SNS COLLEGE OF TECHNOLOGY

Coimbatore-35

An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A++' Grade

Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

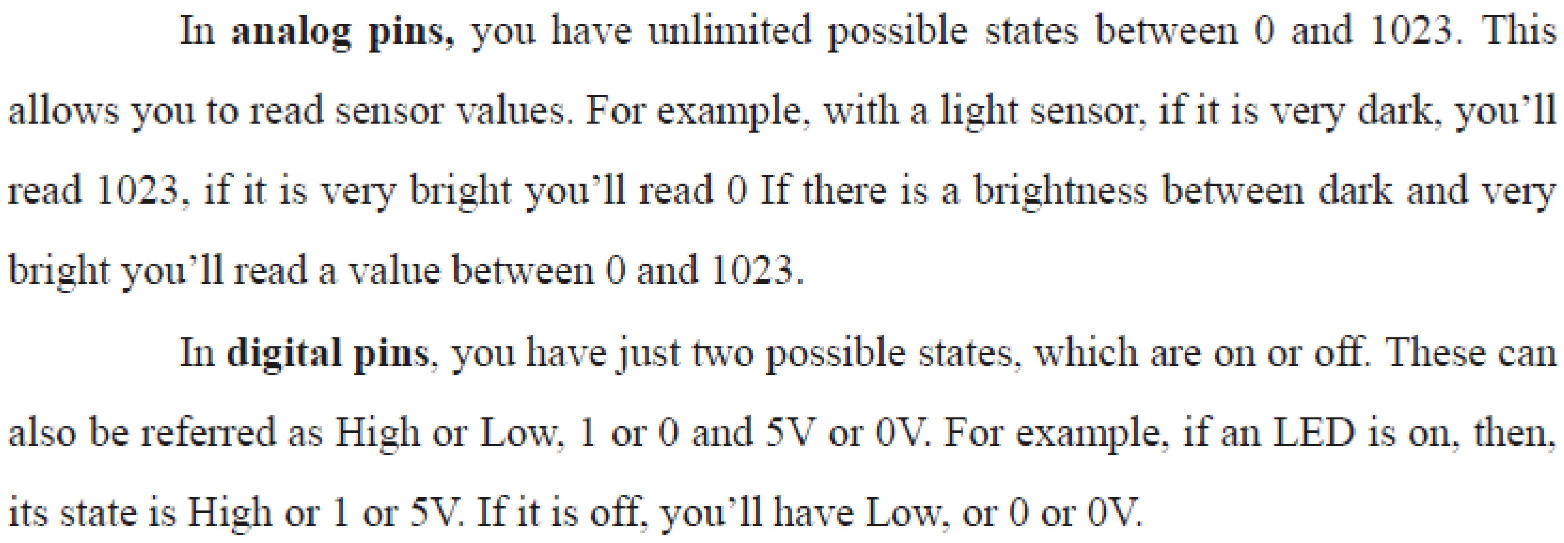
19ECT213- IoT SYSTEM ARCHITECTURE

II ECE / IV SEMESTER

UNIT 2 – MICROCONTROLLER AND INTERFACING TECHNIQUES FOR IoT DEVICES

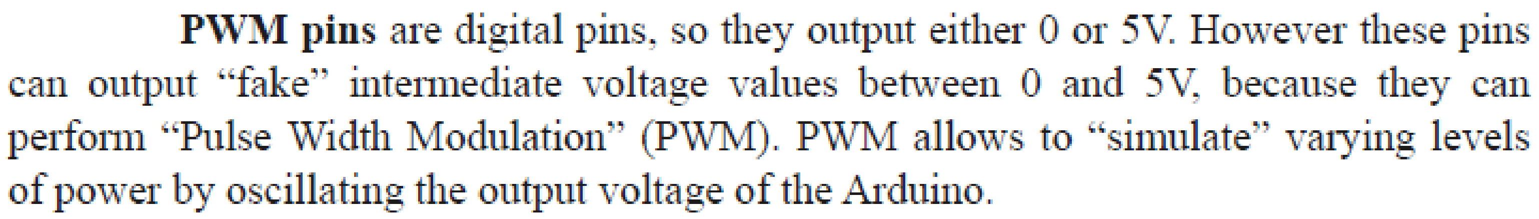
TOPIC 8 –-PWM Technique-Serial Communication

