  - **CIPO (controller Input/peripheral Output)** – Line for the peripheral to send data to the controller (LSB bit first).
  - **SCLK (Clock)** – Line for the clock signal (One bit of data is transferred in each clock cycle).
  - **CS (Chip Select)** – Line for the controller to select which peripheral to send data to (low to activate peripheral).

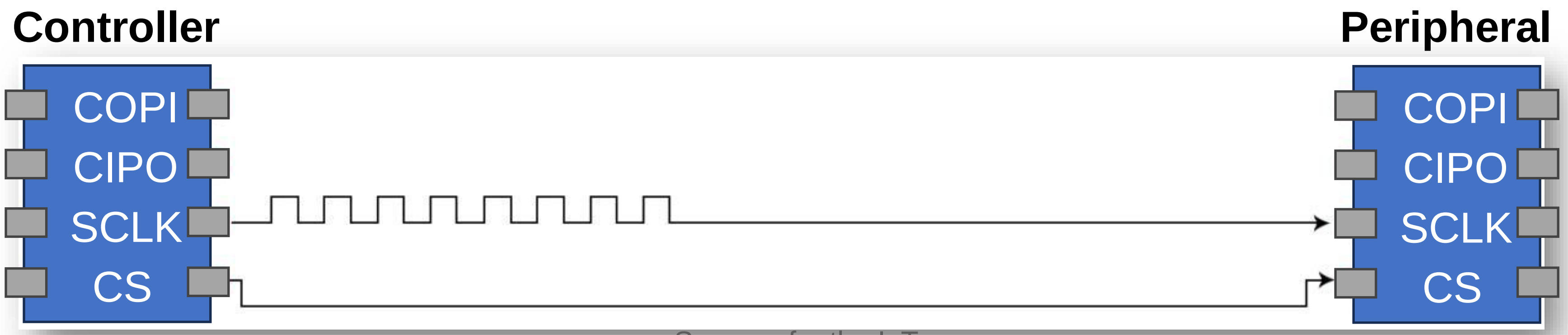


# SPI - Transmission

- ❑ The controller outputs the clock signal

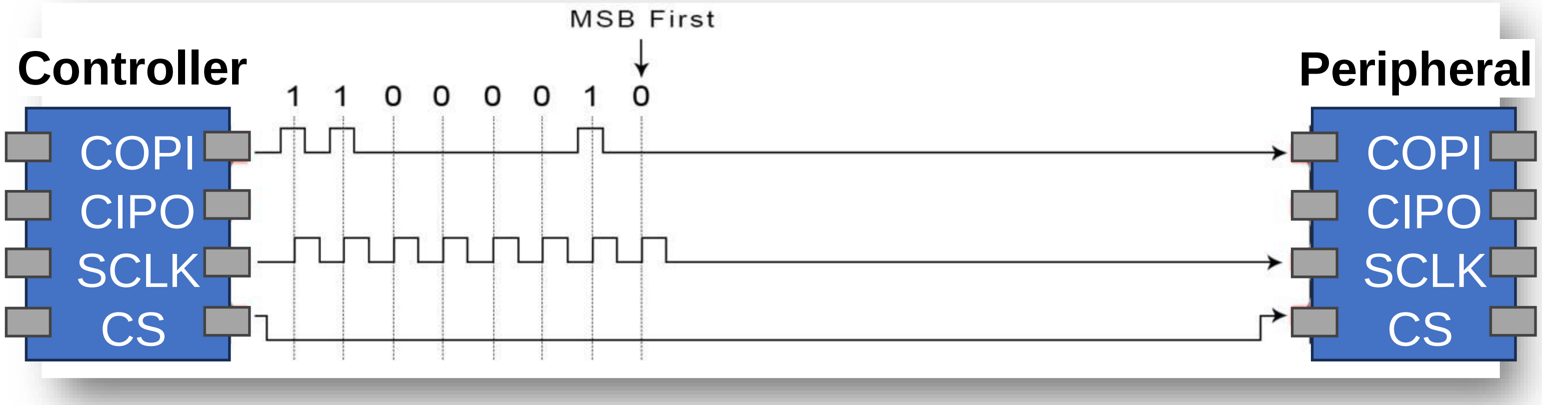



- ❑ The controller switches the SS/CS pin to a low voltage state, which activates the peripheral

