National University of Computer & Emerging Sciences, Karachi
Computer Science Department
Spring 2023, Lab Manual – 06

Course Code: AI-2002	Course: Artificial Intelligence Lab
Instructor(s):	Kariz Kamal, Saeeda Kanwal, Sania Urooj, Shakir Hussain, Omer Qureshi, Muhammad Ali Fatmi

Lab Tasks

Contents:

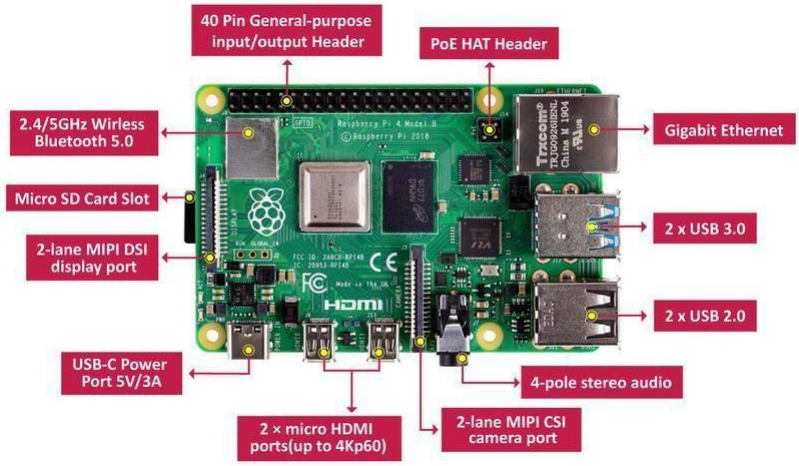
I. Introduction to Raspberry-Pi
II. Raspberry Pi 4 Tech Specs
III. Applications
IV. Complete View
V. Steps to follow
VI. Using direct image
VII. Useful commands
VIII. Precautions
IX. Arduino Uno Features
X. How to acquire
XI. Sketch
XII. Applications
XIII. Serial Communication
XIV. Where its held
XV. Tasks

Objective:

1. Getting familiar with Hardware
2. Installing Operating Systems
3. Implementing languages on Arduino and Raspberry Pi

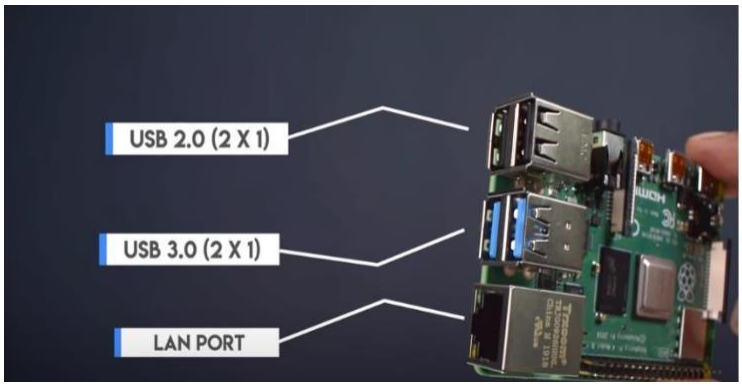
1. Raspberry Pi:

The Raspberry Pi is a low cost, credit-card sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python.



2. Raspberry Pi 4 Tech Specs

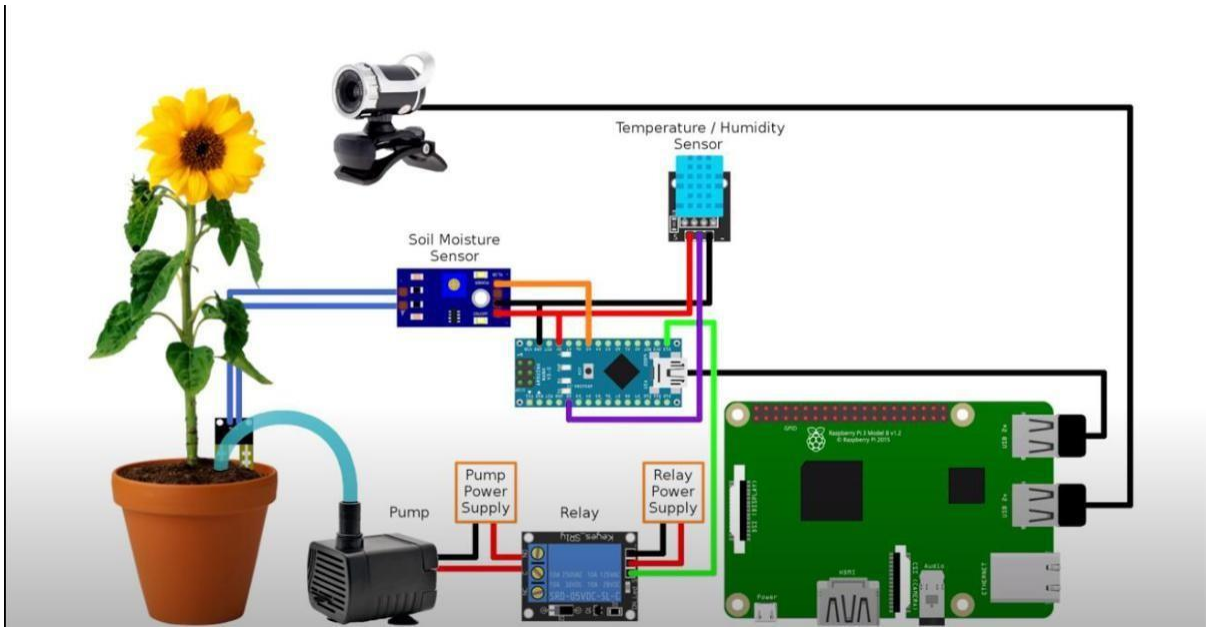
- Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz.
- 2GB, 4GB or 8GB LPDDR4-3200 SDRAM (depending on model)
- 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE.
- Gigabit Ethernet.
- 2 USB 3.0 ports; 2 USB 2.0 ports.
- 2 × micro-HDMI ports (up to 4kp60 supported)
- Dual display
- 5V DC via USB-C connector (minimum 3A*)



3. Applications & Features:

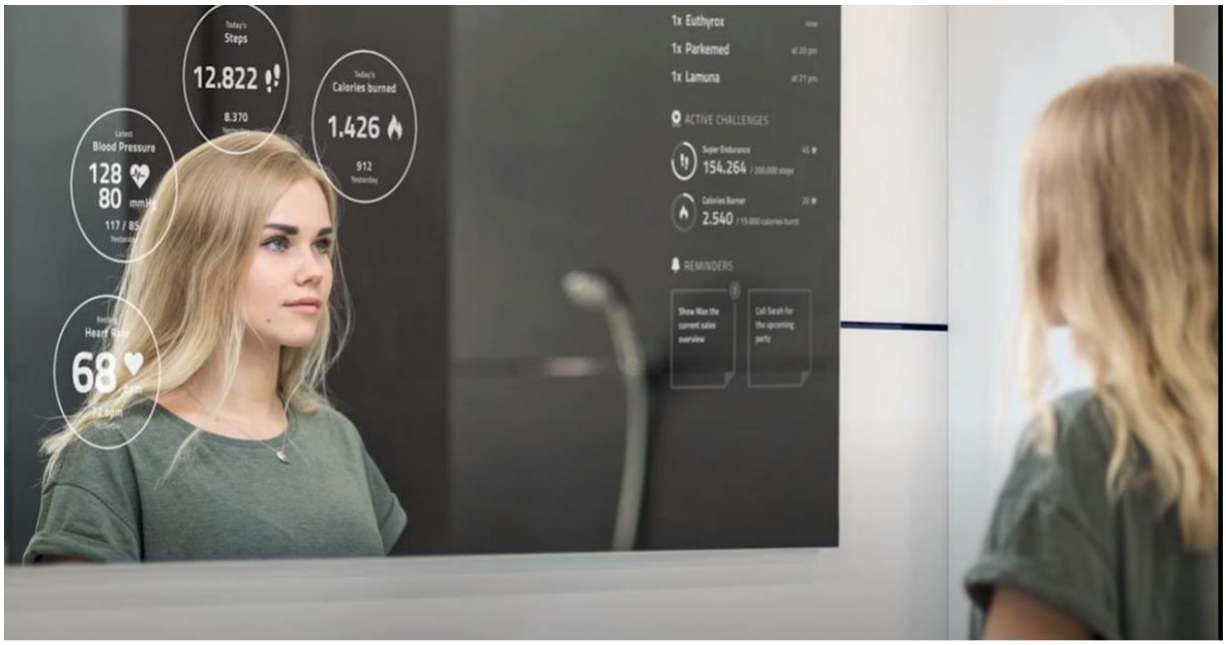
i) Smart automation

Smart Monitoring system where humans are unable to reach or its lethal for a human to be around.

