# PROJECT REPORT
# ON
# DATA TRANSMISSION WITH RASPBERRY PI

Submitted by : Atul Kumar Agrawal
Branch - Computer Science and Engineering
Veer Surendra Sai University of Technology
Burla, Sambalpur
Application id: 103

# CERTIFICATE

This is to certify that this project on "**DATA TRANSMISSION WITH RASPBERRY PI**" is submitted by **Atul Kumar Agrawal** working as trainee in Directorate Central Data Processing Division, Integrated Test Range, Chandipur (DRDO).
During this period he worked under my guidance and supervision. He has successfully completed the above project assigned by me. During his worked period, he was sincere and showed keen interest in doing his project, thus completing it in stipulated time. No classified data was provided to him of any kind of implement the work.

I wish him all success in his future career.

Shri Shariq Ali
Scientist 'D'
(Guide)

Shri Rakesh Barua
Scientist 'F'
Group Director (CDP)

# ACKNOWLEDGEMENT

I am grateful to many people who have given their time and guidance in completing this project. I am thankful to HRD group for selecting and placing me to directorate of Computer & Data Processing, which is having best computer facilities. I am very grateful to **Mr. Rakesh Barua** (Sc F), GD (CDP) for assigning me to Missile Management System and giving me encouragement and guidance
I am very thankful to **Mr. Shariq Ali.** (Sc. D) for assigning me an interesting project and giving me support and technical guidance .I am also indebted to all employees of Directorate of Computer &Data Processing for their generosity in helping me. I am also grateful to officer-in-charge Knowledge Center for permitting me to use rich library facilities.

**Atul Kumar Agrawal**

# Table of Contents

# Introduction

## About DRDO

The Defence Research and Development Organisation(DRDO) is an agency of the Government of India, charged with the military's research and development, headquartered in New Delhi, India. It was formed in 1958 by the merger of the Technical Development Establishment and the Directorate of Technical Development and Production with the Defence Science Organisation. It is under the administrative control of the Ministry of Defence, Government of India.

With a network of 52 laboratories, which are engaged in developing defence technologies covering various fields, like aeronautics, armaments, electronics, land combat engineering, life sciences, materials, missiles, and naval systems, DRDO is India's largest and most diverse research organisation. The organisation includes around 5,000 scientists belonging to the Defence Research & Development Service(DRDS) and about 25,000 other scientific, technical and supporting personnel.

The sphere of activity of DRDO is very vast and it includes all branches of engineering, science & technology. Its contribution in the field of nuclear science and technology is well known. Under the guidance of great luminary Dr. A.P.J Abdul Kalam, It has achieved new heights.

## About ITR:

The Interim Test Range at Chandipur, Odisha is a world class test range which has a facility of 4 launch complexes and a well communicated set-up of departments which play a major role in testing of ranges of the various ballistic as well as cruise missiles. The Island at Dhamra (LC-4) is a naturally gifted island which can be used for the testing of very long range ICBMs because of its vast stretch in the Bay of Bengal. The location of Nilgiri hills provide a natural advantage for a longer tracking range of

RADARs installed at a height of 75 ft. It provides reliable and tested data to its users which are basically the services Army, Navy and the Air Force which come here for testing their missiles before inductions The various departments which work in unison at the ITR are:

a.) Control Centre: This includes the computer data processing, the telecommand.

b.) MET Center: This includes the meteorology department which works on the monitoring of weather and climatic parameters.

c.) RADAR Centre: This includes a hall for the conduction of orientation and a well equipped library for the incoming interns and apprentices.

d.) Communication Center:This center provides communication system.

e.) Telemetry center: For monitoring of on board parameters.

f.) Power Supply Division: For uninterrupted and quality power supply and monitoring during mission and non mission time periods.

g.) CAN Center: For monitoring of the parameters related to the Campus Area Network and work for the improvement in the infrastructure related to the network.

# PROJECT

## Objective of Project

The objective of the project is design to a client server model in which the server sends time to client , Server is sending client the data hour min sec millisec over udp socket @ 1s update rate and is received by the client side and is displayed.

## Description of Project

There are 2 programs :

1. Client Side

2. Server side

Server keeps its port open to serve the client whenever gets a request.The Server then sends the real time clock to the client which is displayed at the client side using a 7 segment LED.

