USER'S MANUAL

FOR

RSAPDUINO LAB

Manufactured By

LOGSUN SYSTEMS

4A, Shree Sadgurunivas Soc.,16/5/1 Hingane Khurd, Sinhgad Road, Pune-411051.
Ph.-020 24356456 / +91 8956711765

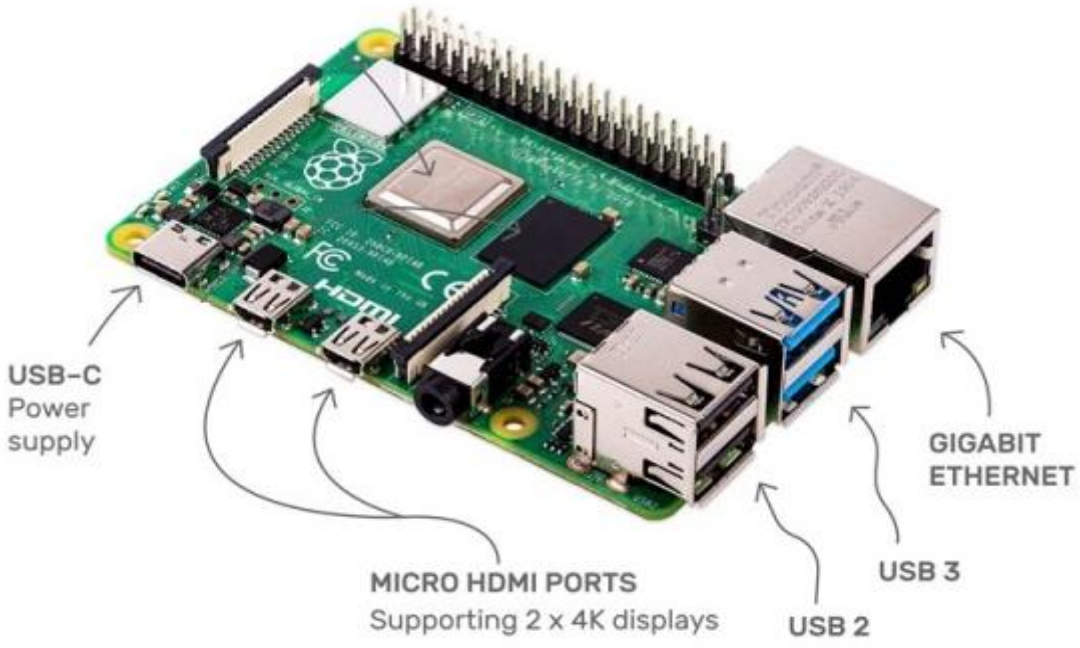
Email: - info@logsun.com

www.logsun.com

RASPDUINO DEVELOPMENT BOARD

INDEX

1. **Introduction of Raspberry Pi**



Raspberry Pi-4

The Raspberry Pi 2 delivers 6 times the processing capacity of previous models. This second generation Raspberry Pi has an upgraded Broadcom BCM2836 processor, which is a powerful ARM Cortex-A7 based quad-core processor that runs at 900MHz. The board also features an increase in memory capacity to 1Gbyte.

SPECIFICATION

Chip- Broadcom BCM2836 SoC

Core Architecture- Quad-core ARM Cortex-A7

CPU- 900MHz

GPU- Dual Core Video Core IV® Multimedia Co-Processor Provides Open GL ES 2.0, hardware-accelerated Open VG, and 1080p30 H.264 high-profile decode Capable of 1Gpixel/s, 1.5Gtexel/s or 24GFLOPs with texture filtering and DMA infrastructure.

Memory- 1GB LPDDR2.

Operating System- Boots from Micro SD card, running a version of the Linux operating system.

Dimensions- 85 X 56 X 17mm.

Power- Micro USB socket 5V, 2A.

CONNECTORS

Ethernet- 10/100 Base Ethernet socket

Video Output- HDMI (rev 1.3 & 1.4)

Audio Output- 3.5mm jack, HDMI

USB- 4 x USB 2.0 Connector

GPIO Connector- 40-pin 2.54 mm (100 mil) expansion header: 2x20 strip Providing 27 GPIO pins as well as +3.3 V, +5 V and GND supply lines

Camera Connector- 15-pin MIPI Camera Serial Interface (CSI-2)

JTAG- Not populated

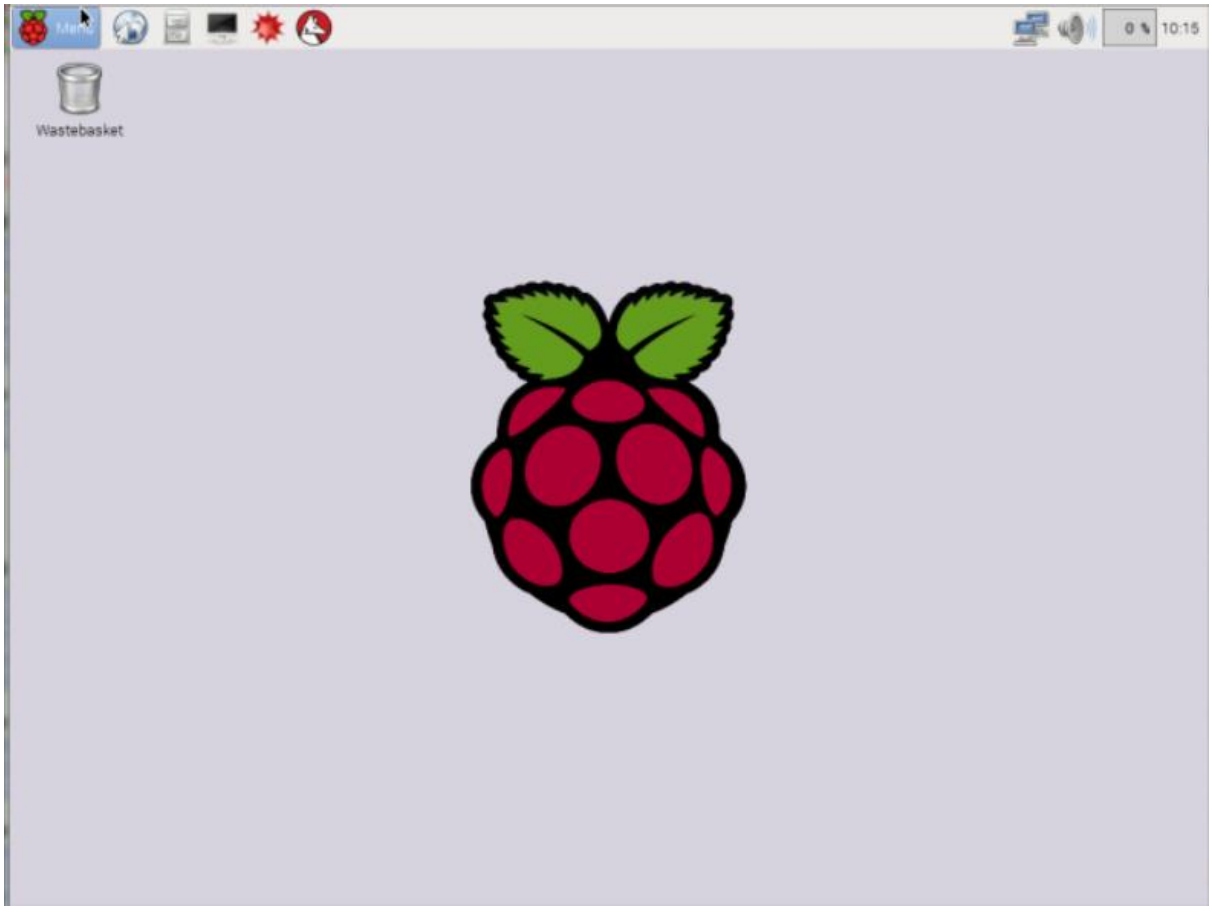
Display Connector- Display Serial Interface (DSI) 15 ways flat flex cable connector with two data lanes and a clock lane