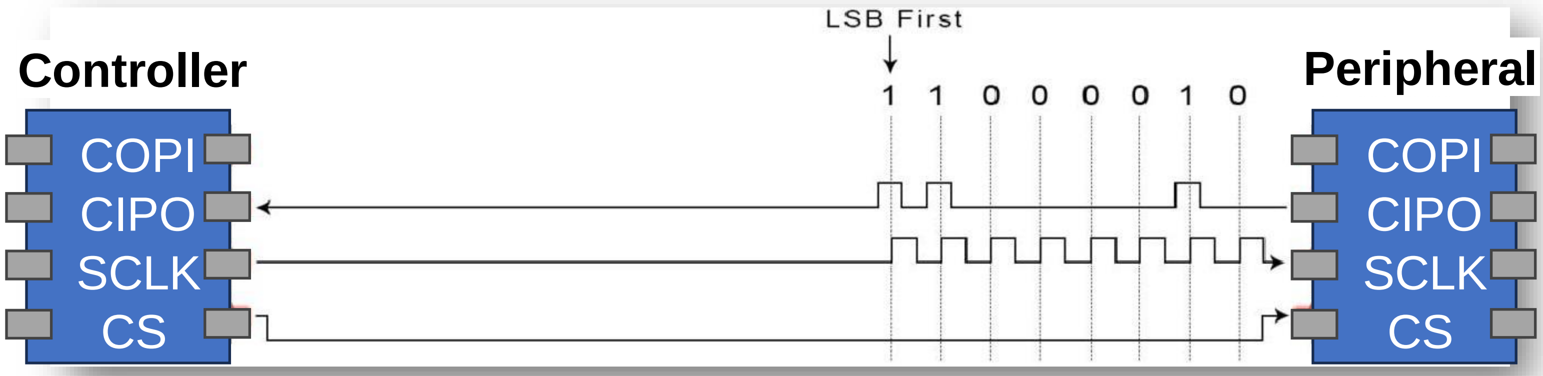


# SPI - Transmission

- ❑ The controller sends the data one bit at a time to the peripheral along the MOSI line.




- ❑ If a response is needed, the peripheral returns data one bit at a time using MISO line.

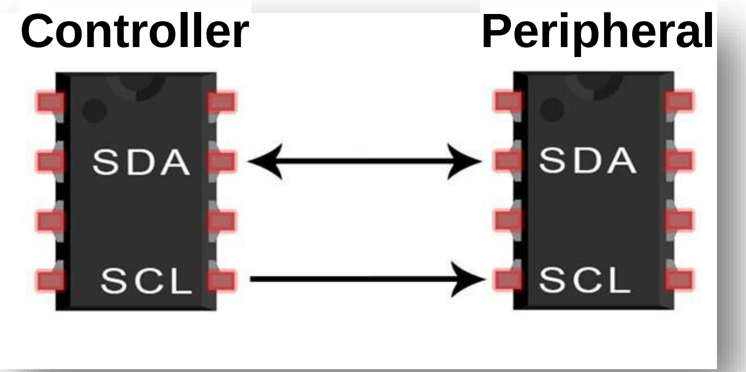


# SPI - Disadvantages

- ❑ Uses four wires (I2C and UARTs use two)
- ❑ No acknowledgement that the data has been successfully received (I2C has this)
- ❑ No form of error checking like the parity bit in UART
- ❑ Only allows for a single controller

# I2C

- I2C combines the best features of SPI and UART:
  - multiple controllers controlling single, or multiple peripherals (+ SPI)
  - uses two wires for data transfer (+ UART)