Difference between Analog, Digital and PWM Pins



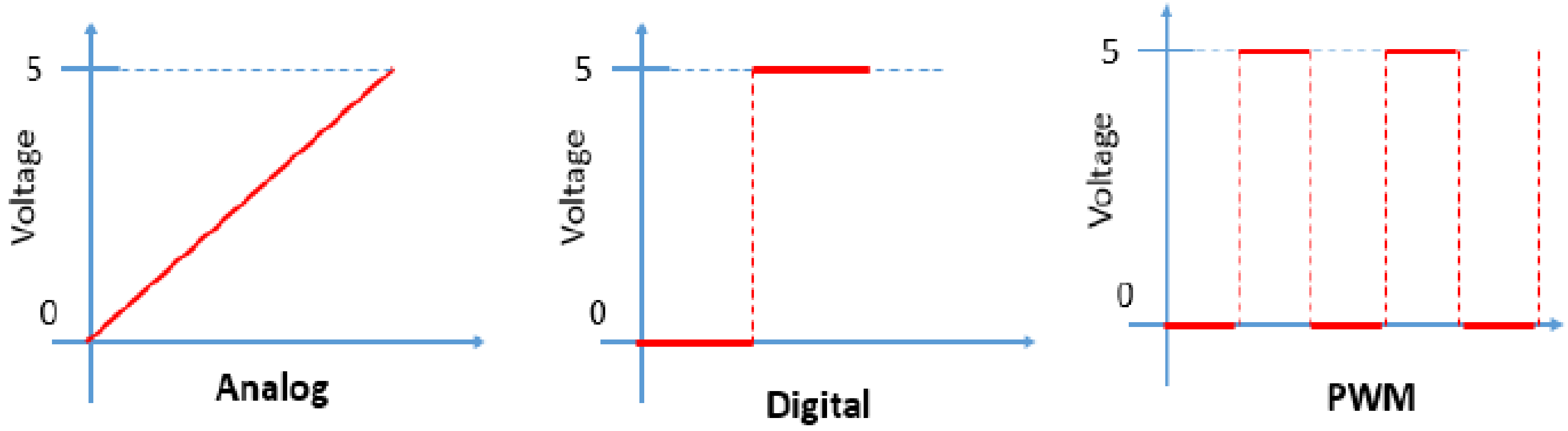
In **analog pins,** you have unlimited possible states between 0 and 1023. This allows you to read sensor values. For example, with a light sensor, if it is very dark, you'll read 1023, if it is very bright you'll read 0 If there is a brightness between dark and very bright you'll read a value between 0 and 1023.

In **digital pins**, you have just two possible states, which are on or off. These can also be referred as High or Low, 1 or 0 and 5V or 0V. For example, if an LED is on, then, its state is High or 1 or 5V. If it is off, you'll have Low, or 0 or 0V.



PWM pins are digital pins, so they output either 0 or 5V. However these pins can output "fake" intermediate voltage values between 0 and 5V, because they can perform "Pulse Width Modulation" (PWM). PWM allows to "simulate" varying levels of power by oscillating the output voltage of the Arduino.

Fundamentals of Arduino Programming

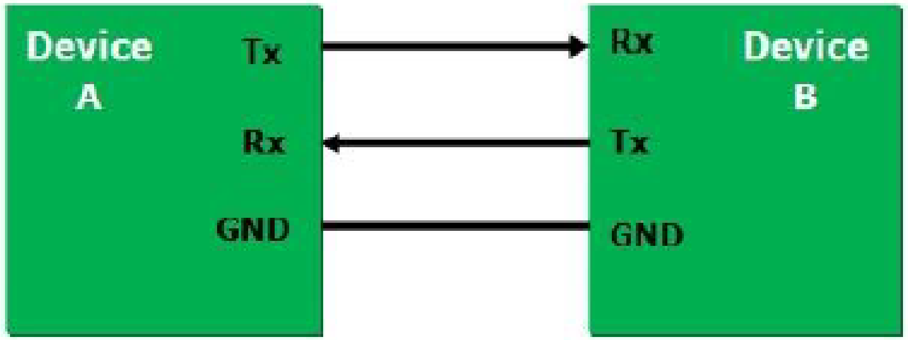


Difference between Analog, Digital and PWM Pins

Introduction to Communications

Serial (UART) communications:

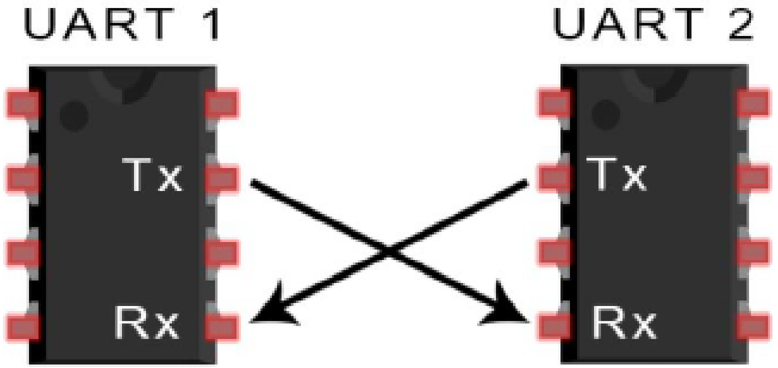
- Serial communication on Arduino pins Tx/Rx uses TTL logic levels which operates at either 5V/3.3V depending the type of the board used.
- Tx/Rx pins should not be connected to any source which operates more than 5V which can damage the Arduino board.
- Serial communication is basically used for communication between Arduino board and a computer or some other compatible devices.



