217529- Internet of Things

Unit Number: 5

Unit Name: **IoT Design & System Engineering**

Unit Outcomes: CO5

Design a Simple IOT system comprising sensors by analyzing the requirements of IoT application.

Syllabus

- Discuss IOT requirements
- Hardware & software
- Study of IOT sensors, Tagging and tracking
- Embedded products, IOT design
- SIM card technology, IOT connectivity and Managements
- IOT security & IOT communication.

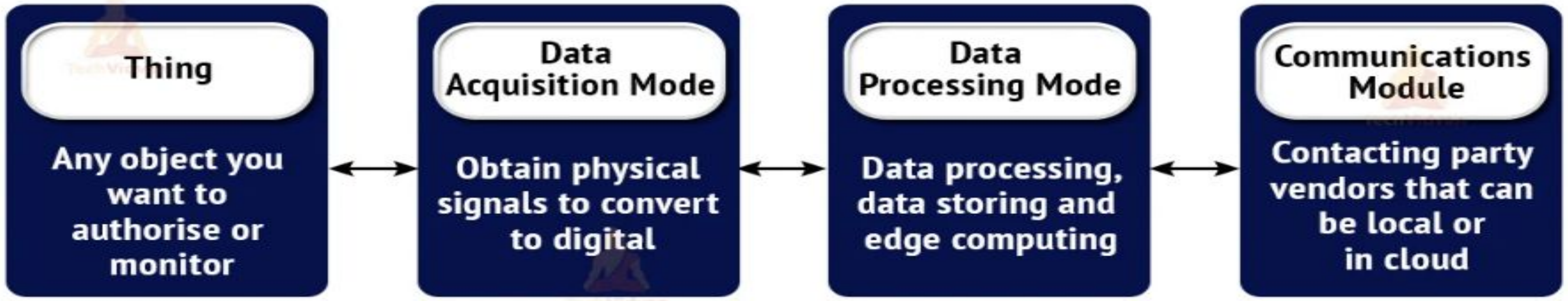
What is IoT and IoT Requirement?

- IoT is the extension of Internet connectivity into physical devices and everyday objects. Embedded with electronics, Internet connectivity, and other forms of hardware (such as sensors), these devices can communicate and interact with others over the Internet, and they can be remotely monitored and controlled. Let's break that down:
- First, IoT is about connectivity
- Second, IoT is about information and communication
- And finally, IoT is about action and interaction

- There are two types of IoT: CIoT and IIoT.
  - The difference is that **CIoT** often focuses on convenience for the "**Customer**", whereas **IIoT** is strongly focused on the "**Industry**" and is more system-centric.
  - Its focus is on improving the efficiency, security, and output of operations with a focus on Return on Investment (ROI).