Memory Card Slot- Micro SDIO

RASPBERRY PI SETTING:

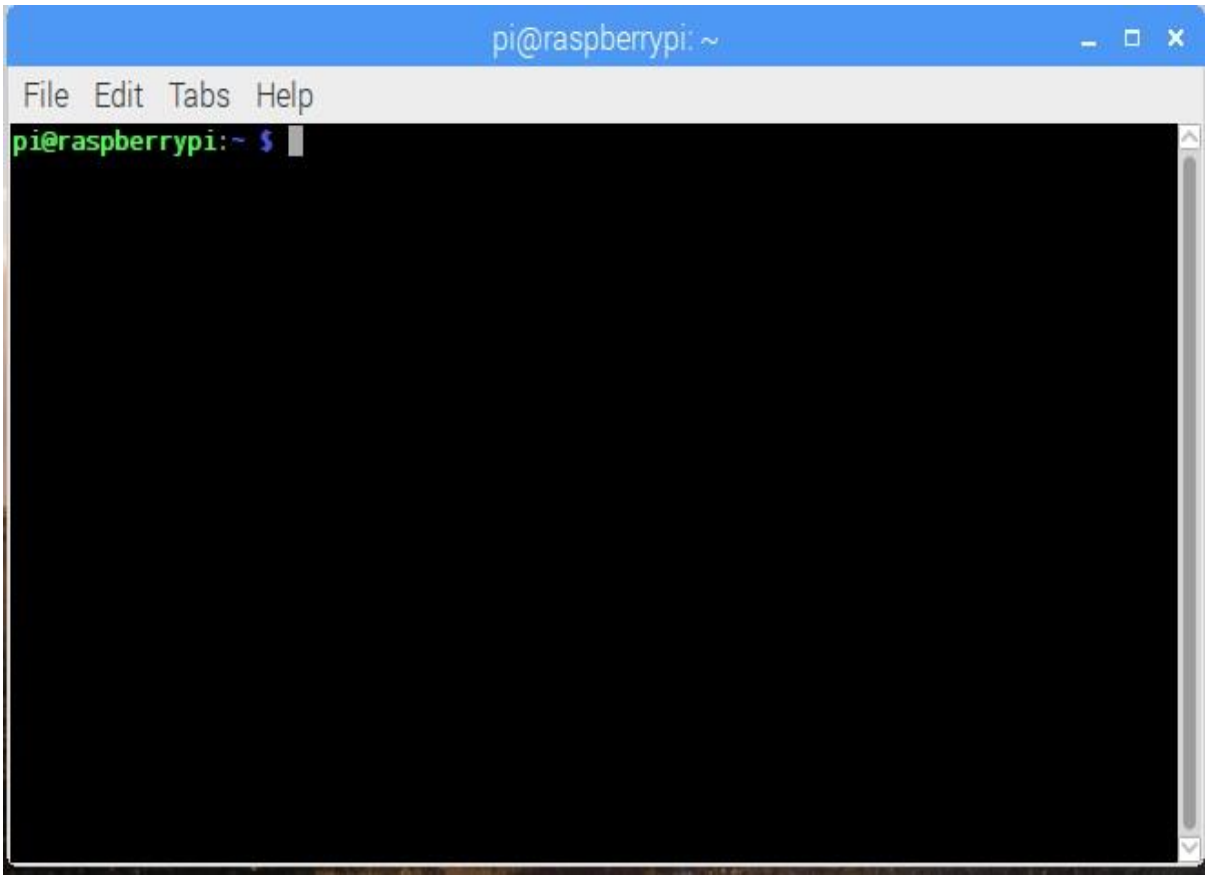
1) Plug in your keyboard, mouse and monitor cables (HDMI).
2) Now plug 5v, 2A power supply to your Raspberry pi board.
3) Your Raspberry pi will boot, and a window will appear with a list of different operating systems that you can install. We recommend that you use Raspbian and click on **Install.**
4) Raspbian will then run through its installation process. Note this can take a while.
5) When the install process has completed the Raspberry pi configuration menu (raspi-config) will load. Here you are able to set the time and date for your region. Enable a Raspberry pi camera board, or even create users. You can exit this menu by using **TAB** on your keyboard to move to **Finish.**

- **Raspberry Pi window will look like this:**



- **Opening a terminal window**

On the Raspberry pi (running Raspbian), the default terminal application is LXTerminal. This is known as a ‘terminal emulator’, this means that it emulates the old style video terminals in a graphical environment. The application can be found on the Raspberry Pi desktop and when started will look something like this



1. To create new file type “sudo nano example.py”
   .
2. Press **Enter** button.
3. Write a program in “example.py” editor window.
4. Then press “Ctrl X” and “Ctrl Y” to save the program.
5. Press **Enter** button.
6. Type “sudo python example.py” to run the program.
7. Press **Enter** button.

OR

You can refer our sample programs written in Python.

EXPERIMENT No 1-

LED LOGIC

AIM:- To study the interfacing of LED and SWITCH to the Raspberry pi.

REQUIREMENTS:- Raspberry pi board, 40 pin FRC cable, 26 pin FRC cable, 5v,2A power supply.

PROCEDURE:

1) Connect Raspberry pi board.

2) Connect the HDMI screen, mouse and keyboard to Raspberry pi.

3) Switch ON the power supply.

4) Connect 40 pin FRC cable from Raspberry pi to the 40 pin connector which is placed on the board. And connect 26 pin FRC cable from the board to "XBEE-LED-SWITCHES" section.

5) Open the Terminal window of Raspberry pi and enter the command >> **cd /home/pi/PYTHON PROGRAMMS.**

6) Now we are into the folder "PYTHON PROGRAMMS"

7) To see the list of files into PYTHON PROGRAMMS folder enter command >> **ls.**

8) To run this specific program enter the command >> **python ledlogic.py**

9) If you want to see program enter the command >> **sudo nano ledlogic.py**

10) To exit from the program press **Ctrl-Z.**

11) You will observe that LEDs will glow when you **ON** the switches and OFF when you **OFF** the switches.

Note: FRC port connection must be proper to run the program.

EXPERIMENT No 2.

LCD DISPLAY

AIM: - To study the interfacing of LCD (16*2) with the Raspberry pi.(4 bit)

REQUIREMENTS:- Raspberry pi board, 40 pin FRC cable, 26 pin FRC cable, 5v,2A power supply, LCD.

PROCEDURE:

1) Connect Raspberry pi board.
2) Connect the HDMI screen, mouse and keyboard to Raspberry pi.
3) Switch ON the power supply.
4) Connect 40 pin FRC cable from Raspberry pi to the 40 pin connector which is placed on the board. And connect 26 pin FRC cable from the board to “LCD-KEYPAD” section.
5) Open the Terminal window of Raspberry pi and enter the command >> **cd /home/pi/PYTHON PROGRAMMS.**
6) Now we are into the folder PYTHON PROGRAMMS
7) To see the list of files into PYTHON PROGRAMMS folder enter command >> **ls**
8) To run this specific program enter the command >> **python lcd.py**
9) If you want to see program enter the command >> **sudo nano lcd.py**
10) To exit from the program press **Ctrl-Z.**
11) You will observe that it displays **“LOGSUN SYSTEMS”** on LCD display.