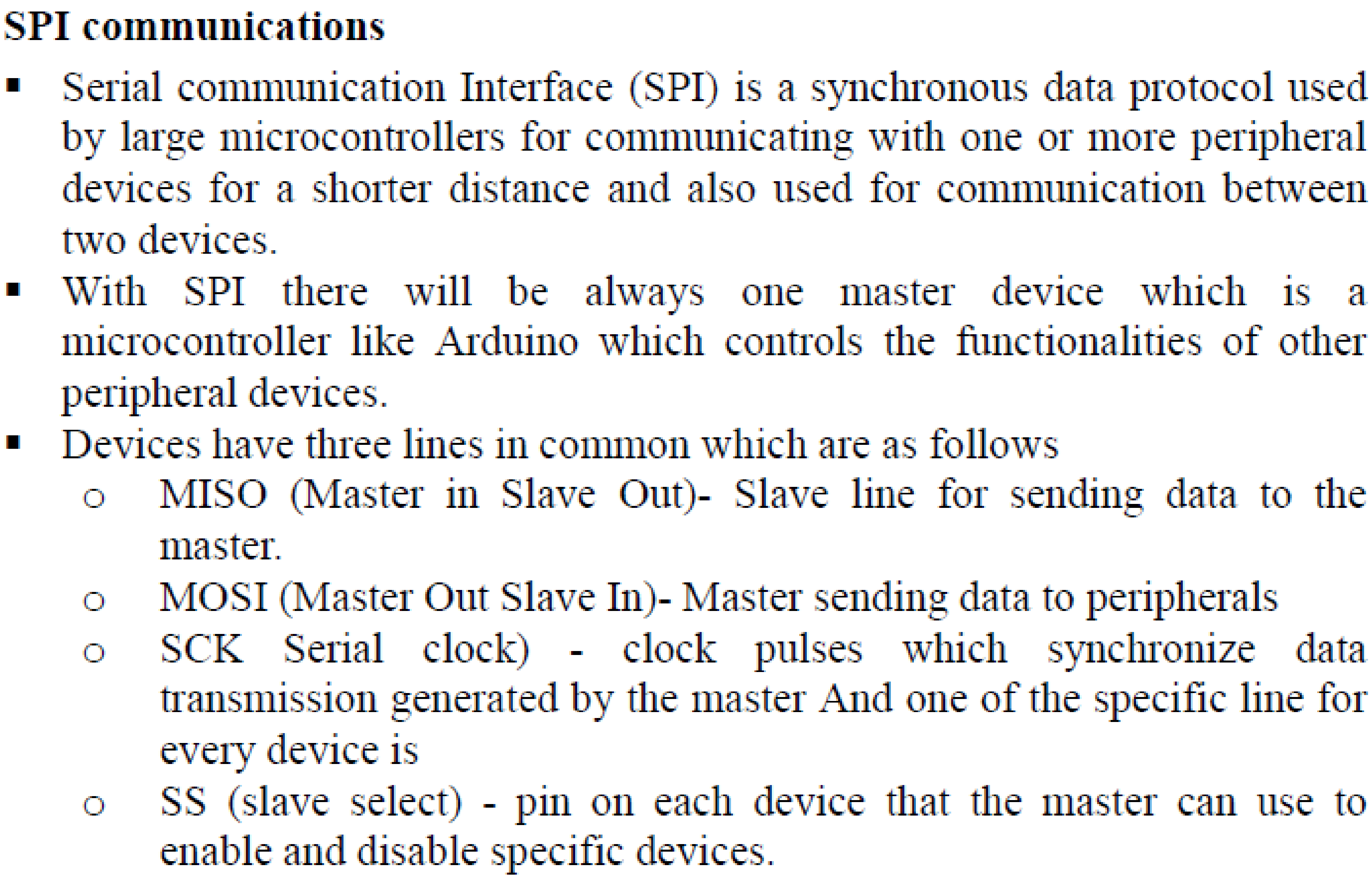
Introduction to Communications

Serial (UART) communications:

- Every Arduino board will have at least one serial port known as UART.
- Serial communicates on digital pins Rx(pin 0) and Tx(pin 1) with the computer via USB, pin 0 and pin 1 cannot be used for digital input or output.
- The built in serial monitor can be used to communicate with an Arduino board by selecting same baud rate that is used in the call to begin () which will come across in the later part of the chapter



Introduction to Communications



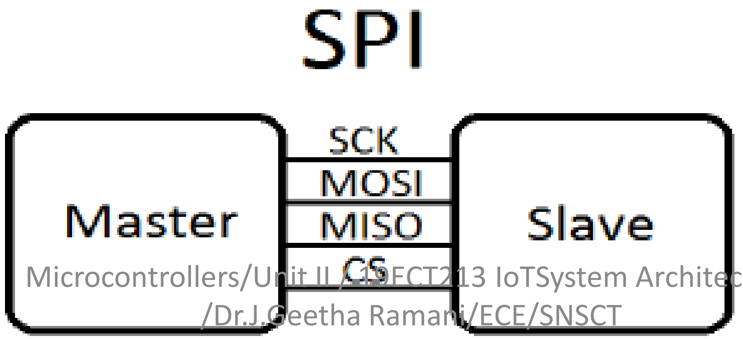
SPI communications

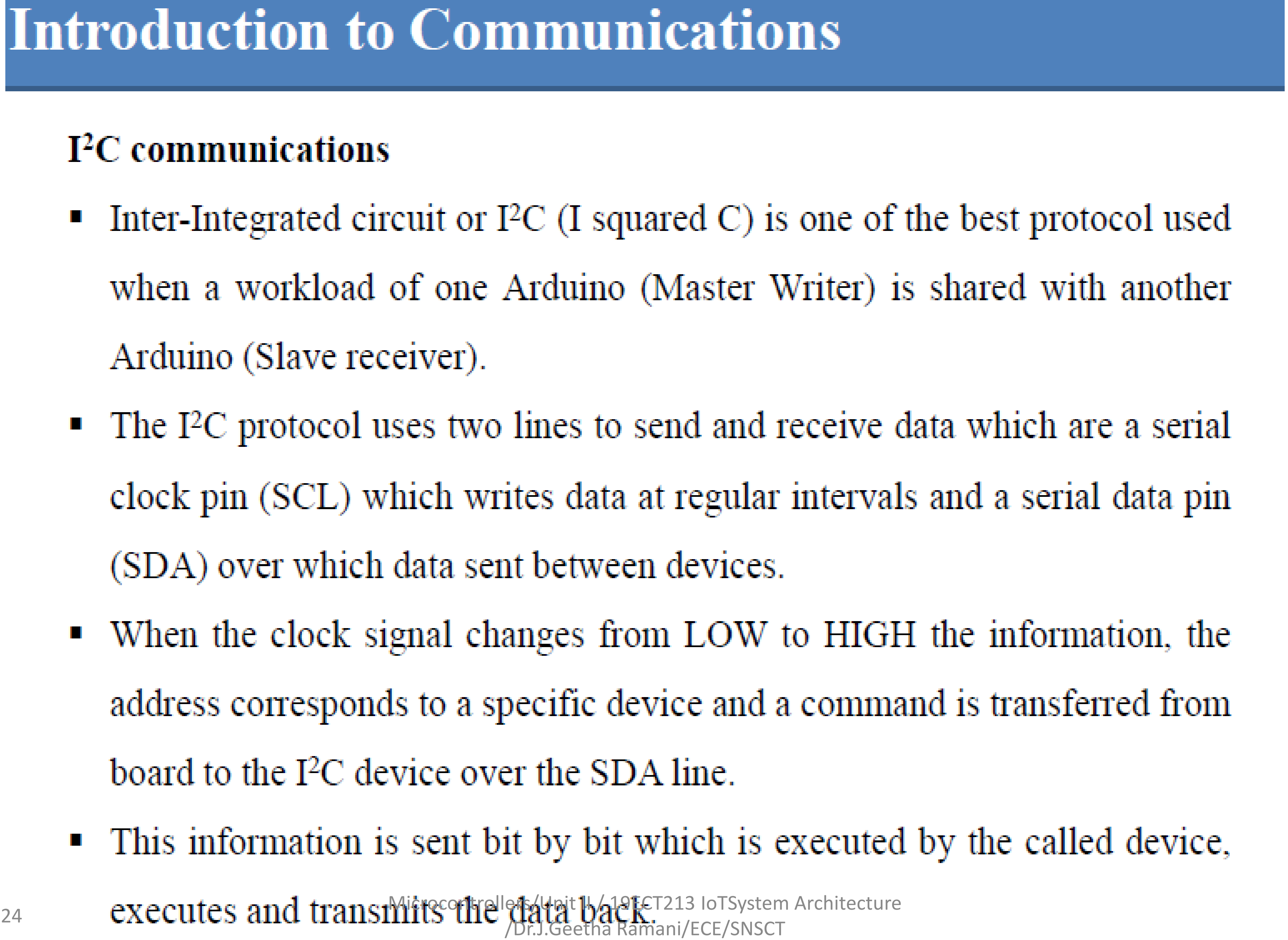
- Serial communication Interface (SPI) is a synchronous data protocol used by large microcontrollers for communicating with one or more peripheral devices for a shorter distance and also used for communication between two devices.
- With SPI there will be always one master device which is a microcontroller like Arduino which controls the functionalities of other peripheral devices.
- Devices have three lines in common which are as follows
  - MISO (Master in Slave Out)- Slave line for sending data to the master.
  - MOSI (Master Out Slave In)- Master sending data to peripherals
  - SCK Serial clock) - clock pulses which synchronize data transmission generated by the master And one of the specific line for every device is
  - SS (slave select) - pin on each device that the master can use to enable and disable specific devices.