IoT Hardware and Software

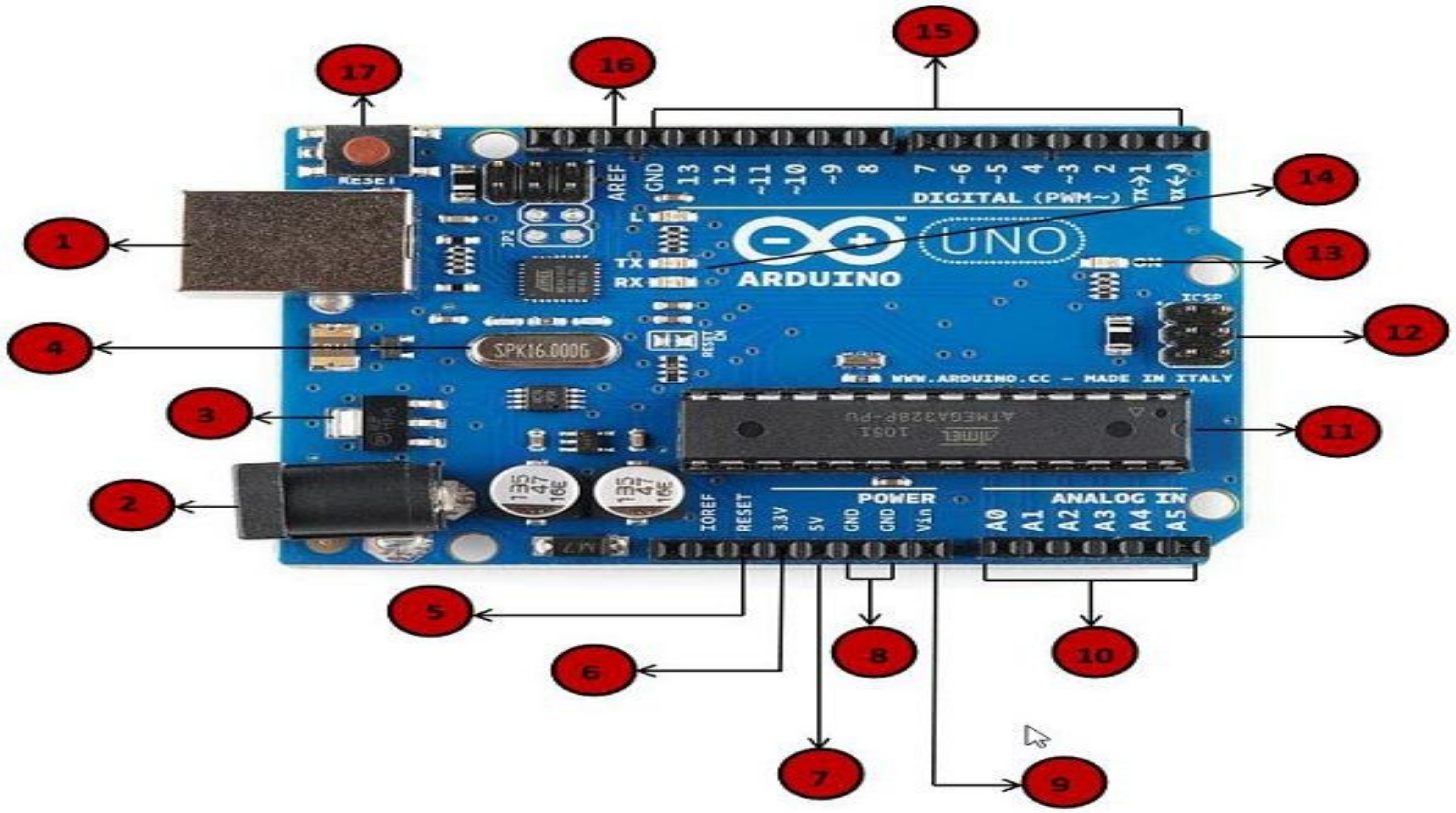
- Hardware and software devices combine to form an Iot ecosystem. Hardware is a **set of devices that wire together to serve some functionality**. A bread board is one such example.
- The bread board usually contains components such as **sensors, microcontrollers, microprocessors, resistors, transistors and voltage regulators.**
- Building blocks of IoT Hardware



- IoT Hardware includes a wide range of devices such as devices for routing, bridges, sensors etc. These IoT devices manage key tasks and functions such as system activation, security, action specifications, communication, and detection of support-specific goals and actions.

IoT Hardware components

- **Arduino:** Arduino/Genuino Uno is a **microcontroller board** based on the ATmega328P. It has **14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button.** It contains everything needed to support the microcontroller.



1	**Power USB** Arduino board can be powered by using the USB cable from your computer. All you need to do is connect the USB cable to the USB connection (1).
2	**Power (Barrel Jack)** Arduino boards can be powered directly from the AC mains power supply by connecting it to the Barrel Jack (2).
3	**Voltage Regulator** The function of the voltage regulator is to control the voltage given to the Arduino board and stabilize the DC voltages used by the processor and other elements.

15	**Digital I/O** The Arduino UNO board has 14 digital I/O pins (15) (of which 6 provide PWM (Pulse Width Modulation) output. These pins can be configured to work as input digital pins to read logic values (0 or 1) or as digital output pins to drive different modules like LEDs, relays, etc. The pins labeled "~" can be used to generate PWM.
16	**AREF** AREF stands for Analog Reference. It is sometimes, used to set an external reference voltage (between 0 and 5 Volts) as the upper limit for the analog input pins.