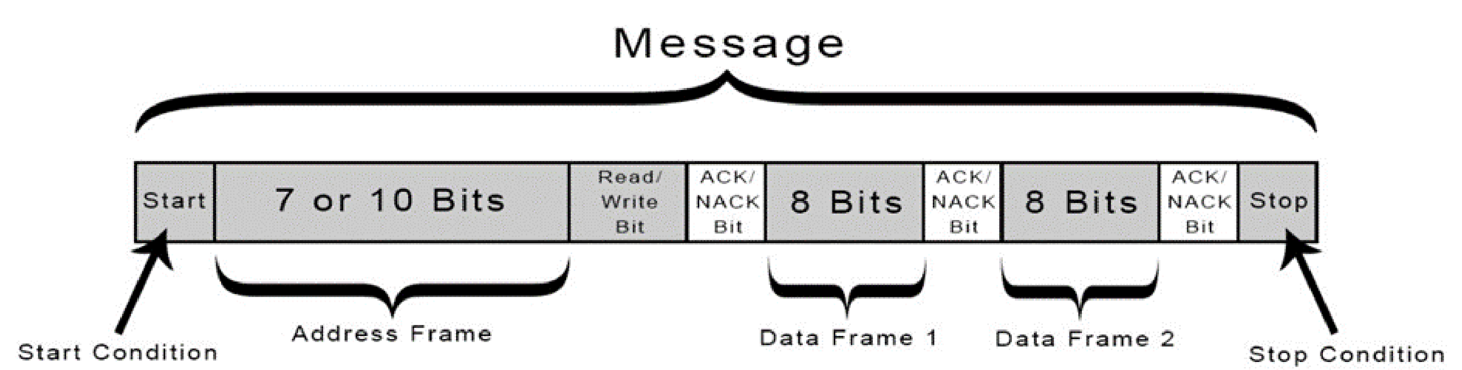



- 100 kbps, 400 kbps, 3.4 Mbps, 5 Mbps.
- Synchronous.

- 2 wires used:
  - **SDA (Serial Data) –** The line for the controller and peripheral to send and receive data.
  - **SCL (Serial Clock) –** The line that carries the clock signal.