THEORY:

Liquid crystal display:

LCD can be connected to the Raspberry pi through the GPIO pins. LCD is connected in the 4-bit mode or 8-bit mode. And the standard subroutine is given with the development board.

So that the application can be easily demonstrated and also for further implementation the subroutine can be easily embedded for which one has to do very few changes. Wide range of instruction functions: Clear displays, cursor home, display ON/OFF, cursor ON/OFF, cursor shift, display shift.

4-bit mode

RS, RW, EN are 1&2 are short.

8-bit mode

RS, RW, EN are 2&3 are short

- **INSTALLING THE RPLCD LIBRARY**

The RPLCD library can be installed from the Python Packages Index, or PIP. It might already be installed on your Pi, but if not enter this at the command prompt to install it.

sudo apt-get install python-pip

After you get PIP installed, install the RPLCD library by entering:

sudo pip install RPLCD

EXPERIMENT No 3.

LCD WITH KEYPAD INTERFACING

AIM: To study the interfacing of LCD and Keypad with Raspberry Pi.(4 bit)

REQUIREMENTS:- Raspberry pi board, 40 pin FRC cable, 26 pin FRC cable, 5v,2A power supply, LCD, 4*4 matrix keypad.

PROCEDURE:

1) Connect Raspberry pi board.
2) Connect the HDMI screen, mouse and keyboard to Raspberry pi.
3) Switch ON the power supply.
4) Connect 40 pin FRC cable from Raspberry pi to the 40 pin connector which is placed on the board. And connect 26 pin FRC cable from the board to “LCD-KEYPAD” section.
5) Open the Terminal window of Raspberry pi and enter the command >> **cd /home/pi/PYTHON PROGRAMMS.**
6) Now we are into the folder PYTHON PROGRAMMS.
7) To see the list of files into PYTHON PROGRAMMS folder enter command >> **ls**
8) To run this specific program enter the command >> **python keypad.py**
9) If you want to see program enter the command >> **sudo nano keypad.py**
10) To exit from the program press **Ctrl-Z.**
11) You will observe that when you press a key (0-F) it will show on LCD display.

EXPERIMENT No 4.
TRAFFIC LIGHT SIGNAL

AIM: To study the interfacing of LED to make traffic light signal with Raspberry Pi.

REQUIREMENTS:- Raspberry pi board, 40 pin FRC cable, 26 pin FRC cable, 5v,2A power supply.

PROCEDURE:

1) Connect Raspberry pi board.

2) Connect the HDMI screen, mouse and keyboard to Raspberry pi.

3) Switch ON the power supply.

4) Connect 40 pin FRC cable from Raspberry pi to the 40 pin connector which is placed on the board. And connect 26 pin FRC cable from the board to “TRAFFIC LIGHT” section.

5) Open the Terminal window of Raspberry pi and enter the command >> **cd /home/pi/PYTHON PROGRAMMS.**

6) Now we are into the folder PYTHON PROGRAMMS.

7) To see the list of files into PYTHON PROGRAMMS folder enter command >> **ls**

8) To run this specific program enter the command >> **python traffic.py**

9) If you want to see program enter the command >> **sudo nano traffic.py**

10) To exit from the program press **Ctrl-Z.**

11) You will observe that red led’s of 1st and 3rd stage is ON for 10 sec. Green led’s of both sides are on for 5 sec. Then yellow led’s 5 times blinks. And red led’s OFF. And same action happens for section 2nd and 4th.

EXPERIMENT No 5.

INTERFACING IR SENSOR WITH RASPBERRY PI

AIM: To study the interfacing of IR sensor with Raspberry Pi.

REQUIREMENTS:- Raspberry pi board, 40 pin FRC cable, 26 pin FRC cable, 5v,2A power supply.

PROCEDURE:

1) Connect Raspberry pi board.

2) Connect the HDMI screen, mouse and keyboard to Raspberry pi.

3) Switch ON the power supply.

4) Connect 40 pin FRC cable from Raspberry pi to the 40 pin connector which is placed on board. And connect 26 pin FRC cable from the board to “ADC-DAC-TMP-IR” section.

5) Open the Terminal window of Raspberry pi and enter the command >>
cd /home/pi/PYTHON PROGRAMMS.

6) Now we are into the folder PYTHON PROGRAMMS.

7) To see the list of files into PYTHON PROGRAMMS folder enter command >> **ls**

8) To run this specific program enter the command >> **python ir.py**

9) If you want to see program enter the command >> **sudo nano ir.py**

10) To exit from the program press **Ctrl-Z.**

11) You will observe that when IR is on or obstacle is detected that time “Ir is detected” prints on the terminal, or relay will switched on. And red led is on when ir Output is high.

Experiment No 6.

INTERFACING DC MOTOR TO RASPBERRY PI

AIM: To study the interfacing of dc motor to Raspberry pi.

REQUIREMENTS:- Raspberry pi board, 40 pin FRC cable, 26 pin FRC cable, 5v,2A power supply, DC motor, external12v dc power supply.

PROCEDURE:

1) Connect Raspberry pi board.

2) Connect the HDMI screen, mouse and keyboard to Raspberry pi.

3) Switch ON the power supply.

4) Connect 40 pin FRC cable from Raspberry pi to the 40 pin connector which is placed on the board. And connect 26 pin FRC cable from the board to "DC & STEPPER MOTOR" section, also connect the 12V power supply externally.

5) Open the Terminal window of Raspberry pi and enter the command >> **cd /home/pi/PYTHON PROGRAMMS.**

6) Now we are into the folder PYTHON PROGRAMMS.

7) To see the list of files into PYTHON PROGRAMMS folder enter command >> **ls**

8) To run this specific program enter the command >> **python dcmotor.py**

9) If you want to see program enter the command >> **sudo nano dcmotor.py**

10) To exit from the program press **Ctrl-Z**.

11) You will observe that when you press the switch **START** motor will start rotating in forward direction. And when you press the switch **REV** motor will rotate in opposite or reverse direction.

12) When you press switch **INC** the speed of motor increase slowly, as you press the same switch again. And when you press switch **DEC** the speed of motor decrease slowly, as you press same switch again. When you press switch **STOP** the motor will stop rotating.

NOTE: - Connect both jumpers (JP2 & JP3) on DC side.

Experiment No 7.

INTERFACING STEPPER MOTOR TO RASPBERRY PI

AIM: To study the interfacing of stepper motor to Raspberry pi.

REQUIREMENTS:- Raspberry pi board, 40 pin FRC cable, 26 pin FRC cable, 5v,2A power supply, stepper motor, external12v dc power supply.

PROCEDURE:

1) Connect Raspberry pi board.

2) Connect the HDMI screen, mouse and keyboard to Raspberry pi.

3) Switch ON the power supply.

4) Connect 40 pin FRC cable from Raspberry pi to the 40 pin connector which is placed on the board. And connect 26 pin FRC cable from the board to "DC & STEPPER MOTOR" section, also connect the 12V power supply externally.

5) Open the Terminal window of Raspberry pi and enter the command >> **cd /home/pi/PYTHON PROGRAMMS.**

6) Now we are into the folder PYTHON PROGRAMMS

7) To see the list of files into PYTHON PROGRAMMS folder enter command >> **ls**

8) To run this specific program enter the command >> **python stepper.py**

9) If you want to see program enter the command >> **sudo nano stepper.py**

10) To exit from the program press **Ctrl-Z**.

