

* UNIT - 3 *

* Introduction to python programming:

★ Just learn about Conditional statements, loops, jumping stmts, functions, modules.

* Control Flow: we can control the flow execution of the program using these statements.

1. Loops
2. conditional Stmt's
3. jumping stmts.

★ Loops: If you want to execute the same block of code for multiple times use these statements.

1. for loop
2. while loop.

* Syntax:

for i in iterable:

 _____ } block of
 _____ code

Syntax:

initialization
while condition:

 _____ }
 block
 at
 while
 increment /
 decrement

* Conditional statements:

- if
- If-else
- else-if ladder
- nested if

* Jumping:

- Break
- continue
- pass

* Introduction to Raspberry Pi:

Raspberry Pi is a small Single board Computer. By connecting peripherals like mouse, keyboard, monitor.

It will act as mini personal Computer.

It is used for real time image/video processing, IOT based applications and robotic applications.

It is also like a Computer but it is slower than computer because it has low configurations, and it is lower power consumption.

It can store the data and process the data, and transfer the data.

* OS for Raspberry Pi

Raspberry Pi foundation officially provides Debian based Raspbian OS. and also they provide Noobs OS for it.

* we can install different OS also in Raspberry Pi like Windows 10 IoT Core, Ubuntu, RISC OS..etc.

* Raspbian OS officially free to use.

* key features

1. Low Cost: Raspbian boards are affordable option for students, teachers, developers. because it available at low cost. (priced range between \$35 to \$55).

2. High performance: Raspberry Pi feature quad-core CPUs, upto 4GB of RAM, and built in WiFi and bluetooth connectivity.

3. Open-Sources: Raspberry Pi operates on open source SW, including the Raspbian OS which is based on Linux.

4. GPIO Pins

Raspberry Pi board features 40 GPIO (General purpose input /output) pins. to connect multiple devices. Such as sensors, motors, LED's.

* Types of Raspberry Pi Models:

① Raspberry Pi 4 - the latest model

featuring a quad-core Cortex-A72 CPU, up to 4GB of RAM, and improved performance.

② Raspberry Pi 3

* A popular model with quad-core Cortex-A53 CPU.

* 1GB of RAM

③ Raspberry Pi 2

* Older model with a quad-core Cortex-A7 model, 1GB of RAM

* Improved performance over original

4. Raspberry Pi Zero:- A small (tiny)

low - cost model with a single core CPU.

* 512 MB of RAM

* limited GPIO Pins.

* Applications:-

1. Learning to Code:- Raspberry Pi is an excellent platform for learning programming languages like Python, Java, etc.

2. Home Automation Use it to control and automate home appliances, lighting, and security systems.

3. Robotics and IoT:- Raspberry Pi is a popular choice for building blocks of robots, IoT projects.

* Functional Architecture of Raspberry Pi

→ First we need to know about the components of raspberry Pi, Mentioned below:

1. CPU (Central processing unit)
2. GPU (Graphics processing unit)
3. RAM (Random Access Memory)
4. Storage (micro-SD)

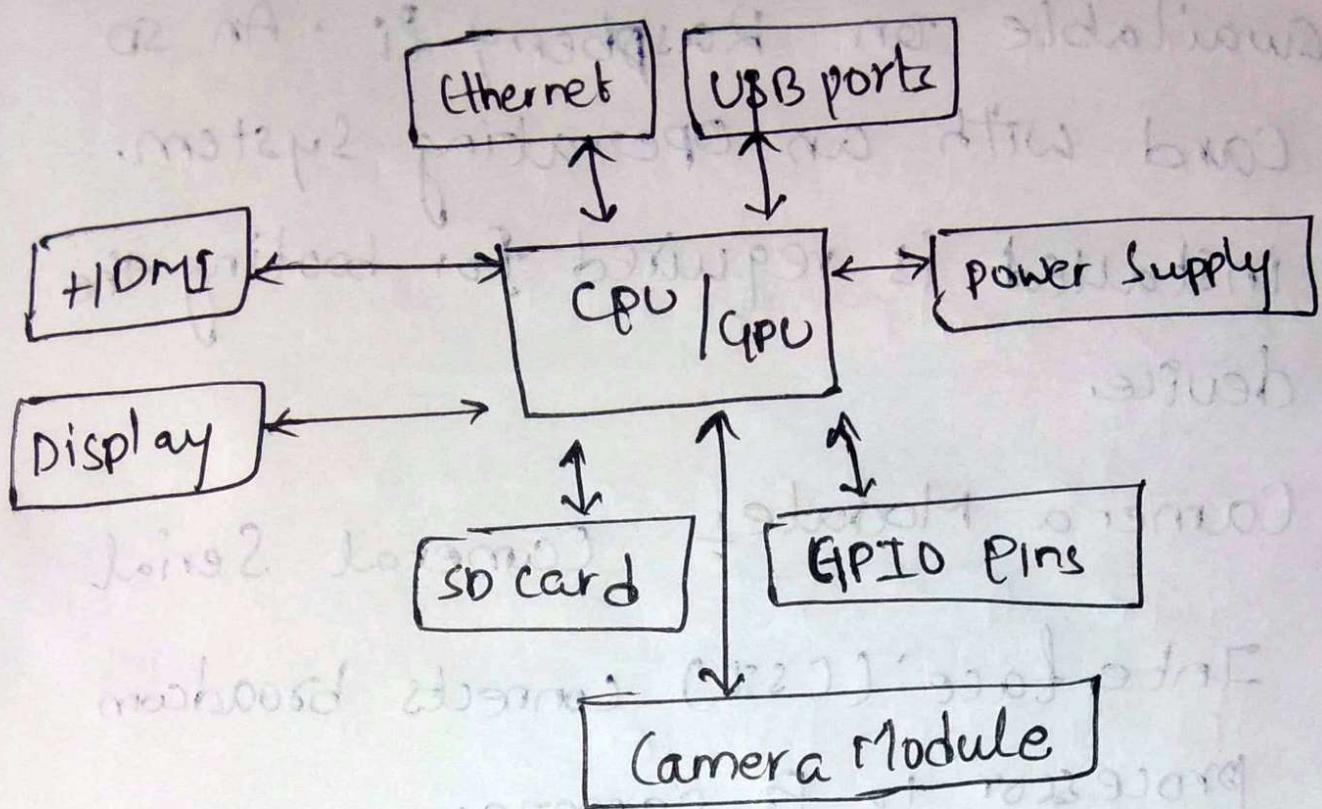
• Used for booting and storing the operating system and data).

5. Interfaces:

- GPIO
- USB ports
- HDMI port
- Audio jack
- Ethernet port
- Camera interface (CSI)
- Display interface (DSI)

6. Power Management;

- Power management IC (PMIC) for voltage regulation, power management.