- Data is transferred in messages. Messages are broken up into frames of data

## I2C – Message frame



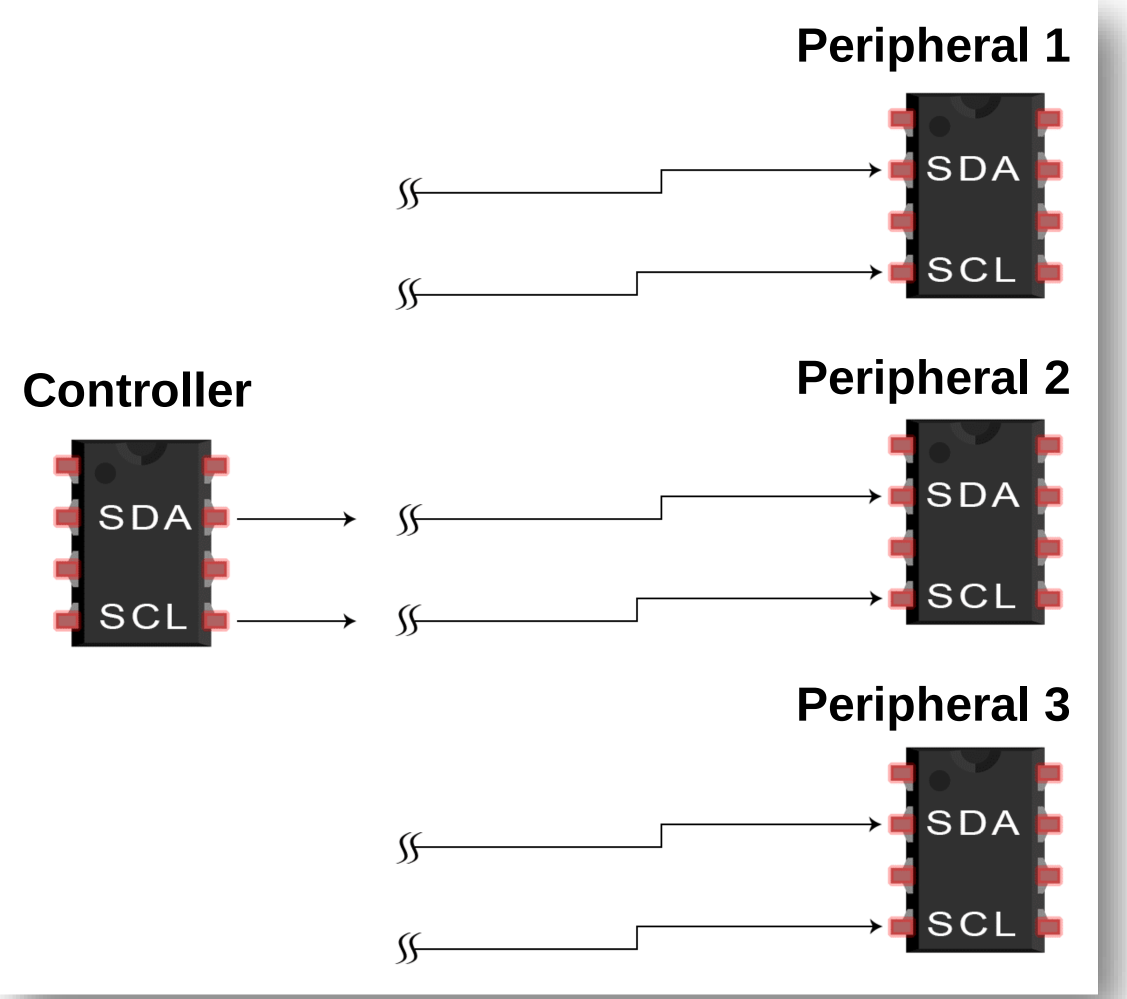

- ❑ Start Condition: SDA line high → low, then SCL line high → low.
- ❑ Stop Condition: SCL line low → high, then SDA line low → high.

- ❑ Address Frame: 7 or 10 bit to identify peripherals (implementation dependent, 7 in most cases).
- ❑ Read/Write Bit:
  - ❑ controller sending to peripheral: low
  - ❑ controller requesting from peripheral: high.

- ❑ ACK/NACK Bit: Each frame in a message is followed by an acknowledge/no-acknowledge bit. If an address frame or data frame was successfully received, an ACK bit is returned to the sender from the receiving device.