Cont..

a 4	**Crystal Oscillator** The crystal oscillator helps Arduino in dealing with time issues. How does Arduino calculate time? The answer is, by using the crystal oscillator. The number printed on top of the Arduino crystal is 16.000H9H. It tells us that the frequency is 16,000,000 Hertz or 16 MHz.
5,17	**Arduino Reset** You can reset your Arduino board, i.e., start your program from the beginning. You can reset the UNO board in two ways. First, by using the reset button (17) on the board. Second, you can connect an external reset button to the Arduino pin labelled RESET (5).
6,7 8,9	**Pins (3.3, 5, GND, Vin)** • 3.3V (6) − Supply 3.3 output volt • 5V (7) − Supply 5 output volt • Most of the components used with Arduino board works fine with 3.3 volt and 5 volt. • GND (8)(Ground) − There are several GND pins on the Arduino, any of which can be used to ground your circuit. • Vin (9) − This pin also can be used to power the Arduino board from an external power source, like AC mains power supply.

10	**Analog pins** The Arduino UNO board has six analog input pins A0 through A5. These pins can read the signal from an analog sensor like the humidity sensor or temperature sensor and convert it into a digital value that can be read by the microprocessor.
11	**Main microcontroller** Each Arduino board has its own microcontroller (11). You can assume it as the brain of your board. The main IC (integrated circuit) on the Arduino is slightly different from board to board. The microcontrollers are usually of the ATMEL Company. You must know what IC your board has before loading up a new program from the Arduino IDE. This information is available on the top of the IC. For more details about the IC construction and functions, you can refer to the data sheet.
12	**ICSP pin** Mostly, ICSP (12) is an AVR, a tiny programming header for the Arduino consisting of MOSI, MISO, SCK, RESET, VCC, and GND. It is often referred to as an SPI (Serial Peripheral Interface), which could be considered as an "expansion" of the output. Actually, you are slaving the output device to the master of the SPI bus.

RaspberryPi:

- Raspberry Pi, developed by Raspberry Pi Foundation in association with Broadcom, is a series of **small single-board computers,microprocessor** and perhaps the most inspiring computer available today.
- Another well-known IoT platform is Raspberry Pi 2/3/4, which is a **very affordable and tiny computer that can incorporate an entire web server.** Often called "RasPi," it has enough **processing power and memory to run Windows 10 on it as well as IoT Core**.
- RasPi exhibits great processing capabilities, especially when using the **Python programming language**.
- Generations and Models
  - In 2012, the company launched the Raspberry Pi and the current generations of regular Raspberry Pi boards are Zero, 1, 2, 3, and 4.
  - Generation 1 Raspberry Pi had the following four options – Model A
  - Model A +
  - Model B
  - Model B +
  - Among these models, the **Raspberry Pi B models are the original credit-card sized format**.

RaspberryPi:

- The Raspberry Pi Model B features:
  More GPIO
  More USB
  Micro SD
  Lower power consumption
  Better audio