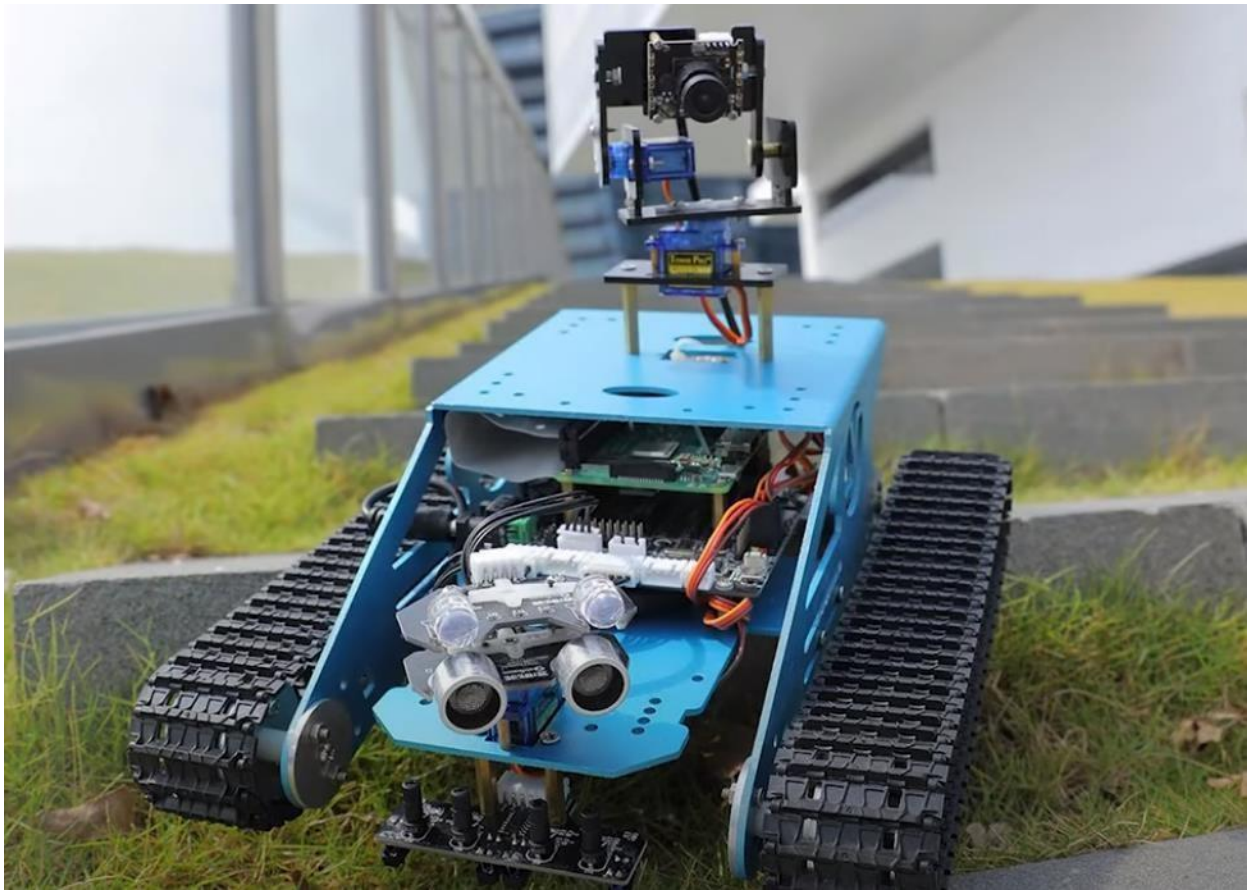


ii) Smart Mirror

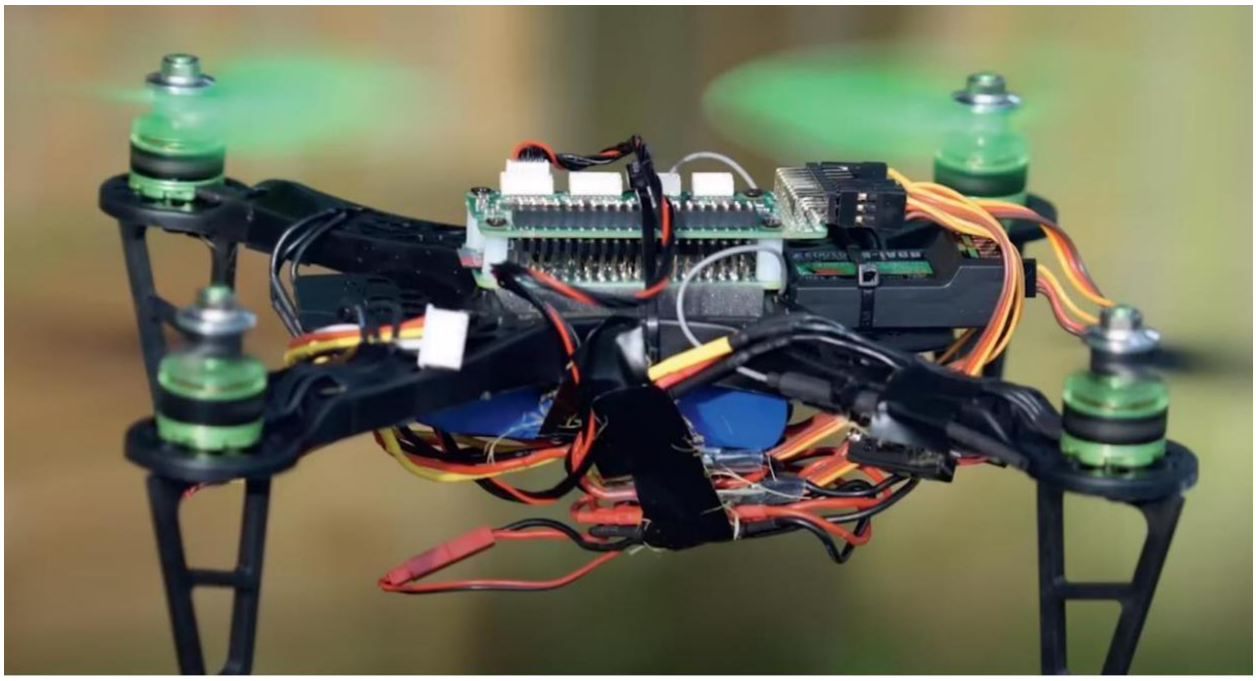
Can u guess the age of the person?? her BP??
her heartrate??