11) You will observe that when you press the switch **START** motor will start rotating stepwise in forward direction. And when you press the switch **REV** motor will rotate in opposite or reverse direction.

12) When you press switch **INC** the speed of motor increase slowly, as you press the same switch again. And when you press switch **DEC** the speed of motor decrease slowly, as you press same switch again.

13) When you press switch **STOP** the motor will stop rotating.

NOTE:- Connect both jumpers(JP2 &JP3) on STP side.

Experiment No 8.

INTERFACING RELAY WITH RASPBERRY PI

AIM: To study interfacing of relay with Raspberry pi.

REQUIREMENTS:- Raspberry pi board, 40 pin FRC cable, 26 pin FRC cable, 5v,2A power supply.

PROCEDURE:

1) Connect Raspberry pi board.
2) Connect the HDMI screen, mouse and keyboard to Raspberry pi.
3) Switch ON the power supply.
4) Connect 40 pin FRC cable from Raspberry pi to the 40 pin connector which is placed on the board. And connect 26 pin FRC cable from the board to “ADC-DAC-TMP-IR” section.
5) Open the Terminal window of Raspberry pi and enter the command >> **cd /home/pi/PYTHON PROGRAMMS.**
6) Now we are into the folder PYTHON PROGRAMMS
7) To see the list of files into PYTHON PROGRAMMS folder enter command >> **ls**
8) To run this specific program enter the command >> **python relay.py**
9) If you want to see program enter the command >> **sudo nano relay.py**
10) To exit from the program press **Ctrl-Z.**
11) You will observe that relay is ON for 5sec and OFF for 5 sec.

Experiment No 9.

LIFT ELEVATOR SIMULATOR

AIM: To study Lift Elevator Simulator.

REQUIREMENTS:- Raspberry pi board, 40 pin FRC cable, 26 pin FRC cable, 5v,2A power supply.

PROCEDURE:

1) Connect Raspberry pi board.

2) Connect the HDMI screen, mouse and keyboard to Raspberry pi.

3) Switch ON the power supply.

4) Connect 40 pin FRC cable from Raspberry pi to the 40 pin connector which is placed on the board. And connect 26 pin FRC cable from the board to "LIFT ELEVATOR SIMULATOR" section.

5) Open the Terminal window of Raspberry pi and enter the command >> **cd /home/pi/PYTHON PROGRAMMS.**

6) Now we are into the folder PYTHON PROGRAMMS

7) To see the list of files into PYTHON PROGRAMMS folder enter command >> **ls**

8) To run this specific program enter the command >> **python lift_rasp.py**

9) If you want to see program enter the command >> **sudo nano lift_rasp.py**

10) To exit from the program press **Ctrl-Z.**

11) You will observe that when you press the switch **Ground** the **zero** will display on 7-segment display and green led is on. And when you press the switch **Top** the **one** will display on 2nd display and yellow led is high, then after some delay **two** display on 3rd display and yellow led is high. Again after some delay **three** displays on 4th display and green led is high.

12) The above procedure is same for 2nd floor and 1st floor.

NOTE- Connect jumper to vcc or ground as per requirement for pull up or pull down the switches.

Experiment No 10.

INTERFACING LM-35 WITH RASPBERRY PI

AIM: Interface LM-35 to Raspberry pi.

REQUIREMENTS:- Raspberry pi board, 40 pin FRC cable, 26 pin FRC cable, 5v,2A power supply, LM-35(Temperature sensor).

PROCEDURE:

1) Connect Raspberry pi board.

2) Connect the HDMI screen, mouse and keyboard to Raspberry pi.

3) Switch ON the power supply.

4) Connect 40 pin FRC cable from Raspberry pi to the 40 pin connector which is placed on the board. And connect 26 pin FRC cable from the board to "ADC-DAC-TMP-IR" section.

5) Open the Terminal window of Raspberry pi and enter the command >> **cd /home/pi/PYTHON PROGRAMMS.**

6) Now we are into the folder PYTHON PROGRAMMS

7) To see the list of files into PYTHON PROGRAMMS folder enter command >> **ls**

8) To run this specific program enter the command >> **python adc.py**

9) If you want to see program enter the command >> **sudo nano adc.py**

10) To exit from the program press **Ctrl-Z.**

11) You will observe that when resistor is heated LM-35 sensor sense the temperature and the output will shows on serial monitor.

12) Also you can see the output on LED's by varying the pot.

NOTE- the LM-35 sensor gives the Analog output. So we use the PC8951 ADC IC to convert Analog output into Digital. You can also use a POT as a analog input and shows the output on led's for study of ADC with raspberry pi.

CONFIGURATION

- The I2C peripheral is not turned on by default. To enable it, do the following.

1. Run `sudo raspi-config`.
2. Use the down arrow to select `9 Advanced Options`
3. Arrow down to `A7 I2C`.
4. Select `yes` when it asks you to enable I2C,
5. Also select `yes` when it asks about automatically loading the kernel module.
6. Use the right arrow to select the `<Finish>` button.
7. Select `yes` when it asks to reboot.

- Now install the i2c-tools package by:

sudo apt-get install i2c-tools

- If you get a 404 error do an update first:

sudo apt-get update

- Then run the install the i2c-tools again.

Note : The installation could take a few minutes to do, depend on how busy the server is.

- After the reboot test to see any device connected by:

sudo i2cdetect -y 1

Experiment No 10.

INTERFACING ZIGBEE WITH RASPBERRY PI

AIM: Zigbee interface to Raspberry pi.

REQUIREMENTS:- Raspberry pi board, 40 pin FRC cable, 26 pin FRC cable, 5v,2A power supply, LM-35(Temperature sensor).

PROCEDURE:

1) Connect Raspberry pi board.

2) Connect the HDMI screen, mouse and keyboard to Raspberry pi.

3) Switch ON the power supply.

4) Connect 40 pin FRC cable from Raspberry pi to the 40 pin connector which is placed on the board. And connect 26 pin FRC cable from the board to “ZIGBEE” section.

5) Configure ZigBee modules first as transmitter and second as a receiver. Configuration procedure will show the link given below. If once configured there is no need to configure again and again.

6) For this experiment we need two Raspberry Pi board.

7) Open the Terminal window of Raspberry pi and enter the command >> **cd /home/pi/PYTHON PROGRAMMS**.

8) Now we are into the folder PYTHON PROGRAMMS.

9) To see the list of files into PYTHON PROGRAMMS folder enter command >> **ls**

10) **I**) First load (**Receiver**) program to Onboard Raspberry Pi enter command >> **python recexbee.py**
II) Then load (**transmitter**)program in other Raspberry Pi enter command >> **python transxbee.py**