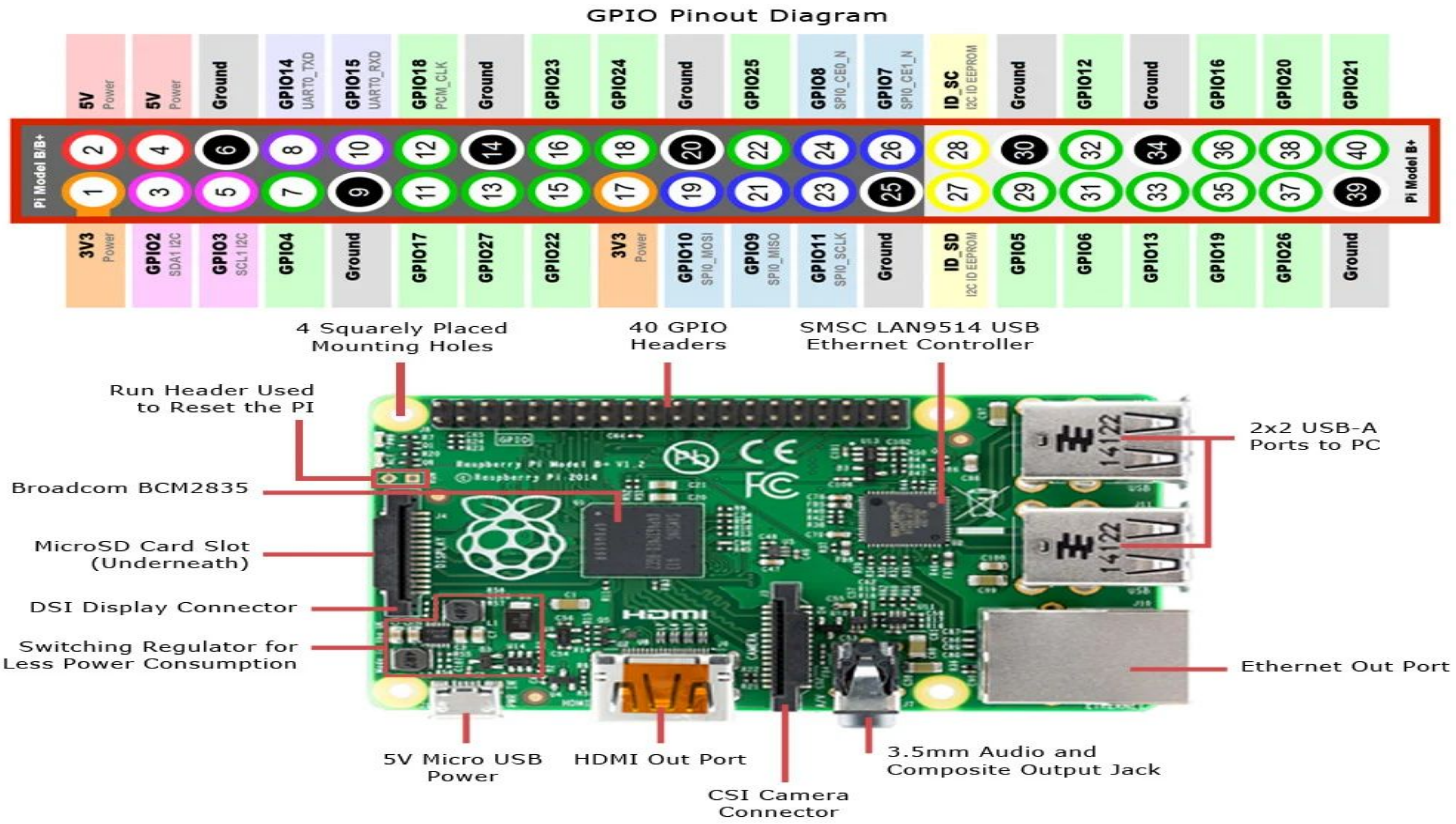
- There are four USB 2.0 ports compared to two on the Model B.

- The SD card slot has been replaced with a more modern type micro SD slot.

- It consumes slightly less power, provides better audio quality

- 5V USB power supply of at least 2 amps with a micro USB cable, any standard USB keyboard and mouse, an HDMI cable and monitor/TV for display, and a micro SD card with the operating system pre-installed.

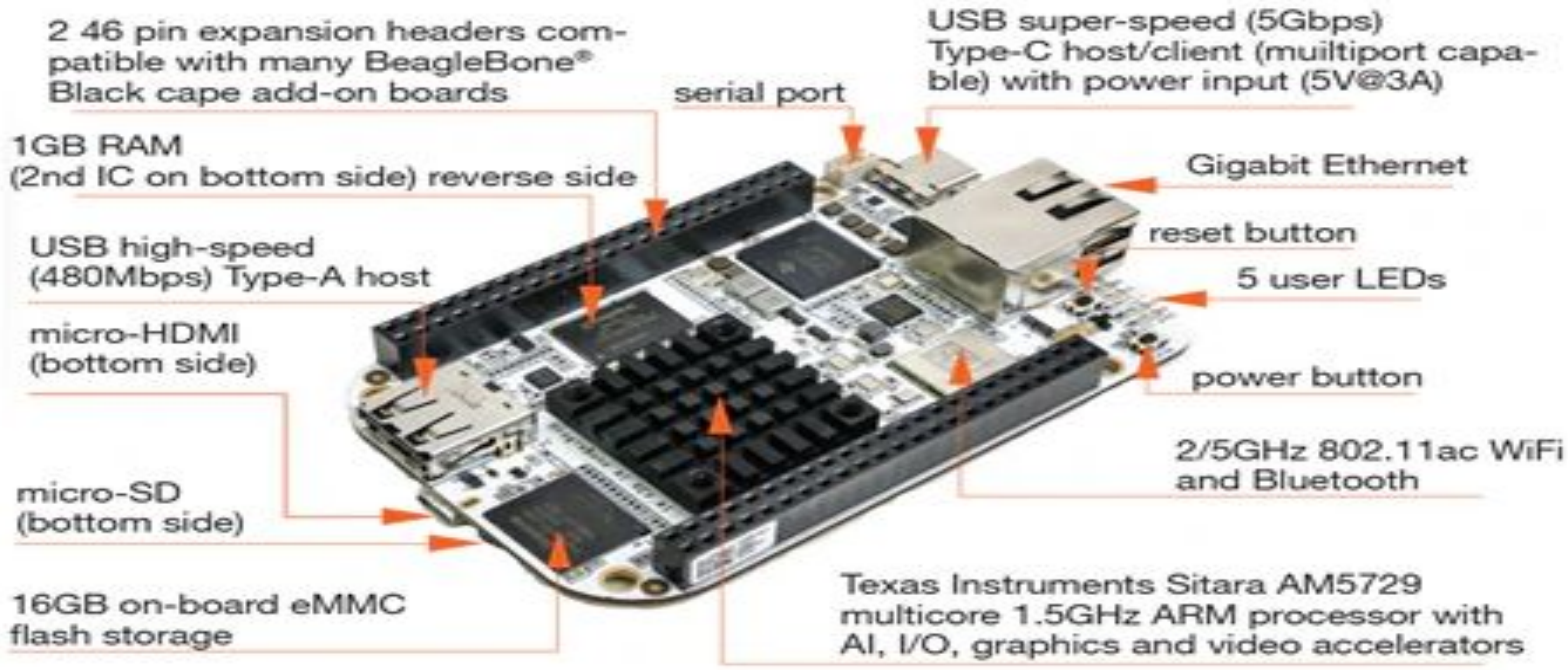
Cont..