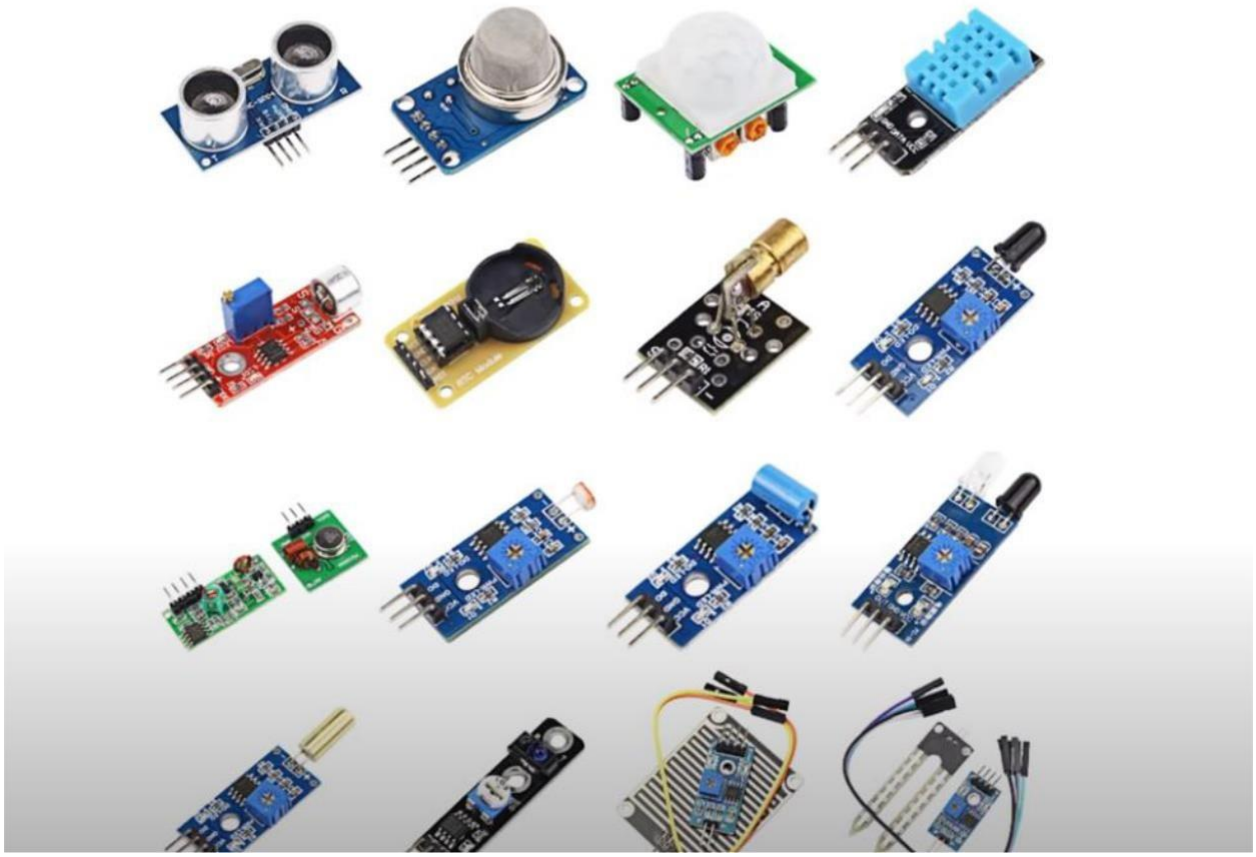


iii) <u>UGV</u>



iv) UAV

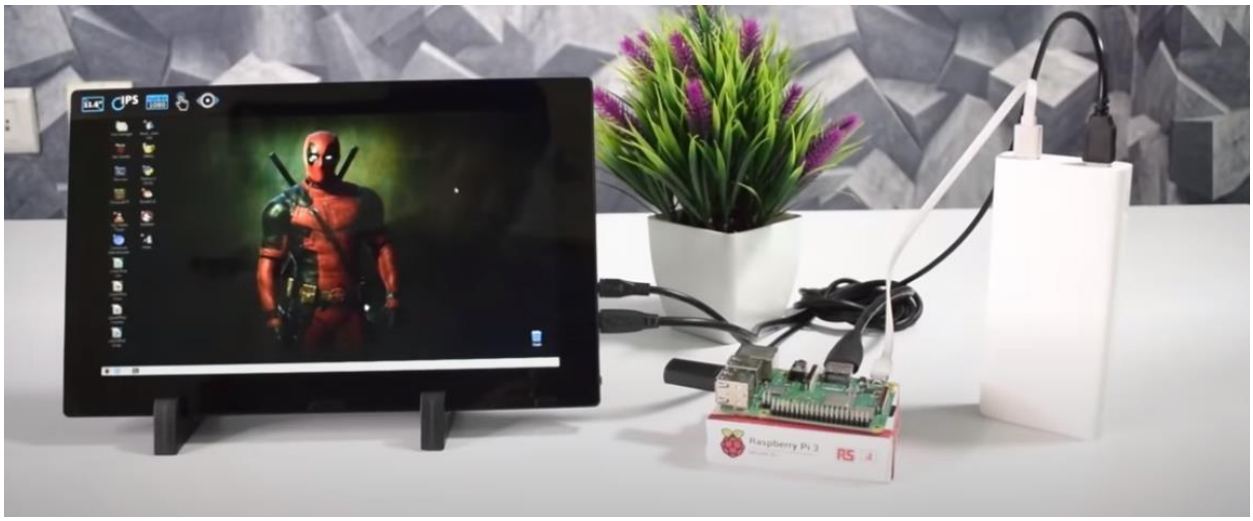




List of Sensors

v) In cyber world

4. A view of complete running module



5. Steps to follow:

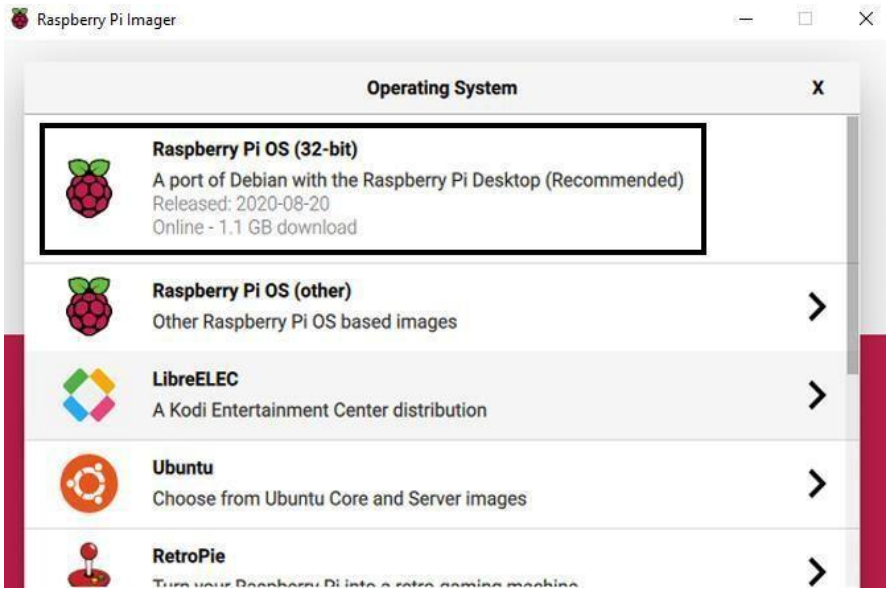
1. Download Raspbian or any related OS from the given links or use any other web source.

   a) https://www.raspberrypi.com/software/operating-systems/

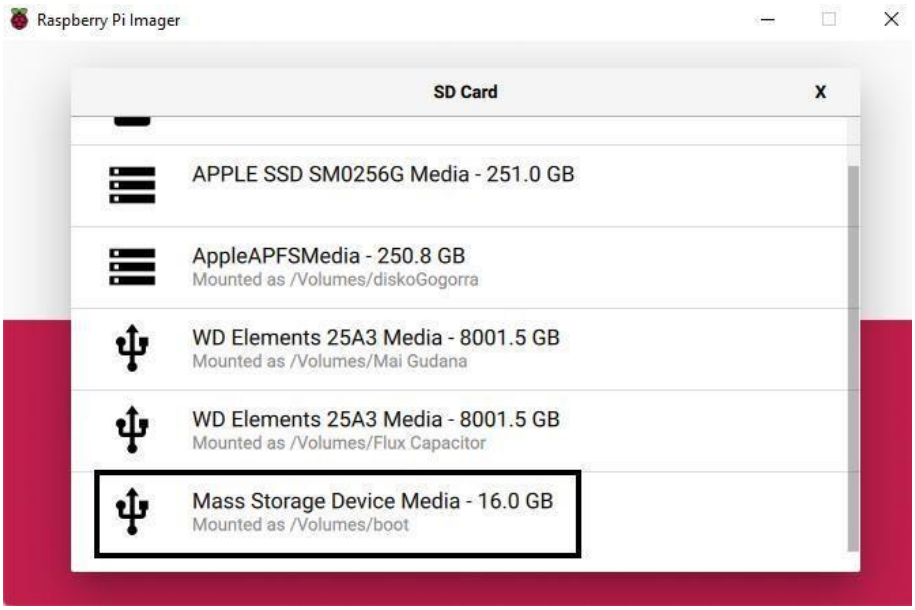
   b) https://www.raspberrypi.com/software/

2. Download Disk32Imager/Rufus or any other as shown below

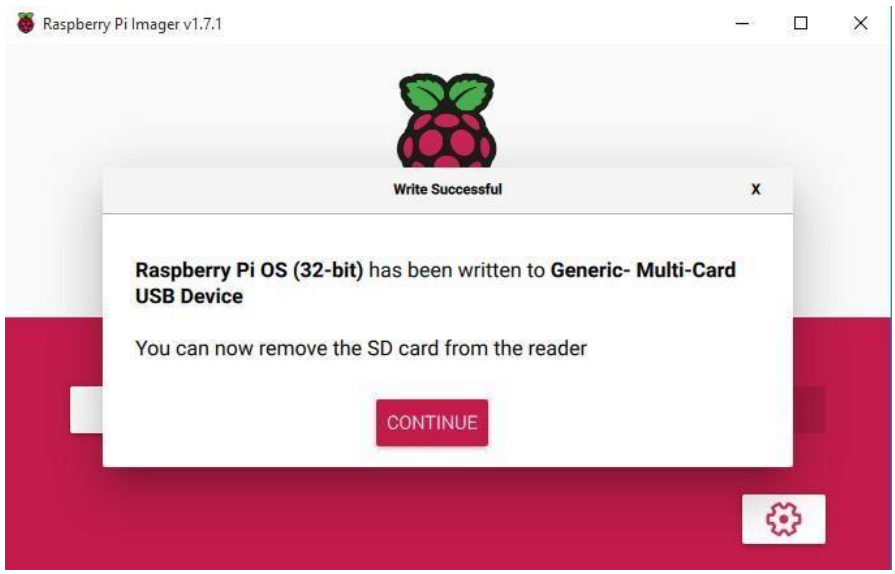
3. Select OS as recommended or any other from the given list