11) If you want to see program enter the command >> **sudo nano xbee.py**

12) To exit from the program press **Ctrl-Z.**

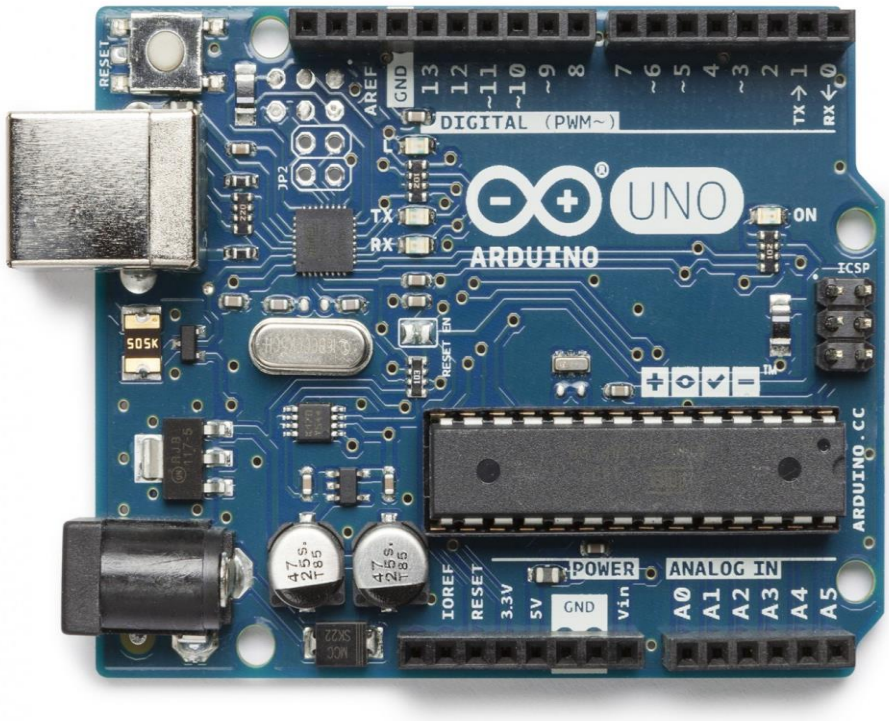
13) You will observe that whatever data will transmitted by transmitter will received on terminal window through receiver.

NOTE-

- **Here we need two Raspberry Pi if in case two Raspberry Pi are not available then you can also use another microcontroller board**
  e.g. Arduino

- **We need 2 ZigBee modules for transmit and receive data. And 1st we have to configure those modules as a transmitter and receiver using XCTU software.**
  For ZigBee configuration refer
  http://www.libelium.com/development/waspmote/documentation/x-ctu-tutorial/
  https://alselectro.wordpress.com/category/xbee-radios/

INTRODUCTION TO ARDUINO UNO



Arduino uno

The **Arduino Uno** is a microcontroller board based on the ATmega328. Arduino is an open-source, prototyping platform and its simplicity makes it ideal for hobbyists to use as well as professionals. The Arduino Uno has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It 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.

Power :-

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector. The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

VIN-

The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

5V-

This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.

3V3-

A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.

GND- Ground pins.

Memory-

The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

Input and Output-

Each of the 14 digital pins on the Uno can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

Serial: 0 (RX) and 1 (TX)-

Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.

External Interrupts: 2 and 3-

These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.

PWM-

3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analogWrite() function.

SPI-

10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.

LED: 13-

There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analog Reference() function. Additionally, some pins have specialized functionality:

TWI: A4 or SDA pin and A5 or SCL pin- Support TWI communication using the Wire library.

There are a couple of other pins on the board:

AREF-

Reference voltage for the analog inputs. Used with analogReference().

Reset-

Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

SPECIFICATION

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- 32KB (ATmega328) of which 0.5 KB used by boot loader

SRAM- 2 KB (ATmega328)

EEPROM- 1 KB (ATmega328)

Clock Speed- 16 MHz

SETTING FOR Arduino Uno :

INSTALL Arduino 1.8.3 -

The open source Arduino software(IDE) makes it easy to write code and upload it to the board. It runs on Windows, MAC os x, Linux. This software can be used with any Arduino board.