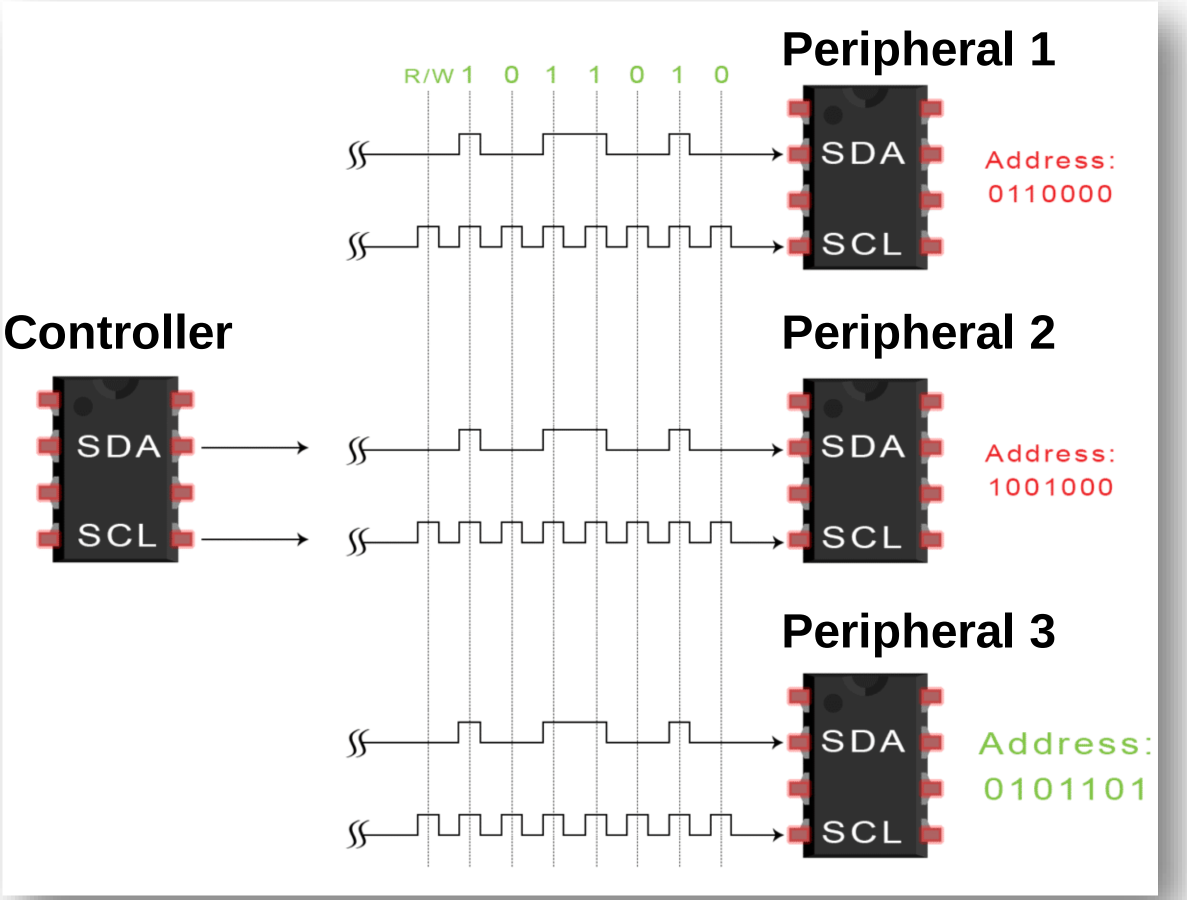
# I2C - Transmission

- ❑ The controller sends the start condition to every connected peripheral by switching the SDA line from a high voltage level to a low voltage level before switching the SCL line from high to low



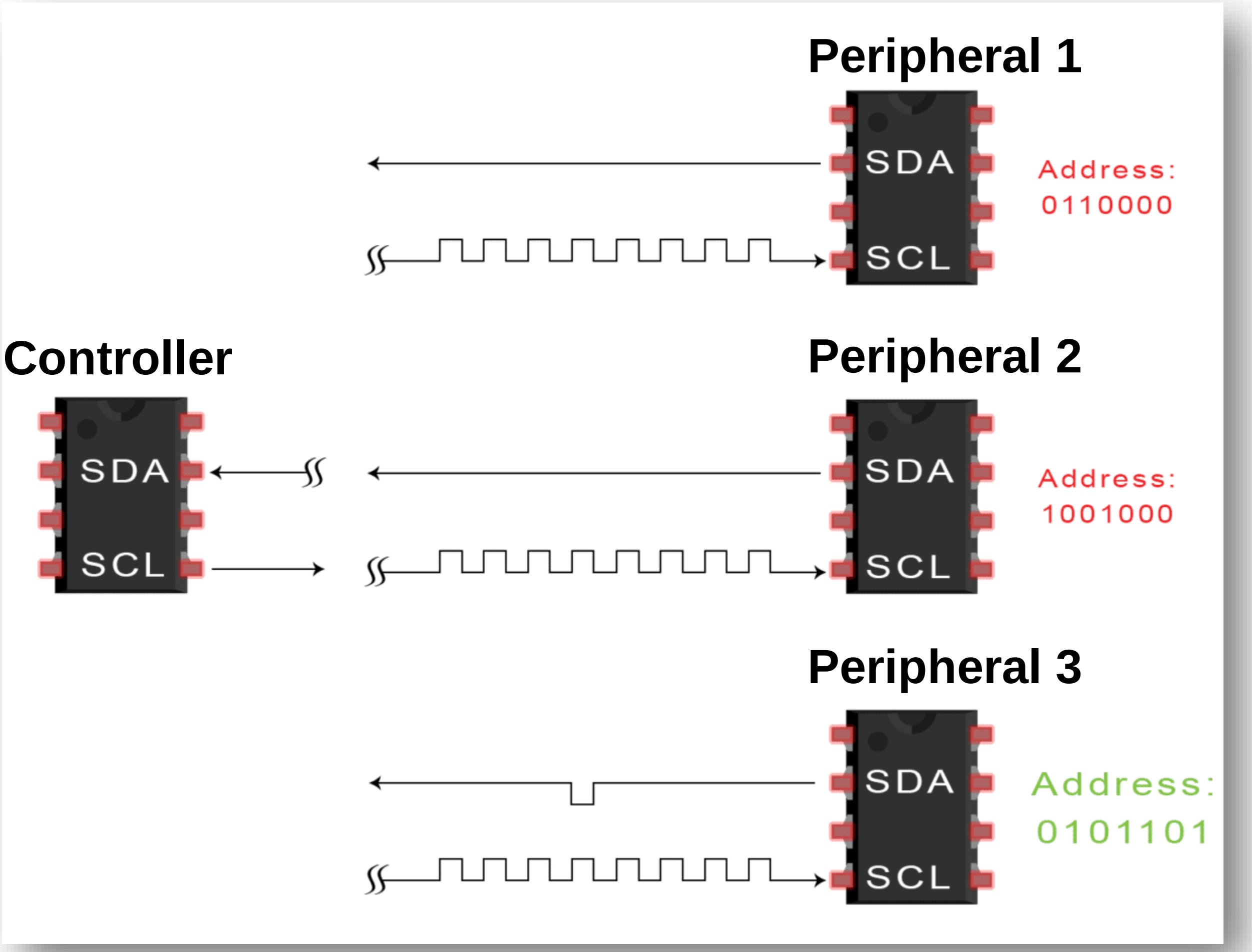
# I2C - Transmission

- ❑ The controller sends each peripheral the 7 or 10 bit address of the peripheral it wants to communicate with, along with the read/write bit