4. Select Storage and click on write



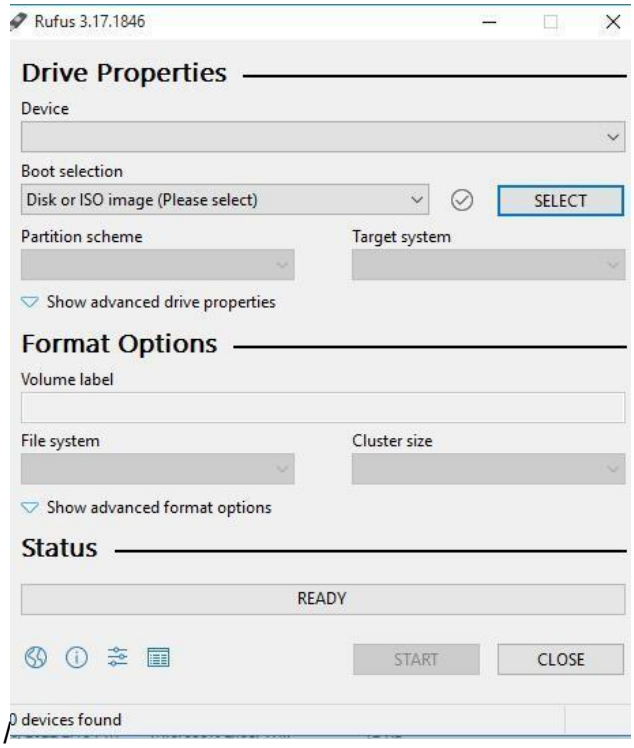
5. Once you get the following message, you can eject your SD card



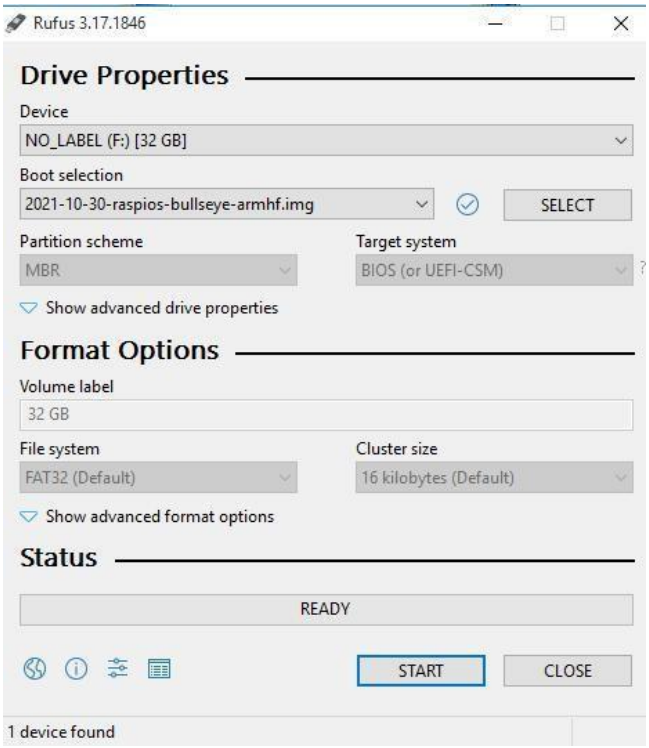
OR

6) Steps to follow using direct image

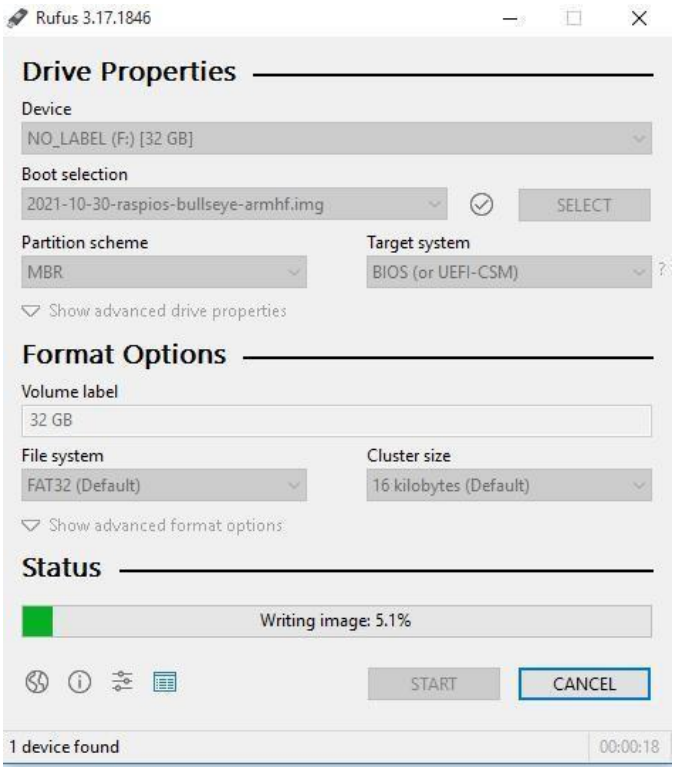
1. Select iso image



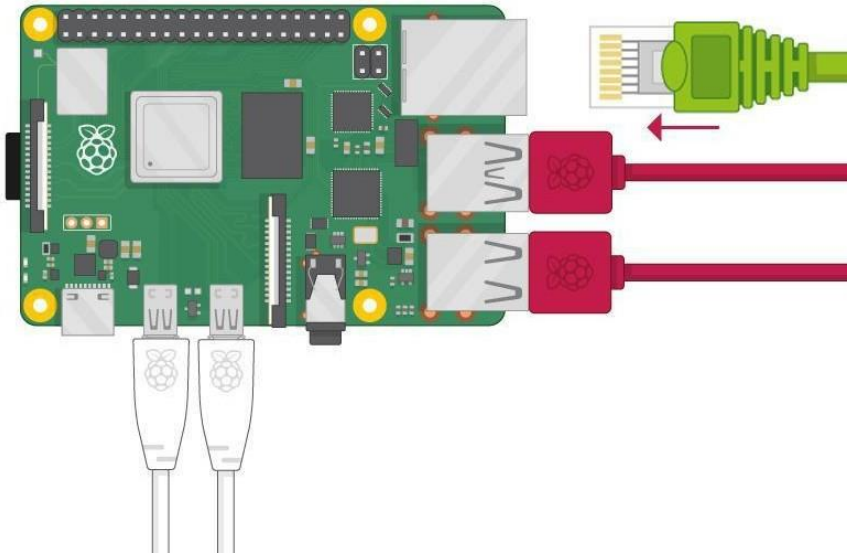
2. Select device and Click on Start



3. Let the image be completed



4. Plug the sd-card into Raspberry Pi
5. Connect all the peripherals(Keyboard, mouse, Lan cable, Display connector, Cam)



6. Power up the Pi

Example on Thonny IDE at Rpi:

```
import RPi.GPIO as GPIO
import time
from time import sleep

GPIO.setmode(GPIO.BOARD)
GPIO.setup(36,GPIO.OUT,initial=GPIO.LOW)
GPIO.setup(22,GPIO.OUT,initial=GPIO.LOW)

while True:
    GPIO.output(36,GPIO.HIGH)
    sleep(1)
    GPIO.output(36,GPIO.LOW)
    GPIO.output(22,GPIO.HIGH)
    sleep(1)
    GPIO.output(22,GPIO.LOW)
```

		Physical Pins			
Function	BCM	pin#	pin#	BCM	Function
3.3 Volts		1	2		5 Volts
GPIO/SDA1 (I2C)	2	3	4		5 Volts
GPIO/SCL1 (I2C)	3	5	6		GND
GPIO/GCLK	4	7	8	14	TX UART/GPIO
GND		9	10	15	RX UART/GPIO
GPIO	17	11	12	18	GPIO
GPIO	27	13	14		GND
GPIO	22	15	16	23	GPIO
3.3 Volts		17	18	24	GPIO
MOSI (SPI)	10	19	20		GND
MISO(SPI)	9	21	22	25	GPIO
SCLK(SPI)	11	23	24	8	CEO_N (SPI)
GND		25	26	7	CE1_N (SPI)
RESERVED		27	28		RESERVED
GPIO	5	29	30		GND
GPIO	6	31	32	12	GPIO
GPIO	13	33	34		GND
GPIO	19	35	36	16	GPIO
GPIO	26	37	38	20	GPIO
GND		39	40	21	GPIO