1) **DOWNLOAD-**

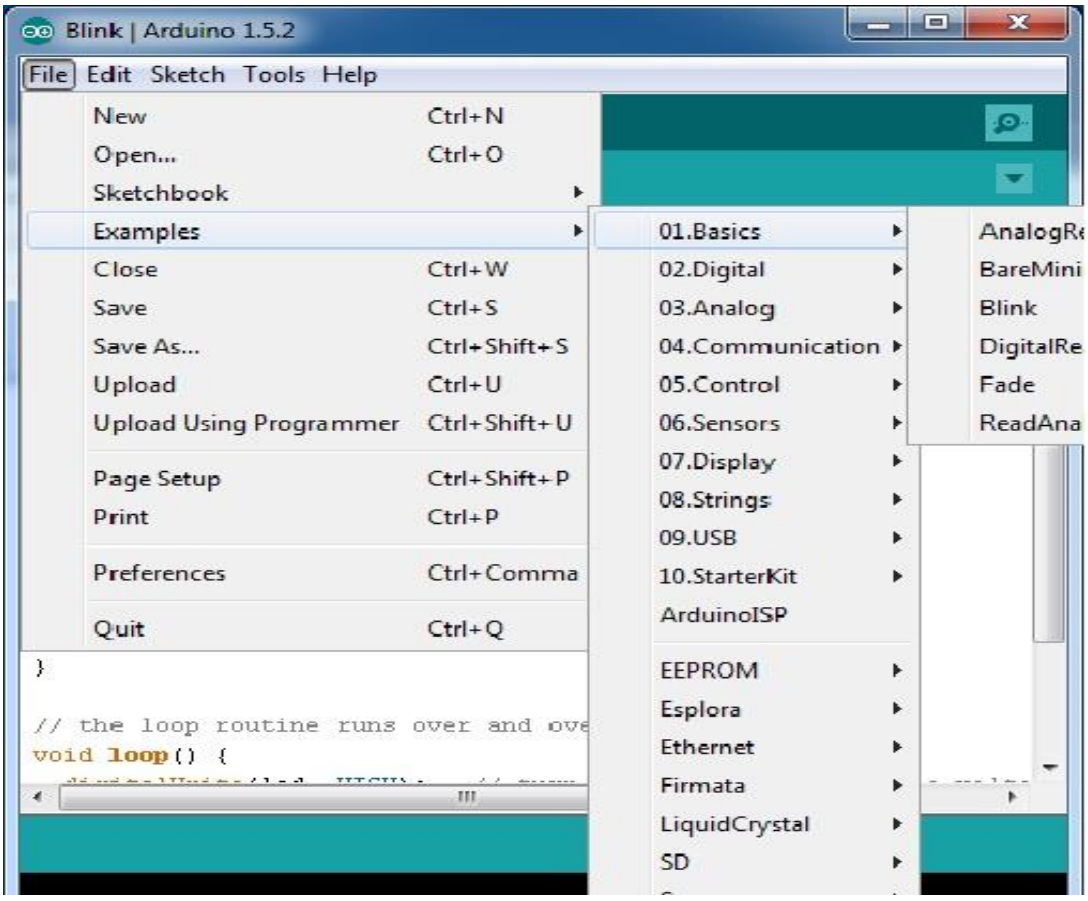
Download the software from-

https://www.arduino.cc/en/Main/Software

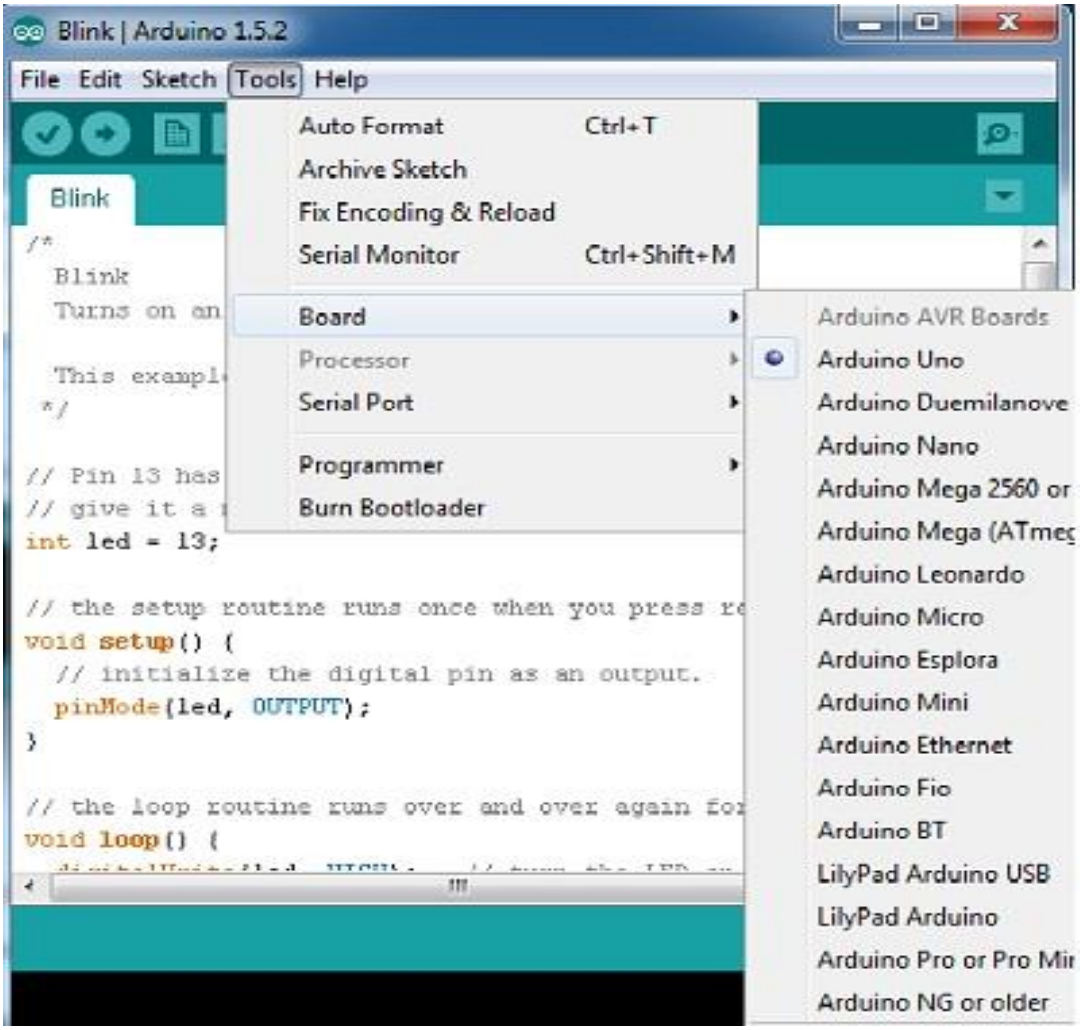
Launch and Blink-

After following the appropriate steps for your software install, we are now ready to test your first program with your Arduino board!

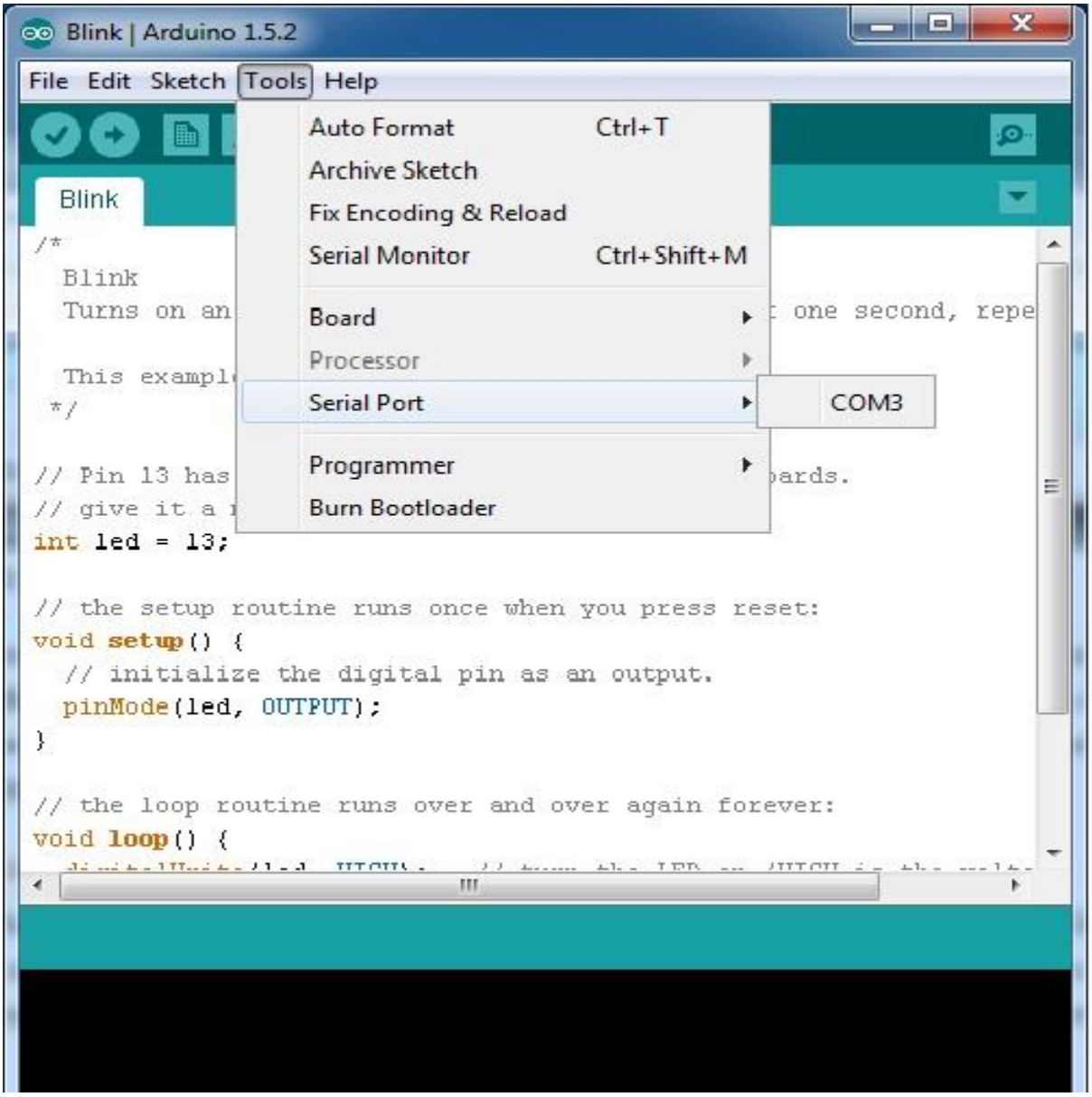
- Launch the Arduino application
- If you disconnected your board, plug it back in
- Open the Blink example sketch by going to: File > Examples > 1.Basics > Blink