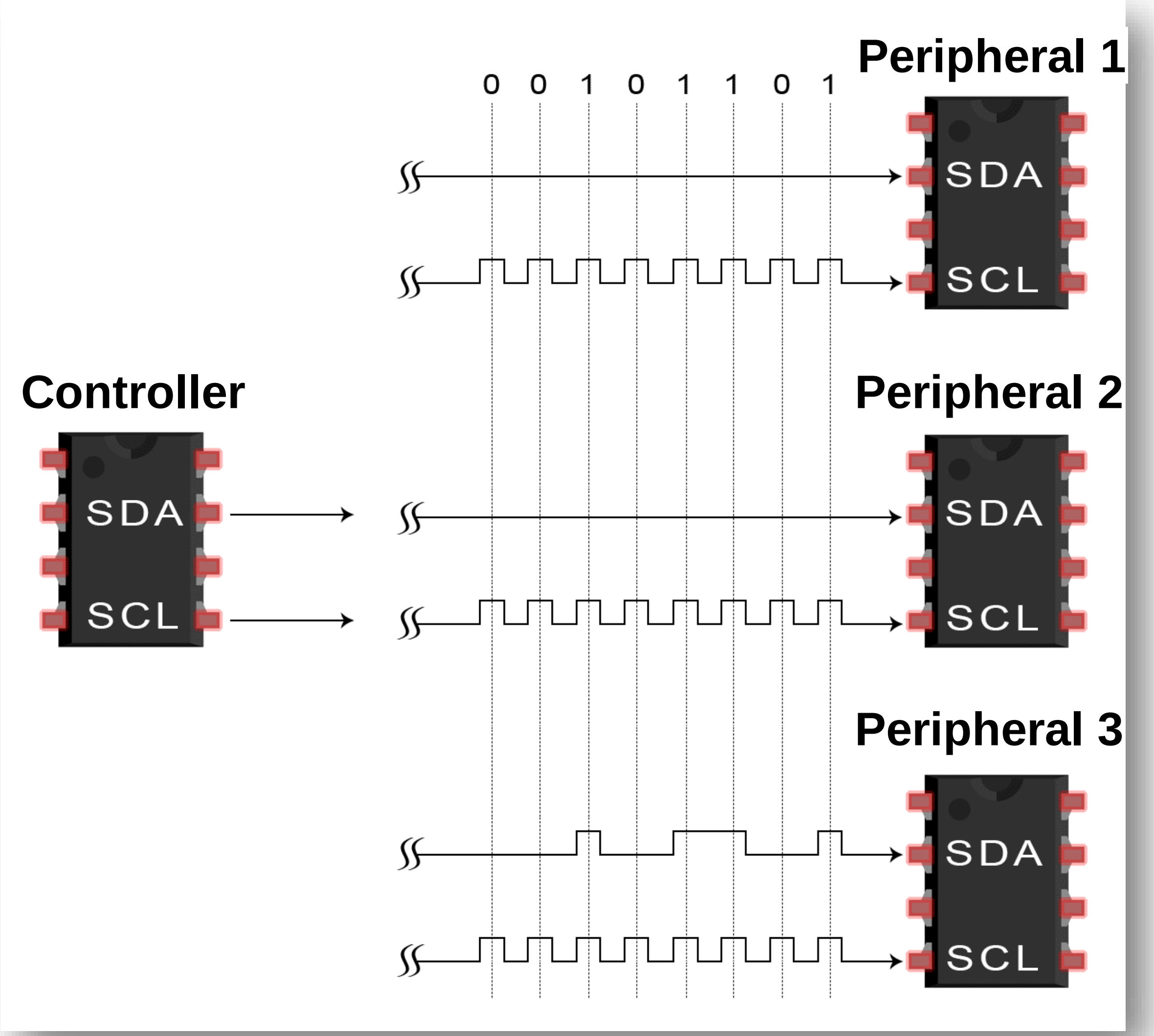
# I2C - Transmission

- Each peripheral compares the address sent from the controller to its own address. If the address matches, the peripheral returns an ACK bit by pulling the SDA line low for one bit. If the address from the controller does not match the peripheral's own address, the peripheral leaves the SDA line high