GPIO Pinout Diagram
5V Power
5V Power
Ground
GPIO14 UART0_TXD
GPIO15 UART0_RXD
GPIO18 PCM_CLK
Ground
GPIO23
GPIO24
Ground
GPIO25
GPIO8 SPI0_CE0_N
GPIO7 SPI0_CE1_N
ID_SC I2C ID EEPROM
Ground
GPIO12
Ground
GPIO16
GPIO20
GPIO21
Pi Model B/B+
Pi Model B+
3V3 Power
GPIO2 SDA1 I2C
GPIO3 SCL1 I2C
GPIO4
Ground
GPIO17
GPIO27
GPIO22
3V3 Power
GPIO10 SPI0_MOSI
GPIO9 SPI0_MISO
GPIO11 SPI0_SCLK
Ground
ID_SD I2C ID EEPROM
GPIO5
GPIO6
GPIO13
GPIO19
GPIO26
Ground
4 Squarely Placed Mounting Holes
40 GPIO Headers
SMSC LAN9514 USB Ethernet Controller
Run Header Used to Reset the PI
2x2 USB-A Ports to PC
Broadcom BCM2835
MicroSD Card Slot (Underneath)
DSI Display Connector
Switching Regulator for Less Power Consumption
Ethernet Out Port
5V Micro USB Power
HDMI Out Port
CSI Camera Connector
3.5mm Audio and Composite Output Jack

Components	Description
Processor	Raspberry Pi uses an ARM processor.It is brain of Raspberry pi
Memory	Model B has 512 MB SDRAM, It store programs that are currently being run in the CPU
USB Ports	Board has 2 X 2 USB ports so that connect more devices
HDMI output	High definition Multimedia Interface(HDMI) supports high quality digital video and audio through singal cable
Composite Video output	It supports composite video output with RCA Jack
Audio Output	Audio output jack is 3.5 mm used for providing audio output to old television along with RCA jack for video
GPIO pins	26 GPIO pins, GPIO reading various environmental sensors, writing output to DC motors, LEDs for status
Power input	Micro USB connector is used for power input
Status LED	It has five status LED
CSI	Camera serial Interface(CSI) can be used to connect a camera module to Raspberry pi
SD card slot	This card is used for loading operating system

BeagleBoard

- BeagleBoard is a **single-board computer** with a **Linux-based OS that uses an ARM processor**, capable of more powerful processing than RasPi.
- Tech giant Intel's Galileo and Edison boards are other options, both great for larger scale production, and Qualcomm has manufactured an array of enterprise-level IoT technology for cars and cameras to healthcare.