- Select the type of Arduino board you're using: Tools > Board > your board type

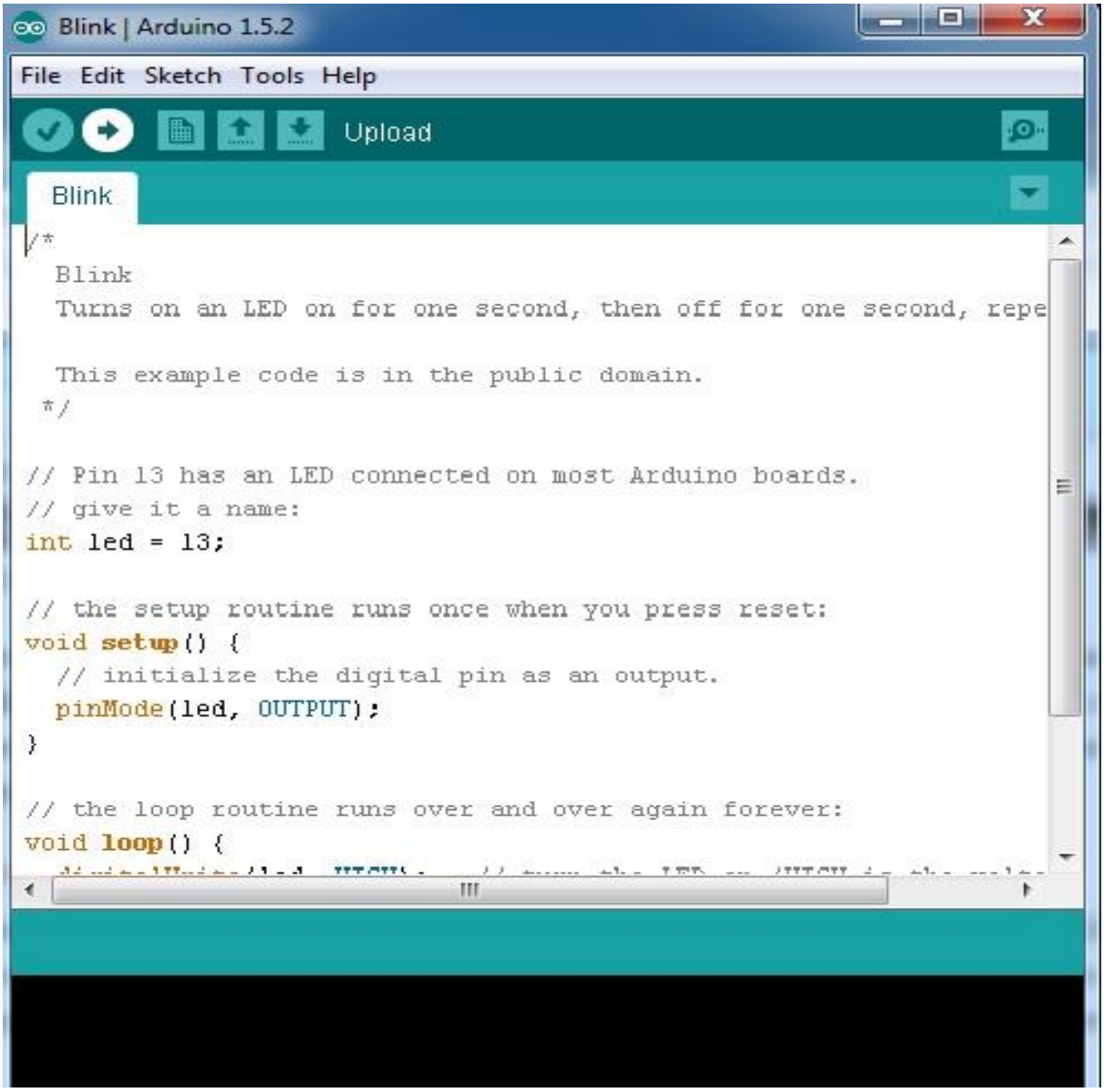


- Select the serial/COM port that your Arduino is attached to: Tools > Port > COMxx



- If you're not sure which serial device is your Arduino, take a look at the available ports, then unplug your Arduino and look again. The one that disappeared is your Arduino.

- With your Arduino board connected, and the Blink sketch open, press the ‘Upload’ button



- After a second, you should see some LEDs flashing on your Arduino, followed by the message ‘Done Uploading’ in the status bar of the Blink sketch.
- If everything worked, the onboard LED on your Arduino should now be blinking! You just programmed your first Arduino!

EXPERIMENT No 1-

LED Logic

AIM: To study the interfacing of LED and SWITCH to the Arduino.

REQUIREMENTS: Arduino, 26 pin FRC cable, 5v, 2A power supply and USB cable.

PROCEDURE:

12) Open the Arduino IDE. Connect USB cable to Arduino Uno.

13) Select Arduino Uno and COM port from **TOOL** section.

14) Now go to file ► open particular "**.ino**" file of led logic program.

15) Compile the program and then upload it.

16) After uploading connect 26 pin FRC cable from the board to "XBEE-LED-SWITCHES" section.

17) You will observe that led's will **ON** when you **ON** the switches and **OFF** when you **OFF** the switches.

Note: FRC port connection must be proper to run the program.

EXPERIMENT No 2.

LCD DISPLAY

AIM: To study the interfacing of LCD(16*2) with the Arduino Uno.

REQUIREMENTS: Arduino, 26 pin FRC cable, 5v,2A power supply or USB cable.

PROCEDURE:

1) Open the Arduino IDE. Connect USB cable to Arduino Uno.
2) Select Arduino Uno and COM port from **TOOL** section.
3) Now go to file ⟶ open particular "**.ino**" file of LCD program.
4) Compile the program and then upload it.
5) After uploading connect 26 pin FRC cable from the board to "LCD-KEYPAD" section.
6) You will observe that it displays "**LOGSUN SYSTEMS**" on LCD display.

THEORY:

Liquid crystal display:

LCD can be connected to the Raspberry pi through the GPIO pins. LCD is connected in the 4-bit mode or 8-bit mode. And the standard subroutine is given with the development board. So that the application can be easily demonstrated and also for further implementation the subroutine can be easily embedded for which one has to do very few changes. Wide range of instruction functions: Clear displays, cursor home, display ON/OFF, cursor ON/OFF, cursor shift, display shift.

4-bit mode

RS, RW, EN are 1&2 are short.

8-bit mode

RS, RW, EN are 2&3 are short

EXPERIMENT No 3.

LCD WITH KEYPAD INTERFACING

AIM: To study the interfacing of LCD and Keypad with Arduino.

REQUIREMENTS: Arduino, 26 pin FRC cable, 5v, 2A power supply or USB cable.

PROCEDURE:

1) Open the Arduino IDE. Connect USB cable to Arduino Uno.
2) Select Arduino Uno and COM port from **TOOL** section.
3) Now go to file ⟶ open particular "**.ino**" file of LCD Keypad program.
4) Compile the program and then upload it.
5) After uploading connect 26 pin FRC cable from the board to "LCD-KEYPAD" section.
6) You will observe that when you press a key (0-F) it will show on LCD display.

EXPERIMENT No 4.
Traffic Light Signal

AIM: To study the interfacing of LED to make traffic light signal with Arduino

REQUIREMENTS: Arduino, 26 pin FRC cable, 5v,2A power supply or USB cable.

PROCEDURE:

1) Open the Arduino IDE. Connect USB cable to Arduino Uno.
2) Select Arduino Uno and COM port from **TOOL** section.
3) Now go to file ⟶ open particular "**.ino**" file of traffic light signal program.
4) Compile the program and then upload it.
5) After uploading connect 26 pin FRC cable from the board to "TRAFFIC LIGHT" section.

12) You will observe that red led's of 1st and 3rd stage is ON for 10 sec. Green led's of both opposite sides are on for 5 sec. Then yellow led's 5 times blinks. And red led's OFF. And same action happens for section 2nd and 4th.

EXPERIMENT No 5.

INTERFACING IR SENSOR WITH ARDUINO

AIM: To study the interfacing of IR sensor with Arduino.

REQUIREMENTS:- Arduino, 26 pin FRC cable, 5v,2A power supply or USB cable.

PROCEDURE:

1) Open the Arduino IDE. Connect USB cable to Arduino Uno.