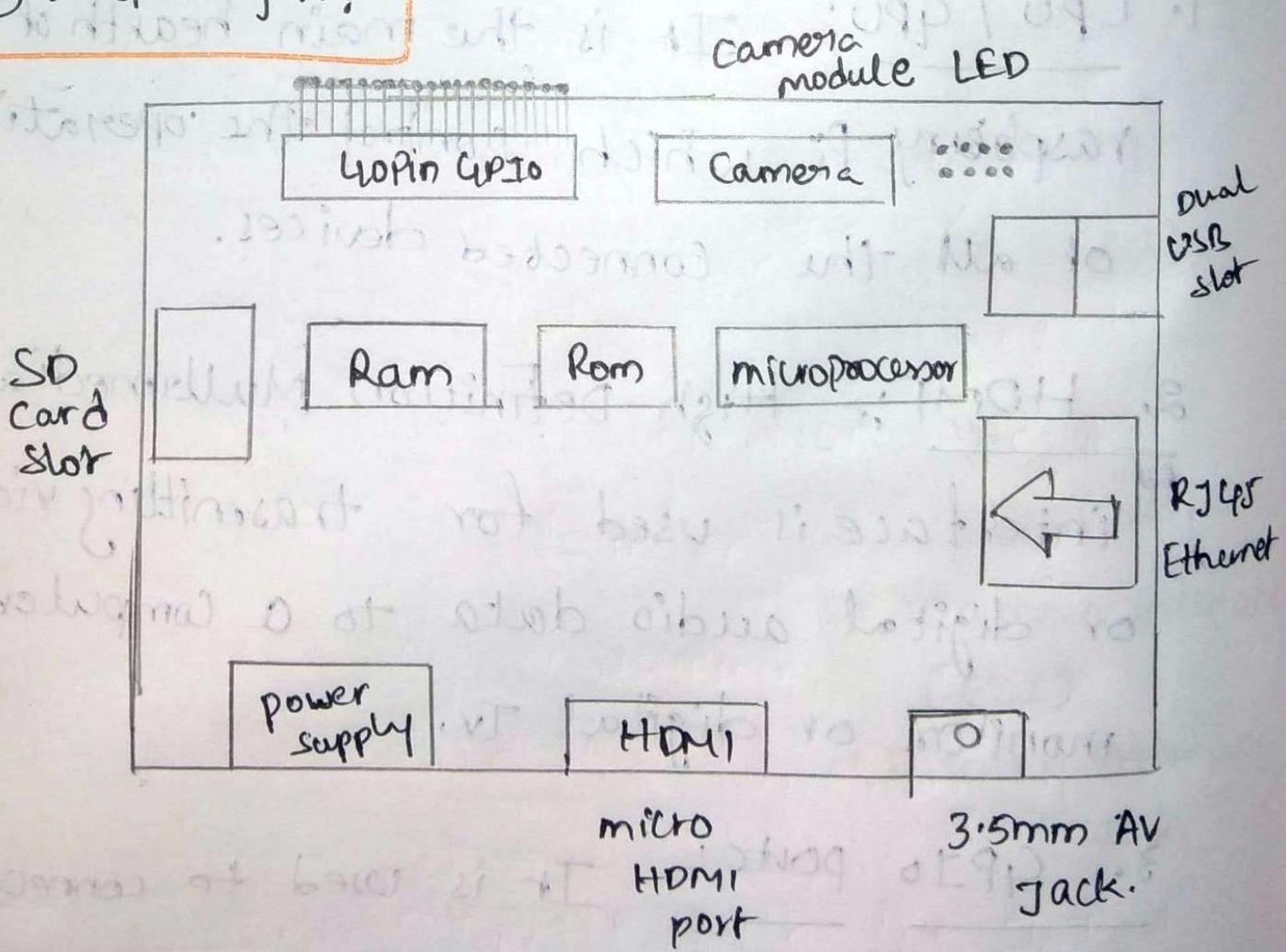
1. CPU / GPU: It is the main heart of the raspberry pi which control the operations of all the connected devices.

2. HDMI :- High Definition Multimedia interface is used for transmitting video or digital audio data to a computer monitor or digital tv.

3. GPIO ports : It is used to connect all periph external devices like sensors, motors, lights, etc.

- SD-card: The SD card slot is available on Raspberry Pi. An SD card with an operating system installed is required for booting the device.
- Camera Module: Camera Serial Interface (CSI) connects Broadcom processor to Pi camera.

* Raspberry Pi:



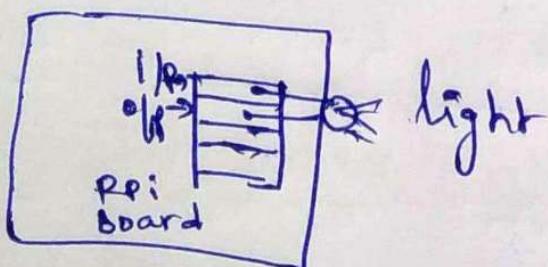
* Raspberry Pi Interfaces

Raspberry Pi boards provides a variety of interfaces that allow to connect peripherals, devices, sensors.

- * These interfaces enable users to interact with the raspberry Pi.

① GPIO Purpose
(General input / output pins)

- * These are helpful to connect & control the external devices.
- * There are 40 pins in the RPi.
- * We can connect sensors, actuators, lights, motors to raspberry Pi through it.



- These are used for:
 - pulse width modulation
 - Analog to digital conversion (ADC).

* It provides 2 rows of pins each.

Row 1: GPIO 0-19

Row 2: GPIO 20-39

2. USB ports:-

These are allowed user to connect peripheral devices like mice, keyboards.

- storage devices
- web cams
- printers
- The number of USB ports varies depending on raspberry pi model.

• model 4: 2 x USB 3.0 + 2 x USB 2.0

• model 3: 4 x USB 2.0

• model 2: 4 x USB 2.0

• model 0: 1 x USB 2.0

3. HDMI: (High Definition multimedia Interface) port:-

* By using this we can connect to TV or monitor.

• Video resolution upto 4K at 60Hz

• Audio output (Stereo or Surround sounds).

