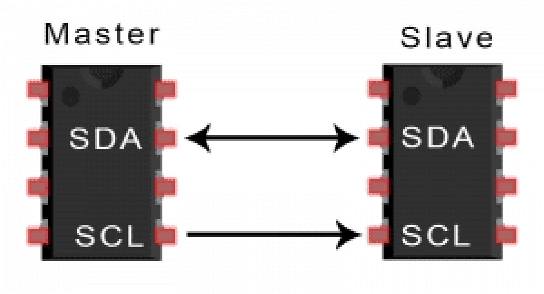
I2C PROTOCOL

The I2C stands for the inter integrated controller. The I2C protocol is a serial communication protocol that is used to connect low-speed devices. For example, **EEPROMs, microcontrollers, A/D and D/A converters,** and **input/output interfaces**. It was developed by **Philips semiconductor** in **1980** for inter-chip communication. It is a master-slave communication in which you can connect and control multiple slaves from a single master. In this, each slave device has a particular address. It supports various data rates according to versions ranging from 100 Kbps, 400 Kbps, 1 Mbps to 3.4 Mbps. It is synchronous communication like SPI.

I2C only uses two wires to transmit data between devices:



1. **SDA (Serial Data)** – The line for the master and slave to send and receive data.
2. **SCL (Serial Clock)** – The line that carries the clock signal.

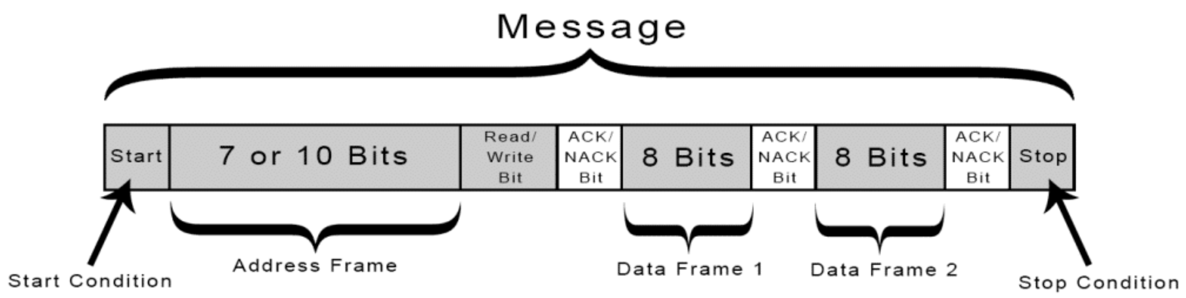
I2C is a serial communication protocol, so data is transferred bit by bit along a single wire (the SDA line).

Like SPI, I2C is synchronous, so the output of bits is synchronized to the sampling of bits by a clock signal shared between the master and the slave. The clock signal is always controlled by the master.

Wires Used	2
Maximum Speed	Standard mode= 100 kbps
	Fast mode= 400 kbps
	High speed mode= 3.4 Mbps
	Ultra fast mode= 5 Mbps
Synchronous or Asynchronous?	Synchronous
Serial or Parallel?	Serial
Max # of Masters	Unlimited
Max # of Slaves	1008

I2C WORKING:

With I2C, data is transferred in messages. Messages are broken up into frames of data. Each message has an address frame that contains the binary address of the slave, and one or more data frames that contain the data being transmitted. The message also includes start and stop conditions, read/write bits, and ACK/NACK bits between each data frame:



Start Condition: The SDA line switches from a high voltage level to a low voltage level *before* the SCL line switches from high to low.

Stop Condition: The SDA line switches from a low voltage level to a high voltage level *after* the SCL line switches from low to high.

Address Frame: A 7 or 10 bit sequence unique to each slave that identifies the slave when the master wants to talk to it.

Read/Write Bit: A single bit specifying whether the master is sending data to the slave (low voltage level) or requesting data from it (high voltage level).

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.

STEPS OF I2C DATA TRANSMISSION:

1. The master sends the start condition to every connected slave by switching the SDA line from a high voltage level to a low voltage level before switching the SCL line from high to low.
2. The master sends each slave the 7 or 10 bit address of the slave it wants to communicate with, along with the read/write bit.
3. Each slave compares the address sent from the master to its own address. If the address matches, the slave returns an ACK bit by pulling the SDA line low for one bit. If the address from the master does not match the slave's own address, the slave leaves the SDA line high.
4. The master sends or receives the data frame.
5. After each data frame has been transferred, the receiving device returns another ACK bit to the sender to acknowledge successful receipt of the frame.

ADVANTAGES:

- Only uses two wires.
- Supports multiple masters and multiple slaves.
- ACK/NACK bit gives confirmation that each frame is transferred successfully
- Hardware is less complicated than with UARTs
- Well known and widely used protocol.

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.