2) Select Arduino Uno and COM port from **TOOL** section.

3) Now go to file ⟶ open particular "**.ino**" file of IR Sensor program.

4) Compile the program and then upload it.

5) After uploading connect 26 pin FRC cable from the board to "ADC-DAC-TEMP-IR" section.

12) You will observe that when obstacle is detected that time "IR is detected" prints on the serial monitor, relay will activated and red led is ON.

Experiment No 6.

INTERFACING DC MOTOR TO ARDUINO

AIM: To study the interfacing of dc motor to Arduino.

REQUIREMENTS:- Arduino, 26 pin FRC cable, 5v,2A power supply or USB cable, Dc motor, 12v dc external power supply.

PROCEDURE:

1) Open the Arduino IDE. Connect USB cable to Arduino Uno.
2) Select Arduino Uno and COM port from **TOOL** section.
3) Now go to file ⟶ open particular "**.ino**" file of DC motor program.
4) Compile the program and then upload it.
5) After uploading connect 26 pin FRC cable from the board to "DC-STEPPER MOTOR" section.
6) You will observe that when you press the switch **START** motor will start rotating in forward direction. And when you press the switch **REV** motor will rotate in opposite or reverse direction
7) When you press switch **INC** the speed of motor increase slowly, as you press the same switch again. And when you press switch **DEC** the speed of motor decrease slowly, as you press same switch again.
8) When you press switch **STOP** the motor will stop rotating.

NOTE:- Connect both jumpers(JP2 & JP3) on DC side.

Experiment No 7.

INTERFACING STEPPER MOTOR TO ARDUINO

AIM: To study the interfacing of stepper motor to Arduino.

REQUIREMENTS: Arduino, 26 pin FRC cable, 5v,2A power supply or USB cable, Dc motor,12v dc external power supply.

PROCEDURE:

1) Open the Arduino IDE. Connect USB cable to Arduino Uno.
2) Select Arduino Uno and COM port from **TOOL** section.
3) Now go to file ⟶ open particular "**.ino**" file of Stepper motor program.
4) Compile the program and then upload it.
5) After uploading connect 26 pin FRC cable from the board to "DC-STEPPER MOTOR" section.
6) You will observe that when you press the switch **START** motor will start rotating in forward direction. And when you press the switch **REV** motor will rotate in opposite or reverse direction
7) When you press switch **INC** the speed of motor increase slowly, as you press the same switch again. And when you press switch **DEC** the speed of motor decrease slowly, as you press same switch again.
8) When you press switch **STOP** the motor will stop rotating.

NOTE:- Connect both jumpers(JP2 &JP3) on STP side.

Experiment No 8.

INTERFACING RELAY WITH ARDUINO

AIM: To study interfacing of relay with Arduino

REQUIREMENTS:- Arduino, 26 pin FRC cable, 5v,2A power supply or USB cable.

PROCEDURE:

1) Open the Arduino IDE. Connect USB cable to Arduino Uno.
2) Select Arduino Uno and COM port from **TOOL** section.
3) Now go to file ⟶ open particular "**.ino**" file of relay program.
4) Compile the program and then upload it.
5) After uploading connect 26 pin FRC cable from the board to "ADC-DAC-TEMP-IR" section.

12) You will observe that relay is ON for 5sec and OFF for 5 sec.

Experiment No 9.

LIFT ELEVATOR SIMULATOR

AIM: To study Lift Elevator Simulator.

REQUIREMENTS:- Arduino, 26 pin FRC cable, 5v,2A power supply or USB cable.

PROCEDURE:

1) Open the Arduino IDE. Connect USB cable to Arduino Uno.
2) Select Arduino Uno and COM port from **TOOL** section.
3) Now go to file ⟶ open particular "**.ino**" file of Lift Elevator Simulator program.
4) Compile the program and then upload it.
5) After uploading connect 26 pin FRC cable from the board to "LIFT ELEVATOR SIMULATOR" section.
1) You will observe that when you press the switch **Ground** the **zero** will display on 7-segment display and green led is on. And when you press the switch **Top** the **one** will display on 2nd display and yellow led is high, then after some delay **two** display on 3rd display and yellow led is high. Then again after some delay **three** display on 4th display and green led is high.
2) The above procedure is same for 2nd floor and 1st floor.

NOTE- Connect jumper to VCC or GND as per requirement for pull up or pull down the switches.

Experiment No 10.

INTERFACING LM-35 WITH ARDUINO

AIM: Interface LM-35 to Arduino.

REQUIREMENTS:- Arduino,26 pin FRC cable, 5v,2A power supply, LM-35(Temperature sensor).

PROCEDURE:

1) Open the Arduino IDE. Connect USB cable to Arduino Uno.
2) Select Arduino Uno and COM port from **TOOL** section.
3) Now go to file ⟶ open particular "**.ino**" file of LM-35 program.
4) Compile the program and then upload it.
5) After uploading keep JP8 jumper on LM-35 side.
6) Now connect 26 pin FRC cable from the board to "ADC-DAC-TEMP-IR" section.
7) You will observe that when resistor is heated LM-35 sense the temperature and the output will shows on serial monitor.
8) Also you can see the output on LED's by varying the pot.

NOTE- The LM-35 sensor gives the Analog output. So we use the ADC of arduino to convert Analog output into Digital. You can also use a POT as a analog input and shows the output on led's for study of ADC with Arduino.

Experiment No 11.

INTERFACING ZIGBEE WITH ARDUINO

AIM: ZigBee interface to Arduino.

REQUIREMENTS: Arduino, 26 pin FRC cable, 5v,2A power supply or USB cable ZigBee modules.

PROCEDURE:

14) Open the Arduino IDE. Connect USB cable to Arduino Uno.

15) Select Arduino Uno and COM port from **TOOL** section.

16) Now go to file→ open particular "**.ino**" file of Lift Elevator Simulator program.

17) Configure ZigBee modules first as transmitter and second as a receiver. Configuration procedure will show the link given below.

18) For this experiment we need two Arduino uno board.

19) Upload **receiver.ino** program to onboard Arduino and **transmitter.ino** to the other Arduino.

20) Now open serial monitor window in Arduino software on pc.

21) Connect 26 pin FRC cable from the board to "XBEE-LED-SWITCHES" section.

22) Whatever data will transmitted by transmitter will received on serial monitor through receiver.

NOTE- We need 2 ZigBee modules for transmit and receive data. And 1st we have to configure that modules as a transmitter and receiver using XCTU software. For

ZigBee configuration refer
http://www.libelium.com/development/waspmote/documentation/x-ctu-tutorial/
https://alselectro.wordpress.com/category/xbee-radios/

7. FRC Connector Details

A) 26 PIN FRC CONNECTOR DETAILS RASPBERRY PI (J7) :

PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	SCL	2	SDA
3	GP27	4	GP22
5	GP16	6	GP10
7	GP12	8	GP20
9	GP7	10	GP21
11	GP8	12	GP26
13	GP25	14	GP19
15	GP24	15	GP13
17	GP23	18	GP6
19	TX	20	GP5
21	RX	22	GP11
23	GP18	24	GP9
25	GND	26	5V

B) 26 PIN FRC CONNECTOR DETAILS ARDUINO UNO (J8):

PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	A5	2	A4
3	D10	4	D11
5	D9	6	D12
7	D8	8	D13
9	D7	10	A0
11	D6	12	A1
13	D5	14	A2
15	D4	15	A3
17	D3	18	

19	D1 TX	20	
21	D0 RX	22	
23	D2	24	
25	GND	26	5V