7) Useful commands on terminal:

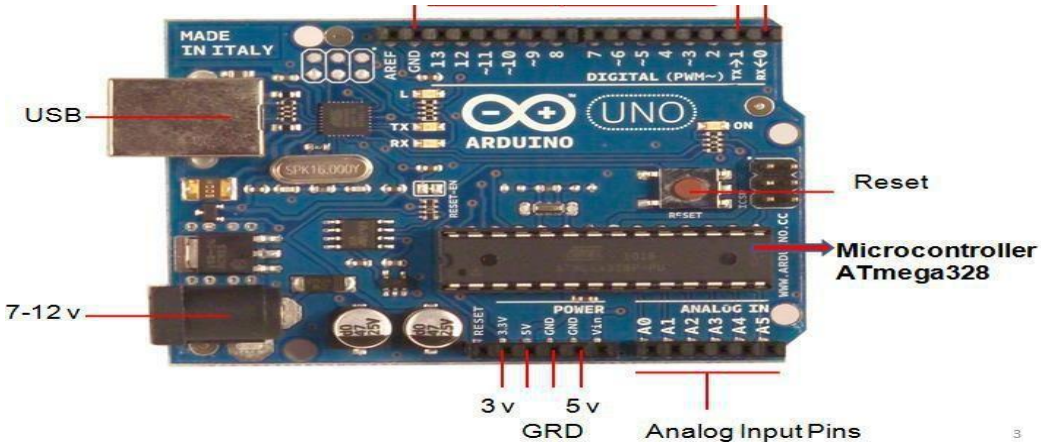
Sudo apt install-get update Sudo
apt install-get upgrade –y Sudo apt
sudo apt-get install fswebcam
fswebcam –r 1920x1080 name.jpg

8) Precautions:

Do not plug in or out any accessory(sd card, external internal peripherals etc) from Raspberry Pi when its powered on.

9) Arduino

- Arduino is a relatively inexpensive, yet versatile open-source microcontroller
- It is designed to facilitate interaction with the physical world via sensors while being able to perform calculations and various functions.



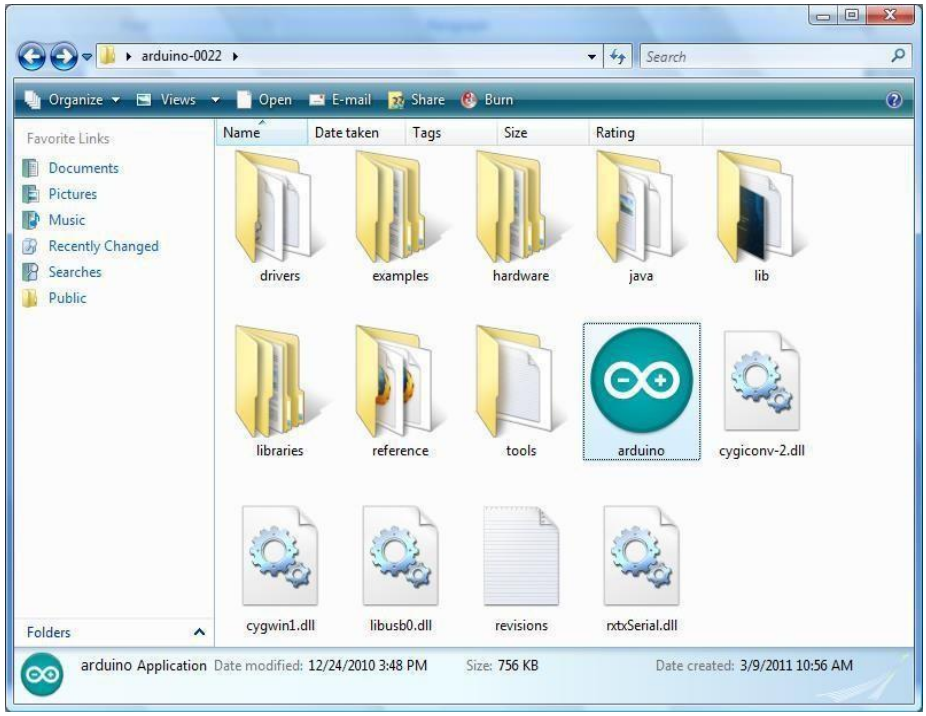
- Arduino (Uno) Board houses the ATmega328 chip, an 8-bit microcontroller.
- The board contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

10) Arduino Uno Features:

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 2 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

11) How to Acquire it??

- Download Arduino Software from Arduino.cc and unzip the folder to your computer. A file within the folder called Arduino, allows you to launch the programming environment.
- You need to install a driver that comes with Arduino to be able to communicate with the board



12) Steps for Code Compilation in Arduino IDE

- Open Arduino IDE and type the code in the front panel
- Press **Ctrl + S** or Icon on the IDE to save your code
- Click the Verify Button to check if the sketch has no errors.
- To select the board; go to
  Tools =>Board=>Arduino UNO
- Select the COM port to which board is connected;
  go to
  Tools =>Serial Port
  Note: the microcontroller board's COM no. is found near the name of the device's driver in ***Device Manager***.
- Connect your board to PC via USB cable
- Finally, upload the code into the board by *clicking the icon* in IDE

13) Sketch description:

- **setup()**

  The setup() function is called when a sketch starts. Use it to initialize variables, pin modes, start using libraries, etc. The setup function will only run once, after each powerup or reset of the Arduino board.
- **loop()**

  After creating a setup() function, which initializes and sets the initial values, the loop() function does precisely what its name suggests, and loops consecutively, allowing your program to change and respond. Use it to actively control the Arduino board.
- **pinMode()**

  Configures the specified pin to behave either as an input or an output. ***Syntax***:
  pinMode(pin, mode)
  Parameters: pin: the number of the pin whose mode you wish to set
  mode: INPUT, OUTPUT
- **digitalWrite() / analogWrite()**

  Write a HIGH or a LOW value to a digital pin. (5V for HIGH, 0V (ground) for LOW) ***Syntax***:
  digitalWrite(pin, value)
  Parameters: pin: the pin number
  value: HIGH or LOW
- **delay()**

  Pauses the program for the amount of time (in miliseconds) specified as parameter. (There are 1000 milliseconds in a second.)
  Syntax:
  delay(ms)
  Parameters: *ms: the number of milliseconds to pause*

- **digitalRead()**

  Reads the value from a specified digital pin, either HIGH or LOW.

  Syntax:

  digitalRead(pin)

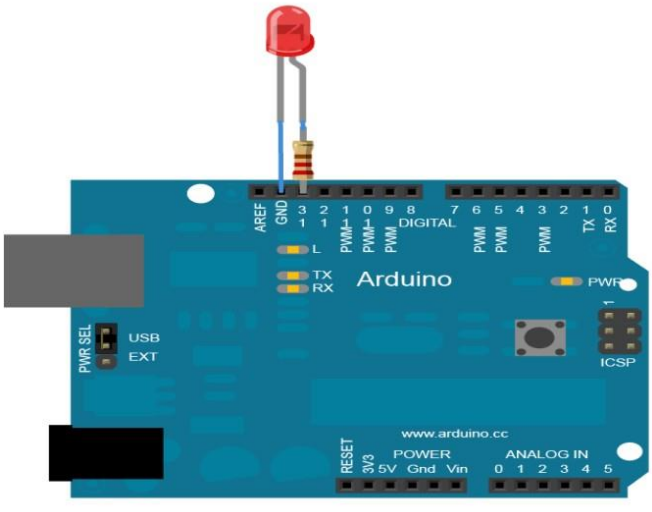
  Parameters: *pin: the number of the digital pin you want to read*

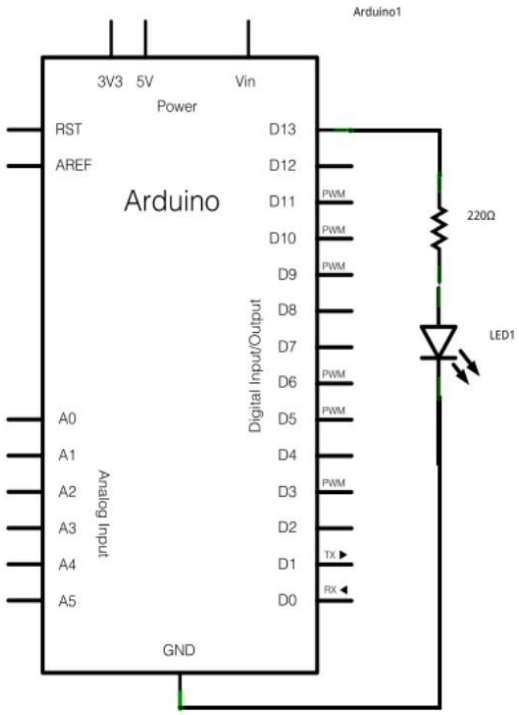
  Returns*: HIGH or LOW*

14) Sample Code:

```
void setup()
{
pinMode(13,OUTPUT);
}
void loop()
{
digitalWrite(13,HIGH);
delay(1000);
digitalWrite(13,LOW);
delay(1000);
}
```