Introduction to Communications

SPI communications

- When device SS pin is low, communication happens with the master, if SS pin is high device ignores the maser. This allows multiple SPI devices sharing the the same MISO, MOSI and CLK lines.
- To program a new SPI device some key points to be noted which are
  - Maximum SPI speed of the device used?
  - How data is shifted like MSB/LSB?
  - Data clock is idle when high/low.





Introduction to Communications

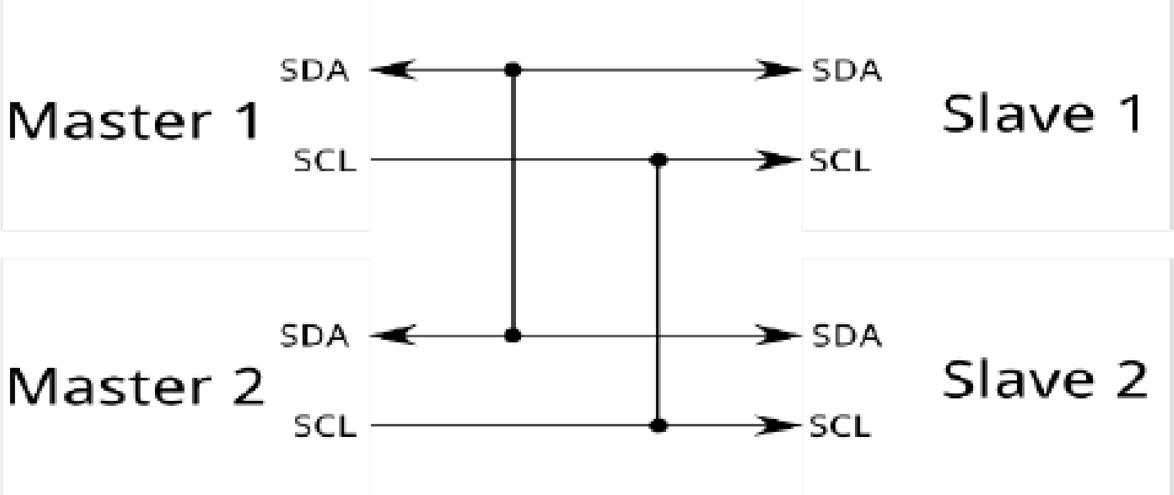
I^2C communications

- Inter-Integrated circuit or I^2C (I squared C) is one of the best protocol used when a workload of one Arduino (Master Writer) is shared with another Arduino (Slave receiver).
- The I^2C protocol uses two lines to send and receive data which are a serial clock pin (SCL) which writes data at regular intervals and a serial data pin (SDA) over which data sent between devices.
- When the clock signal changes from LOW to HIGH the information, the address corresponds to a specific device and a command is transferred from board to the I^2C device over the SDA line.
- This information is sent bit by bit which is executed by the called device, executes and transmits the data back.

Introduction to Communications

I^{2}C communications

- If the device require execution from another a slave device, the data is transferred to the board on the same line using pulse generated from Mater on SCL as timing.
- Each slave should have unique identity and both Master and slave turns out communicating on the same data line. In this way many of the Arduino boards are communicated using just two pins of microcontroller with each unique address of a device.


Master 1
SDA
SCL
Master 2
SDA
SCL
SDA
SCL
Slave 1
SDA
SCL
Slave 2