IoT Software

C & C++: Widely used for hardware systems due to their roots in embedded systems, C++ extends C with object-oriented features, popular in Linux and Arduino.

Java: Portable language allowing code to run on different systems with ease, thanks to its write once, run anywhere approach.

Python: Highly popular for web development and IoT, especially on Raspberry Pi, known for its readability and data handling capabilities.

B#: Specifically designed for embedded systems, compact and memory-efficient.

Data Collection: Filters, secures, and measures data, aiding decision-making, and centralizing transmission to a central server.

Device Integration: Ensures seamless connectivity among devices, facilitating information sharing.

Real-Time Analytics: Analyzes user input for immediate insights, enabling automation and productivity improvements.

Application and Process Extension: Enhances existing systems by integrating new devices, boosting productivity, and ensuring accurate data collection.

IoT hardware requirements for deploying your IoT project

- **Security requirements**: Security is an essential component of the Internet of Things. Considering the device's security requirements is imperative at all the development and designing stages. Even when prototyping, make sure the security and integrity of data captured by any device remain intact. All IoT devices, their network, service applications of websites, and mobiles apply the security requirements.
- **Ease of development**:Ease of development is a requirement of high priority when prototyping. It enables the user to get the IoT device up and running quickly and efficiently when capturing data and interconnecting with other devices and the cloud. When deploying your IoT projects, have in mind the API documentation's quality, accessibility, and availability. Also, consider the tools of development, and support provided by the device's manufacturer or by the development team.
- **Data acquisition, processing, and storage requirements:**The number of sensors connected to the captured data's resolution and the sampling rate are the main determinants of the data's volume to be processed. They also influence the requirements of storing and processing data.
- **Connectivity requirements:** Wireless networking has connectivity requirements such as operating range, distance covered by the transmitting signal, and the predicted data and transmitted volume. When checking on the device's connectivity requirements, it is vital to contemplate the fault tolerance, the device's reconnecting capability, and how long a device takes when retrying to send data after it disconnects.

Cont....

- **Power requirements:** The power requirements are impacted mainly by the network transmission rate and the number of sensors in the device. Therefore, when deploying your IoT project, it is essential to consider if the device needs a mobile power source such as a super capacitor or a battery or hardwired for power. Also, know the battery's size, capacity requirements, weight, and if the battery is recharged, replaced, or discarded when it dies. In case the battery is rechargeable, check by what means and how often it is charged?

- **Physical device design requirements:** They include the size and physical appearance of the hardware device. When designing an IoT device, it is essential to consider the ecological situations in which the device will be installed. For instance, consider if the device will require a ruggedized(hard wearing or shock resistant) or a waterproof? All appliances installed on a truck's underside as part of a fleet monitoring application should always be safeguarded to ensure that it works well, even when under harsh conditions. The device has to be water-resistant and impervious to shock, dirt, and vibration.

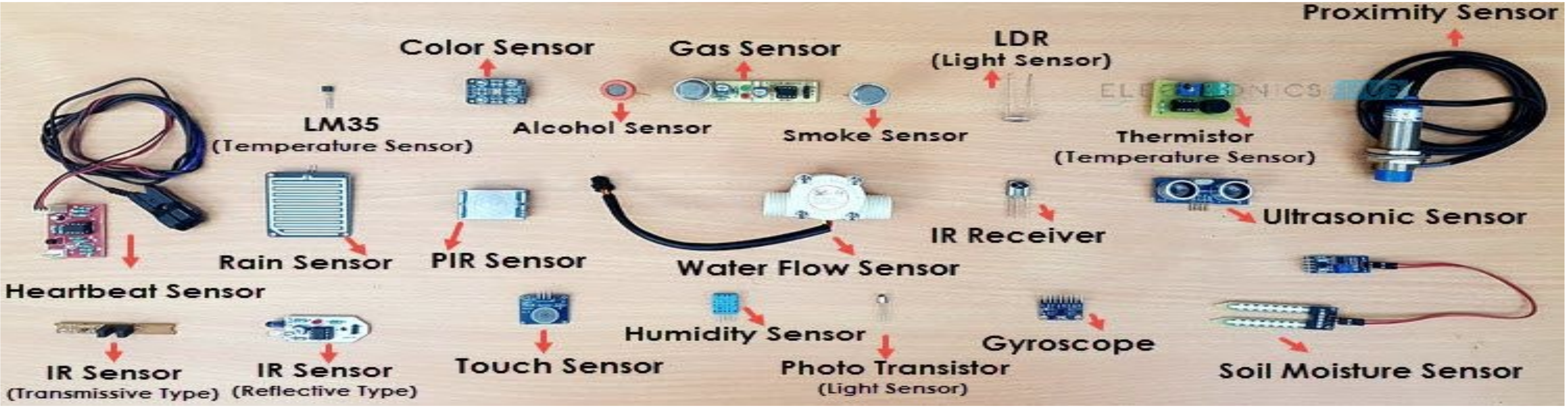
Cont....