15) Schematic/Block diagram behind:

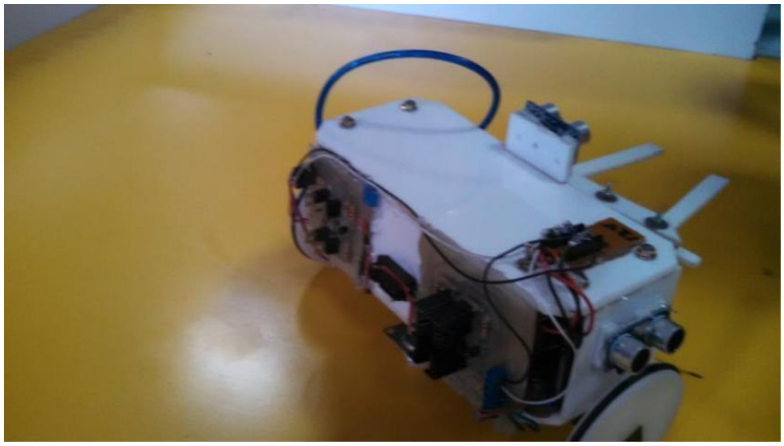


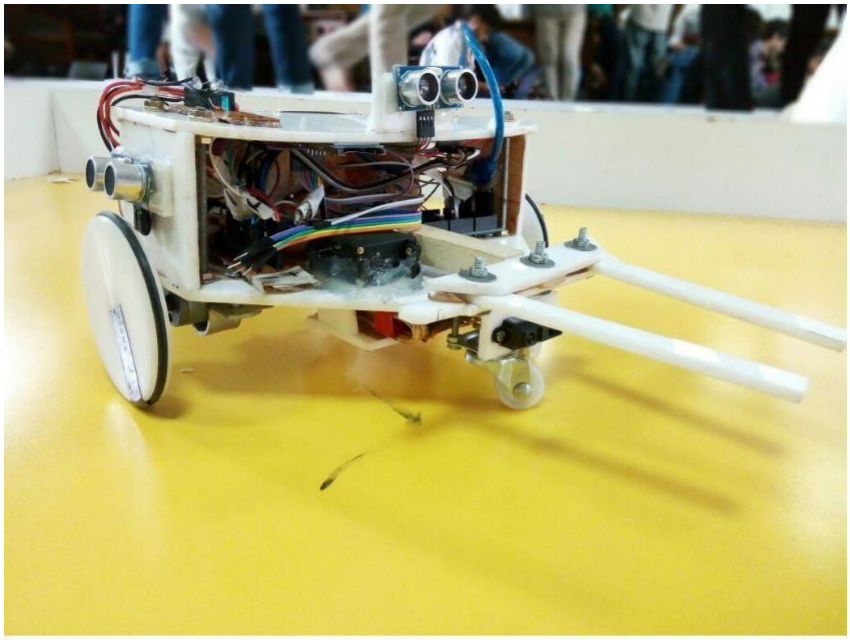


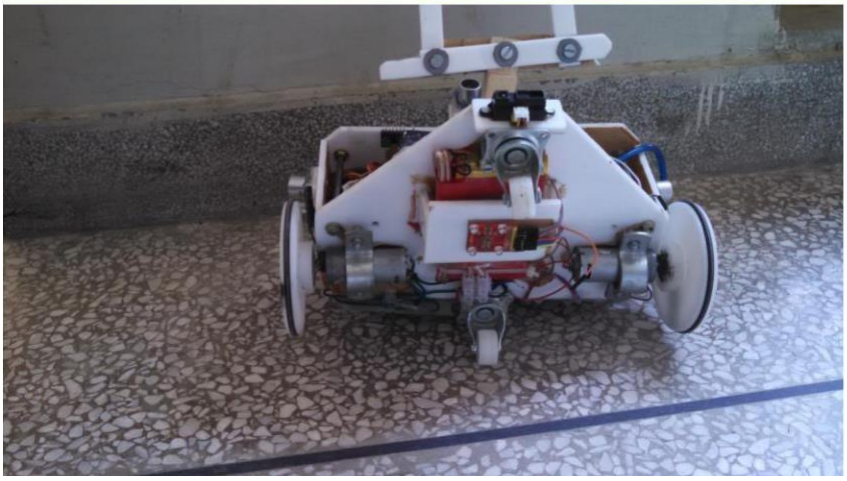
16) Application areas:

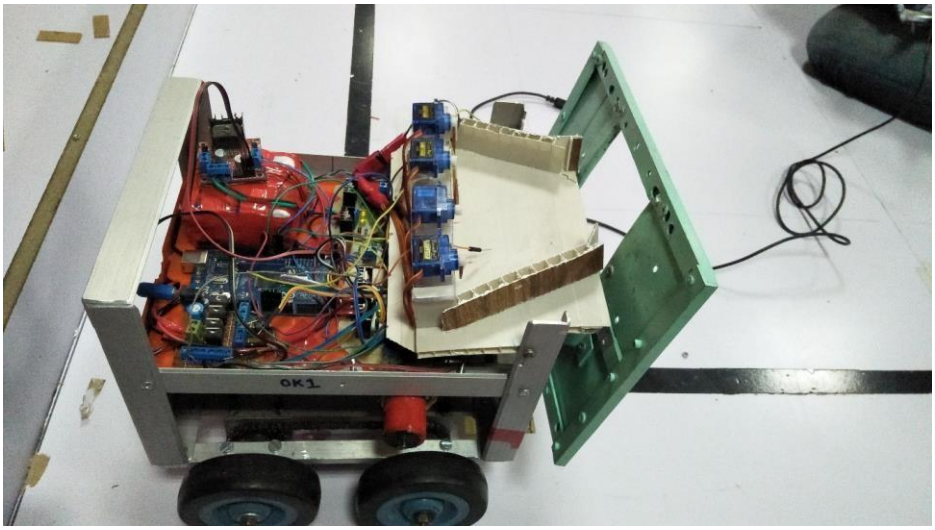
i. A Heterogeneous Humanoid Robot:

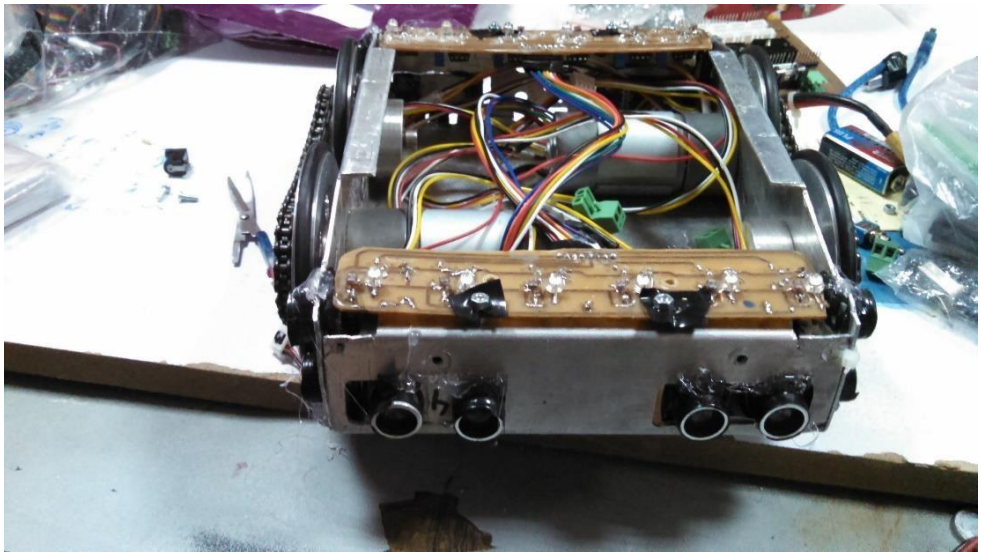
ii. An Indigenous category Robot:



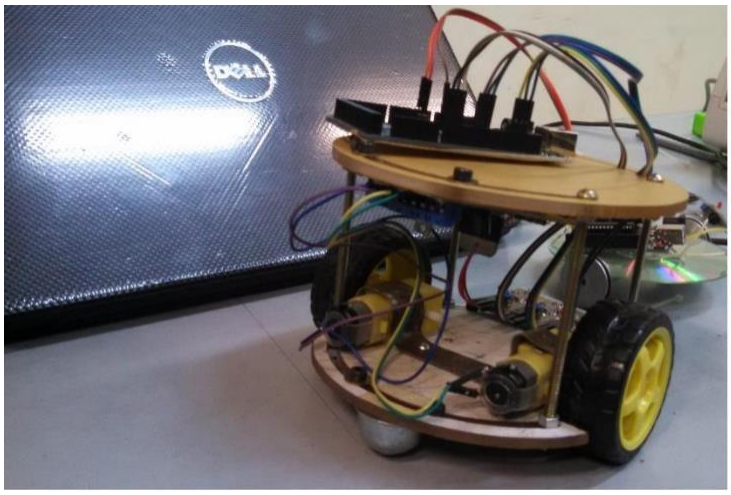


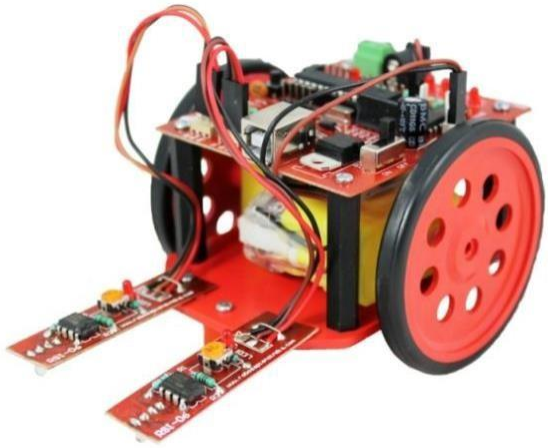


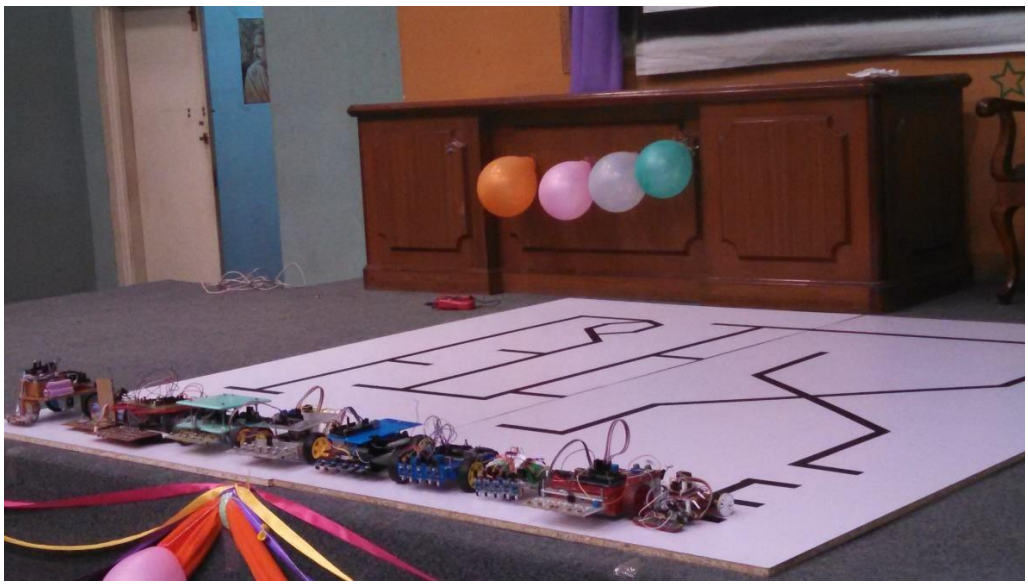

OK1

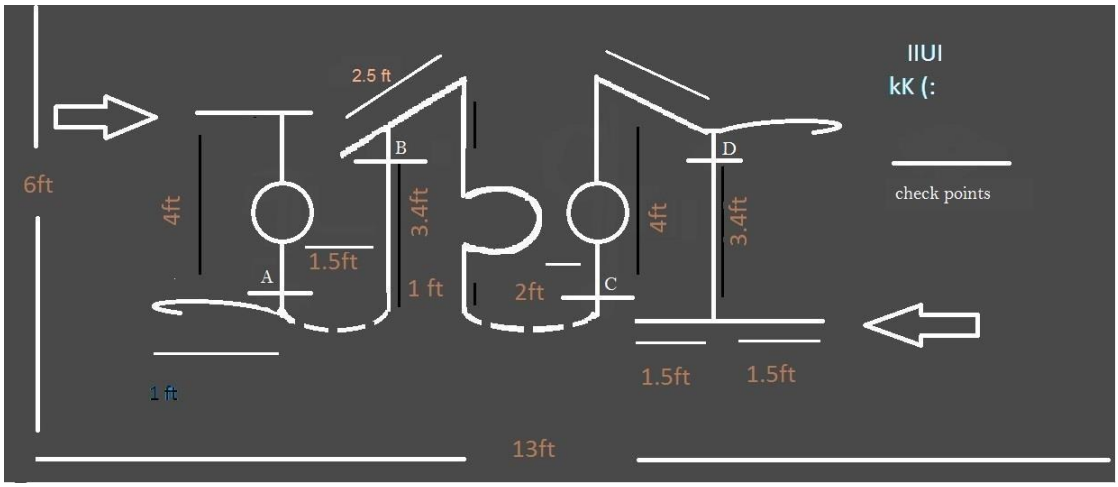


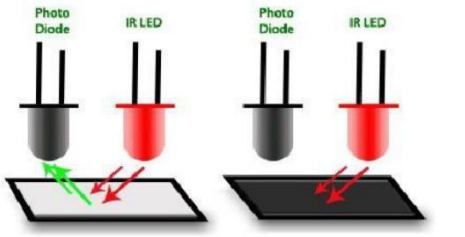
iii. LFR Category





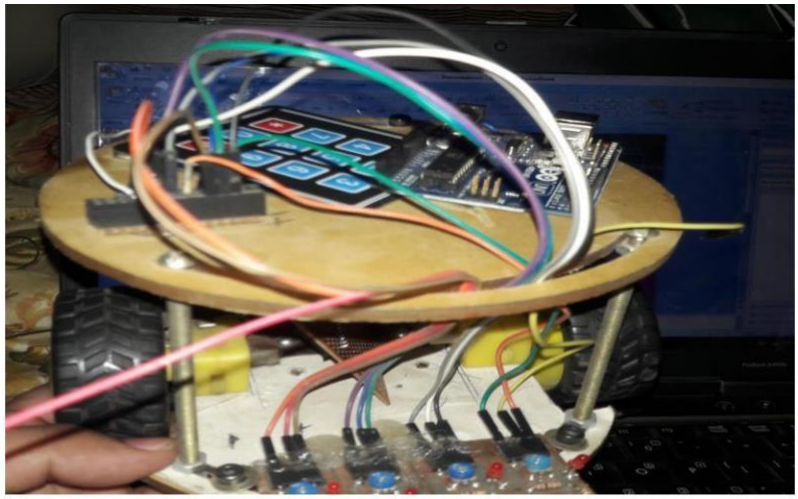


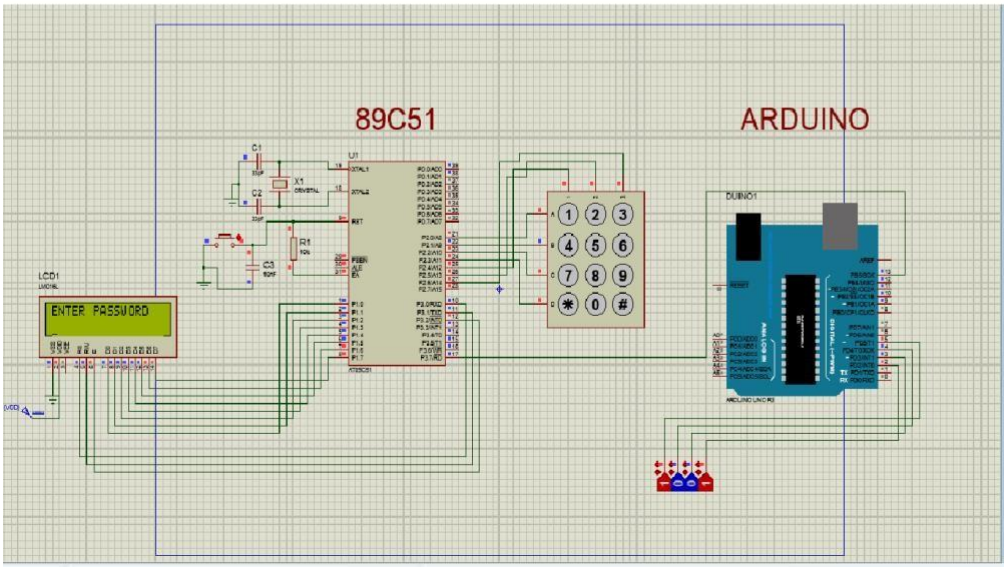




High Value of reflectance/voltage Low Value of reflectance/voltage

iv. Electronic Postman:





Schematic simulation:

17) Arduino's Serial COM:

- The Arduino Uno board can communicate with a PC.
- The board's microcontroller chip, ATmega328, provides the Universal Asynchronous Receiver/Transmitter (UART) TTL (5V) serial communication which is accessed by digital pins 0 (RX) and 1 (TX) for communication.
- The Arduino IDE has a serial monitor which can be used to send data to and from the Arduino Uno board.