4. Audio Jacks

The audio jack 3.5mm Stereo audio connector that allows users to connect headphones, speakers.

5. Ethernet

Used to connect Raspberry Pi to wired network, enabling internet access.

6. Camera Interface

The Camera Interface (CSI) is a 15 pin connector that allows the user to connect a Raspberry Pi camera module, enabling camera functionality.

7. Micro - SD card slot

* It is used to store the operating system of Raspberry Pi, and data generated and gathered from sensors.

* Implementation of Raspberry Pi with IoT

- . Raspberry Pi is a popular choice for IoT projects. due to its flexibility, affordability, and easy of use.
- . Raspberry Pi can support different operating systems like Linux, Windows..etc
- It uses GPIO pins to connect IoT devices to raspberry Pi board.
- It reads the data from sensors and transform to IoT cloud.

Step 1:

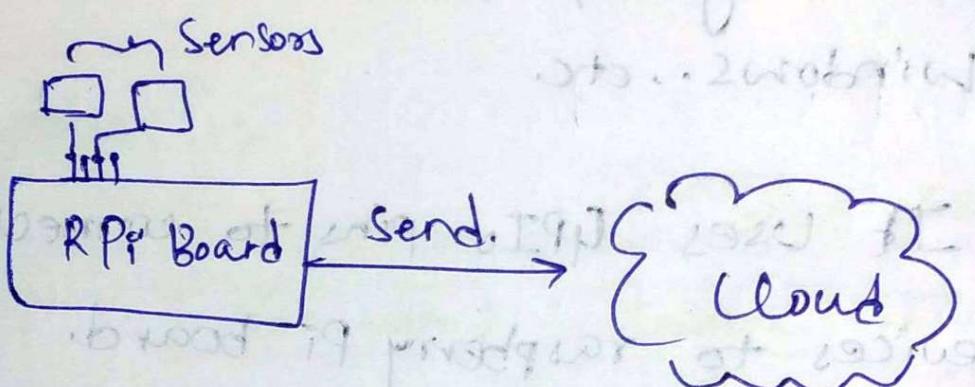
- Install raspbian OS on the raspberry pi board.
- Configure the network settings.
- Install all the necessary tools and IoT modules for our project.

* Step 2:

- Connect the sensors and actuators via GPIO pins.

* Step 3: Write the IoT code:

- Write a program to read data from the sensors and send it to the IoT platform or cloud service



- Use the IoT protocol library to establish a connection with the IoT platform or Cloud Service

- Use any programming language to write the logic for the IoT.

* Step 4: Integrate with the IoT platform / Cloud Services

- Create an account on the IoT platform or Cloud Service.

- Configure the IoT cloud platform to receive the data from Raspberry Pi.
- Use the cloud services to analyze the data.

Step 5: Deploy the IoT Applications

- Deploy the IoT application on the Raspberry Pi.
- Test the application to ensure it's working properly, or as expected.
- Debug any issues that arise during testing.

*. Blinking an LED with Raspberry Pi Using python

```
import RPI.GPIO as GPIO  
import time  
  
# Set up the GPIO module  
GPIO.setmode(GPIO.BCM)  
  
# Set up the LED pin as output.  
LED-PIN=17  
GPIO.setup(LED-PIN, GPIO.out)  
  
try:  
    while True:  
        # Turn on the LED  
        GPIO.output(LED-PIN,  
                    GPIO.High)  
        print("LED is on")  
        time.sleep(1) # wait for  
                      1 second.  
  
        # turn off the LED  
        GPIO.output(LED-PIN,  
                    GPIO.Low)
```

print("LED is off")

time.sleep(1) # wait for 1 second

except Keyboard Interrupt:

clean up the GPIO pins & exit
GPIO.cleanup()

• Requirements:

- Raspberry Pi model (any)
- Bread board and jumper wires.
- Resistor ($1k\Omega$, $2k\Omega$)
- LED (any colour)

* Connections:

- Connect LED's positive leg (anode) to raspberry pi GPIO pin 17.
- Connect LED negative to $1k\Omega$ or $2k\Omega$ resistor.
- Connect the other end of the resistor to a ground pin of raspberry Pi.

Notes Make sure to use the correct GPIO number and LED connections to avoid damaging your raspberry pi or LED.

☺ Subscribe to

☺ CodingFatake yt channel
for more like this ☺