- **Cost requirements:** The original hardware's outlay and allied components like sensors are the main determinants of the hardware's price. Other components that determine the hardware cost include the ongoing operating cost like the maintenance and power cost. Also, it is essential to think through the reasonable licensing fees for some device's drives and components. Assembling custom boards is more expensive than purchasing commercially accessible off-the-shelf development boards. It is a wiser alternative to consecrate hardware devices when scaling out in the IoT network with numerous instruments.

- **Processors:** Data is processed once the sensor data capture it before conveying the outcomes to the cloud. Thus, the quantity of data processing needed to create the subsequent sensor data and the complexity of sensors determines the processing level. For example, the temperature reading is a simple illustration of an average of set values or a single data value over time. Moreover, a security camera unable to record digital video without the scene detection algorithm flagging an event can be more complex.

Sensors in Internet of Things(IoT)

- A Sensor is an electronic device that is used to measure some sort of physical parameters (e.g. temperature, pressure, light intensity, etc). The output of an electronic sensor is an electrical signal that is either analog or digital. Processing the sensor's output can be done in hardware (using discrete electronic elements) or in software (using some sort of microcontrollers or MPUs).



- Sensors Classification: . We can classify sensors depending on the type of output signal or the physical parameters they measure and other considerations could be taken resulting in a variety of ways to classify sensors. First of which is the type of output signal and the second one is the physical parameter they measure.

Cont..

Analog Output	Digital Output
The output of these sensors is an analog voltage that you can measure then determine the desired physical parameter using the sensor's transfer function. It may also be capacitive or resistive or anything analog.	The output of these sensors is digital data that you can read via serial or parallel communication buses (as UART, SPI, I2C, etc). The typical format for the data is demonstrated exactly in the sensor's datasheet.
Example: The temperature sensor (specifically LM35) is an analog sensor whose output	Example: The accelerometer sensor (ADXL345) is a digital sensor that sends out its output digital data

- Sensor's Physical Parameter Classification
- There are sensors to measure everything you can possibly think of and here is a table for the most common ones.

Temperature Sensors	Chemical Sensors	Proximity Sensors	Touch Sensors
Light Sensors	Tilt Sensor	Metal Detectors	Cameras
Humidity Sensor	Vibration Sensor	Magnetic Sensor	Color Sensor
Current Sensor	Pressure Sensor	Fingerprint Sensor	GPS
Motor Speed Sensor	Bending Sensor	PIR Sensor	Position Sensor
Lidar Sensor	Ultrasonic Sensor	Gyroscope Sensor	Accelerometer Sensor
Digital Compass Sensor	Sound Sensor (Mic.)	IR Sensor	Odometer Sensor

Applications For Sensors

- Sensors have been around since the early days of electricity and have been in use in a very wide range of applications. We use sensors in electronics projects, robotics, industry, and much more. Down below is a brief list of typical applications of sensors.
- Robotics
- Embedded Systems-home appliances,office automation, security.
- Computers
- Smart Cars
- Avionics-development and production of electronic instrument for use in aircraft and astronauts
- Satellites
- Smart Homes
- Smartphones
- Smart Watches
- Energy plants
- Remote Sensing
- Communications
- etc.