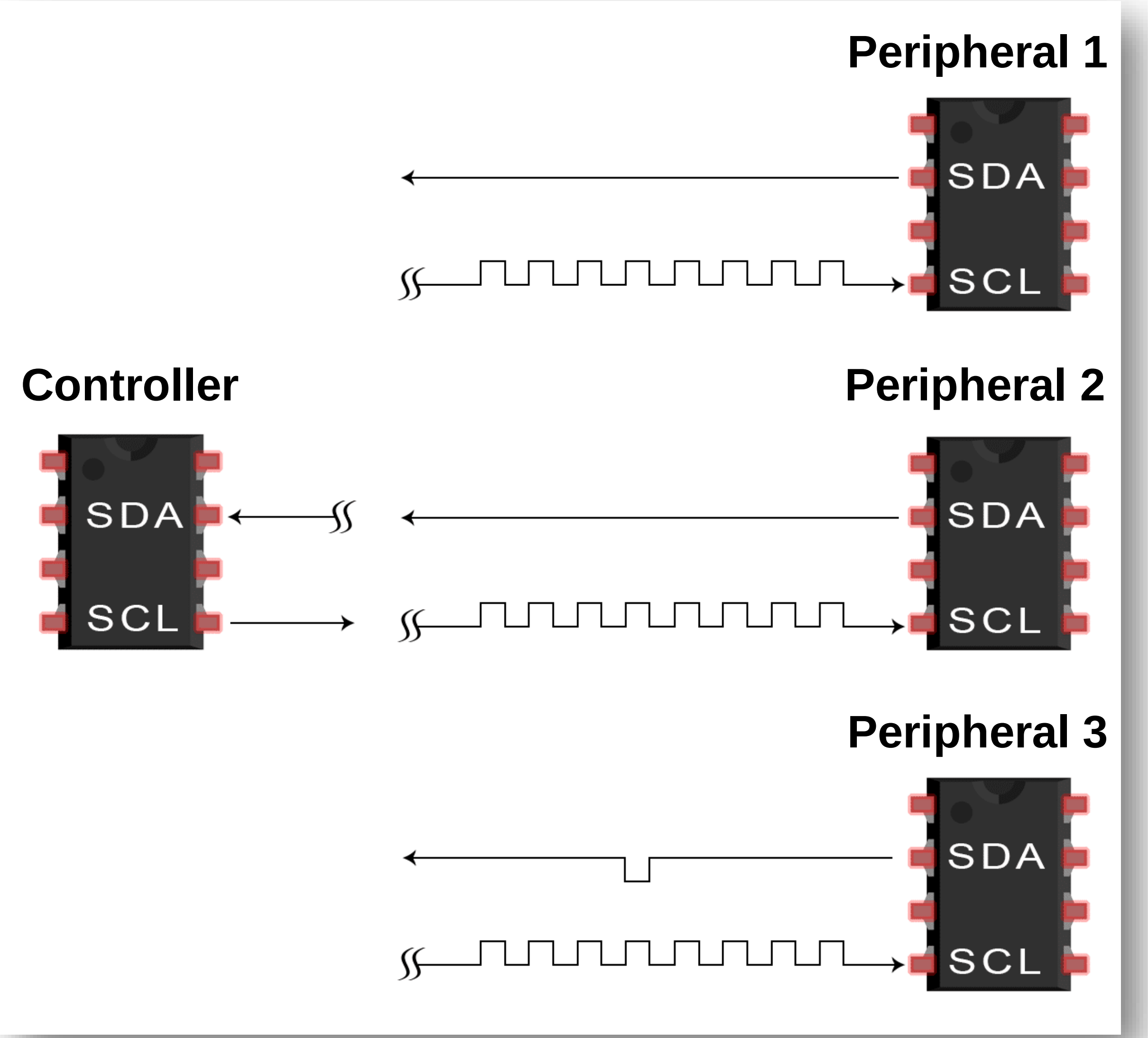
# I2C - Transmission

- ❑ The controller sends or receives the data frame



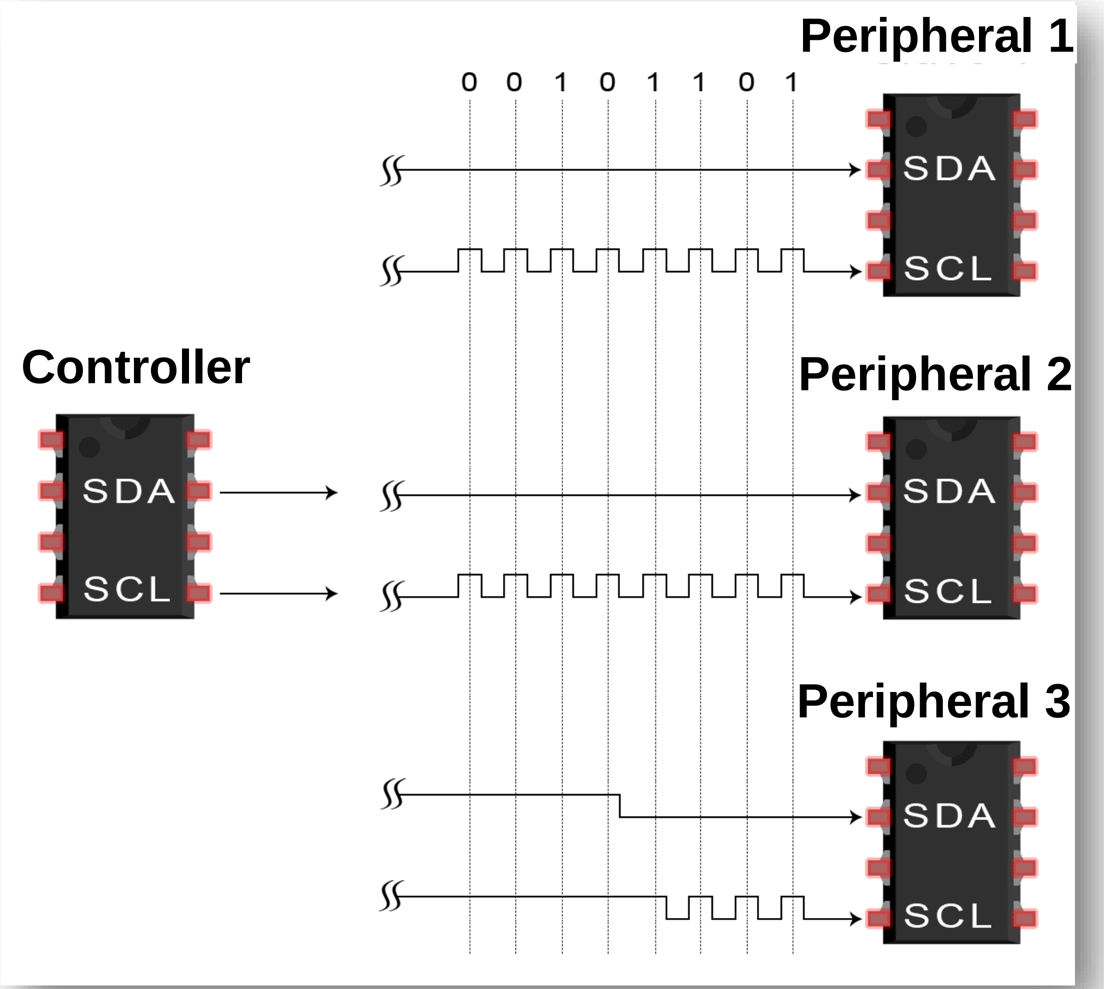
# I2C - Transmission

- After each data frame has been transferred, the receiving device returns another ACK bit to the sender to acknowledge successful receipt of the frame



# I2C - Transmission

- To stop the data transmission, the controller sends a stop condition to the peripheral by switching SCL high before switching SDA high



# I2C - Disadvantages

- Slower data transfer rate than SPI
- The size of the data frame is limited to 8 bits
- More complicated hardware needed to implement than SPI

HEIG-VD
HAUTE ÉCOLE D'INGÉNIERIE ET DE GESTION DU CANTON DE VAUD

Prof. Fouad HANNA, PhD.

fouad.hanna@heig-vd.ch

Hes·so
Haute Ecole Spécialisée de Suisse occidentale