18) Serial Communication:

- **Serial.print()**

Syntax:

Serial.print(val)
Serial.print(val, format)

Parameters:

val: the value to print

format: specifies the number base (for integral data types) or number of decimal places (for floating point types)

- **Example 1:**

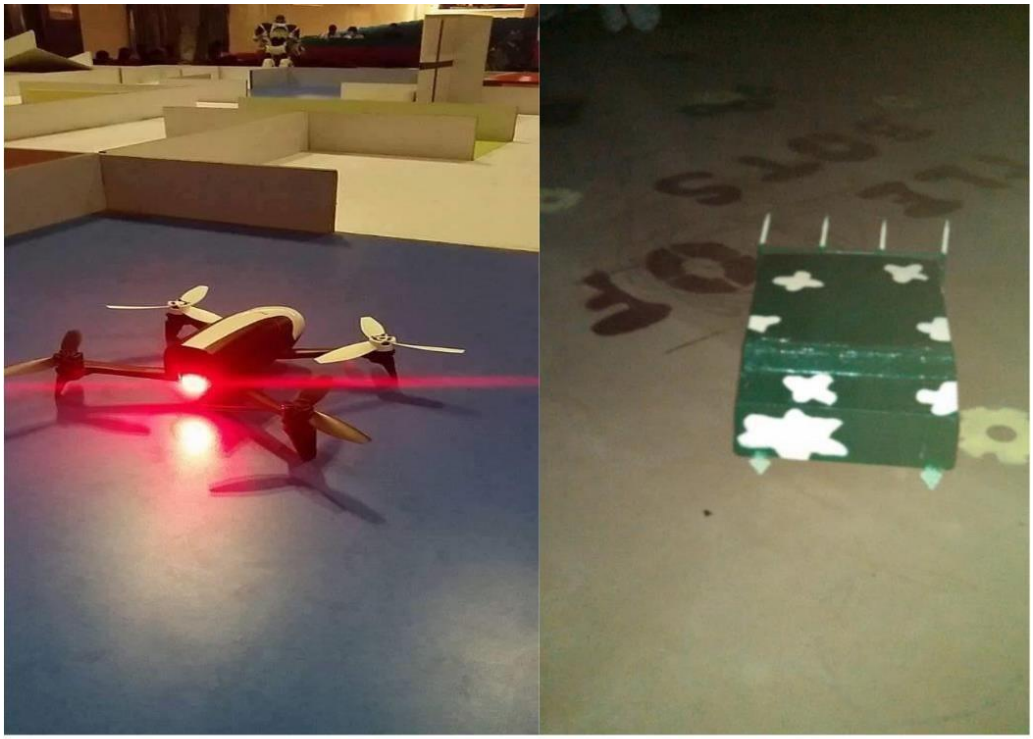
```
void setup()
{

Serial.begin(9600);
Serial.println(78) ;
Serial.println(1.23456);
Serial.println('N') ;
Serial.println("Hello world.") ;
Serial.println(78, BIN) ;
Serial.println(78, OCT) ;
Serial.println(78, DEC) ;
Serial.println(78, HEX) ;
Serial.println(1.23456, 2) ;
}
void loop()
{
// nothing to do here
}
```

- **Example 2:**

```
void setup()
{
int i = 27; // try different values for
this Serial.begin(9600);
// println is just print with an added newline character
Serial.println(i, DEC);
Serial.println(i, BIN);
Serial.println(i, HEX);
}
void loop()
{
// nothing to do here
}
```

19) Few Applications in Serial Comm:



20) Where its Held??

21) Links to refer:

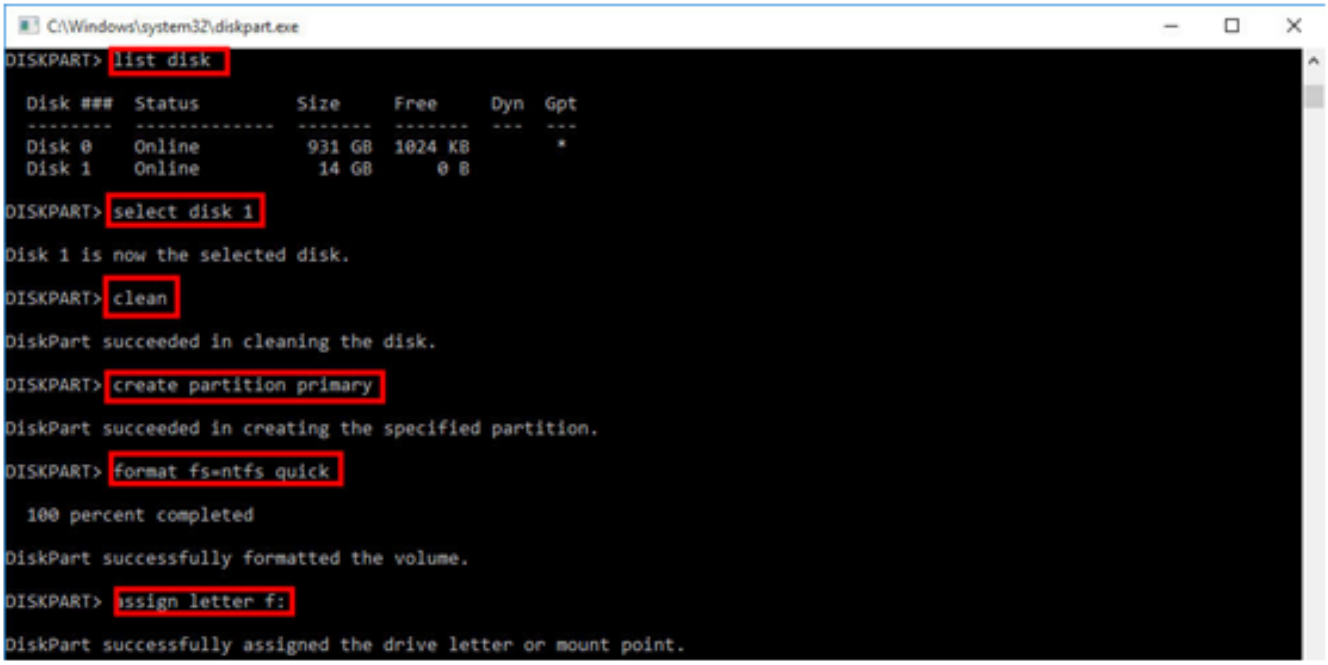
For Basics and ideas:

https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi
https://www.circuitbasics.com/raspberry-pi/
https://github.com/androidthings
https://thingspeak.com/login?skipSSOCheck=true
https://api.internetcollaboratif.info/#index

Commands to deal with:

https://www.raspberrypi.com/documentation/computers/os.html

For Cleaning up the card in case of boot failure in Rpi:



For downloading direct software:

https://www.raspberrypi.com/software/operating-systems/#raspberry-pi-desktop

For variety of images:

https://downloads.raspberrypi.org/raspios_arm64/images/

Tasks

Perform the following tasks using Arduino:

Task#01: Prepare a code for Line following robot in Arduino.

Description: A typical line follower robot has two sets of motors, let's call them left motor and right motor. Both motors rotate on the basis of the signal received from the left and the right sensors respectively. The robot needs to perform 4 sets of motion which includes moving forward, turning left, turning right and coming to a halt.

Task#02: Prepare a code for gradually change in the LED's brightness.

Perform the following tasks using Rpi:

Task#03: Download OS of Raspberry Pi and get the desktop ready for programming.

Task#04: Prepare a code for writing the values in multiple LEDs with 3 seconds delay.

Task#05: Explain in your words the detailed technical differences between Raspberry pi and Arduino and How they contribute in the world of AI with an example? (Max 750 words)

Task#06: Submit Project Proposal before Next Lab.