Tagging and Tracking

- Asset tracking
- People Tracking
- Animal Tracking
- Inventory Management
- Supply chain visibility
- Geofencing

IoT tracking and tagging technologies:

- RFID tags
- Bluetooth Low Energy (BLE) beacon
- NFC(Near Field Communication)
- Zigbee
- LTE (Long-Term Evolution) Advanced
- LiFi(Light fidelity)
- GPS(Global Positioning)
- LPWAN(Low Power Wireless Wide Area Network)

Embedded Products

- **smart home devices**- smart lighting, smart locks, smart cameras, smart speakers, smart thermostats, smart TV, smart AC, smart plugs

- **Industrial Automation system-** Industrial robots, industrial control system. asset tracking system, industrial sensors

- **Wearables-** Fitness trackers, smart watches, smart glasses, smart clothing, smart jewellery

- **Connected Vehicles-** Remote vehicle monitoring, Autonomous vehicles,Telematics system(track location, speed, and performance of a vehicle)

- **Smart Cities-** Smart lighting system, smart waste management, smart public transportation systems, smart traffic management system, smart building system

Introduction to IoT Design

- IoT design is the process of creating connected devices that can exchange data with other devices or systems over the internet.
- The design process involves several key stages including:
  1. **Requirement gathering-** This involves understanding the specific needs & goals of the IOT solution including the types of the data that need to be collected and how that data will be used.

  1. **Hardware design-** This involves selecting the appropriate components and designing the hardware architecture for the device including sensors, wireless connectivity and power management.

  1. **Software development-** This involves software applications taht will run on the device and enable it to collect, process and transmit data.

  1. **System integration-** This involves integrating the hardware and software componmets of the device and testing the device to ensure that it meets the functional and performance requirements.

Introduction to IoT Design..

5. **Cloud integration-** This involves connecting the device to the cloud and configuring the cloud services that will be used to store and analyse the data collected by the device.

6. **Security and privacy-** This involves implementing security measures to protect the device and the data ie collects from unauthorized access and ensuring that the device complies with data privacy regulations.

SIM card Technology

- IoT SIM card technology is a specialized type of SIM card that is designed specifically for IoT devices.IoT SIM cards are used to provide cellular connectivity to IoT devices, allowing them to connect to the internet and exchange data with other devices or system.
- **M2M communication**- IoT SIM cards are designed for machine to machine(M2M) communication which means they are optimized for low power, low bandwidth connections that can operate for long periods of time.
  - Network coverage
  - data plans
  - security
  - Integration with cloud platform
  - remote management
  - support and customer survive
- **Global coverage-**IOT SIM cards can provide global coverage, enabling IoT devices to operate in different countries and regions.
- **Data Plans-** IoT SIM cards typically offer data plans that are designed specifically for IoT devices with flexible data usage and pricing options.
  - Pay As You Go plans
  - Fixed data plans

SIM card Technology..

- Unlimited data plans
- Tiered data plans

- **Roaming**- IOT SIM cards can support roaming which means that IOT devices can continue to operate even when they are outside of their home network coverage area.

- **Security**- IOT SIM cards can include security features such as embedded SIM(eSIM) technology, which provides a secure way to store and manage SIM card information as well as encryption and authentication protocols to protect data.
  - Secure communication protocols
  - Authentication and encryption
  - Secure storage
  - Two factor authentication
  - physical security
  - compliance

SIM card Technology..

- **Management Tools-** IOT SIM cards often come with management tools that enable users to monitor and control the data usage of their IOT devices as well as provision and deactivate SIM cards remotely.
  - Real Time monitoring
  - Analytics and reporting
  - Security Features
  - Integration with cloud platforms
  - Alerts and Notifications
  - API access

IOT connectivity and Management

- IOT devices typically require some form of connectivity to exchange data with other devices or systems and effective management is necessary to ensure that devices are functioning properly and that data is being collected and analysed efficiently.
- Here some key aspects:
  - Connectivity options-
    - Cellular devices
    - WiFi
    - Bluetooth
    - ZigBee
    - LoRaWAN
    - Satellite
  - Device Management
    - Remote device monitoring
    - Device configuration
    - Security Management
    - Analytics and reporting
    - Integration with cloud platform

IOT connectivity and Management

- Data Management-
    - data collection
    - Data storage
    - Data processing
    - Data Analysis
    - Data visualization
- Security
    - Authentication and access control
    - Encryption
    - Secure communication protocols
    - Physical security
    - Network Security
    - Privacy
- Cloud connectivity
- Analytics and insights

IOT Security & IOT Communication

-

SPPU Question

Q5) a) Explain IoT Information model specification. [9]

b) Explain Various IoT sim card Technologies [9]

OR

Q6) a) What are the criterias for selection of controllers in Embedded Products? [9]

b) What are different security parameters considered while designing any IoT system? [9]