### Hardware Specifications:

1.Raspberry Pi 3 model B Specifications

SoC: Broadcom BCM2837

CPU: 4× ARM Cortex-A53, 1.2GHz

GPU: Broadcom VideoCore IV

RAM: 1GB LPDDR2 (900 MHz)

Networking: 10/100 Ethernet, 2.4GHz 802.11n wireless

Bluetooth: Bluetooth 4.1 Classic, Bluetooth Low Energy

Storage: microSD

GPIO: 40-pin header, populated

Ports: HDMI, 3.5mm analogue audio-video jack, 4× USB 2.0, Ethernet, Camera Serial Interface (CSI), Display Serial Interface (DSI)

2.Push button

3. 333 Ohm Resistor

4. LED 5V 3 nos. - 1.Red 2.Green 3.Yellow

5. Breadboard

6. 2* 4 bit 7 segment LED for Real Time Display

7. Jumper wires

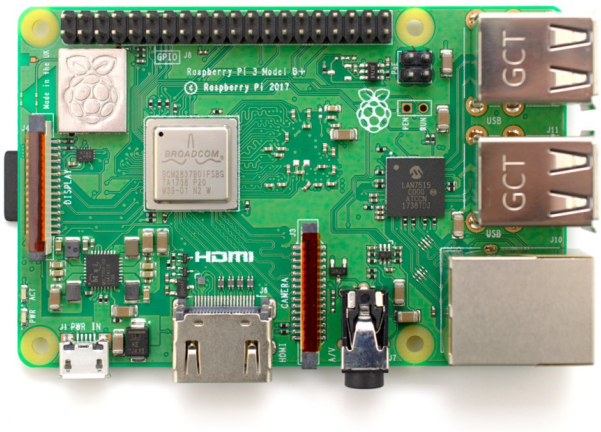
8. Ethernet Cables

9. Power supply

**Software Specification:**

Operating System:Raspbian OS based on debain build of Linux Kernel
Programming Languages: Python3
Putty/VNC viewer

## Raspberry Pi



It is a small and affordable computer that you can be used to learn programming.

The Raspberry Pi launched in 2012, and there have been several iterations and variations released since then. The original Pi had a single-core 700MHz CPU and just 256MB RAM, and the latest model has a quad-core 1.4GHz CPU with 1GB RAM. The main price point for Raspberry Pi has always been $35 and all models have been $35 or less, including the Pi Zero, which costs just $5.

All over the world, people use Raspberry Pis to learn programming skills, build hardware projects, do home automation, and even use them in industrial applications.

The Raspberry Pi is a very cheap computer that runs Linux, but it also provides a set of GPIO (general purpose input/output) pins that allow you to control electronic components for physical computing and explore the Internet of Things (IoT).

# LED

A light-emitting diode (LED) is a semiconductor light source that emits light when current flows through it. Electrons in the semiconductor recombine with electron holes, releasing energy in the form of photons. This effect is called electroluminescence.The color of the light (corresponding to the energy of the photons) is determined by the energy required for electrons to cross the band gap of the semiconductor.White light is obtained by using multiple semiconductors or a layer of light-emitting phosphor on the semiconductor device.

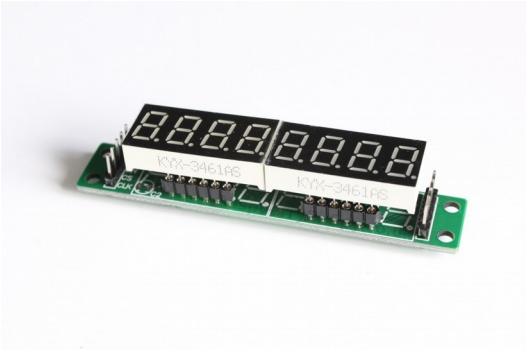
# 7 Segment LED Display

A seven-segment display (SSD), or seven-segment indicator, is a form of electronic display device for displaying decimal numerals that is an alternative to the more complex dot matrix displays.

Seven-segment displays are widely used in digital clocks, electronic meters, basic calculators, and other electronic devices that display numerical information.

4 bit type

8 bit type

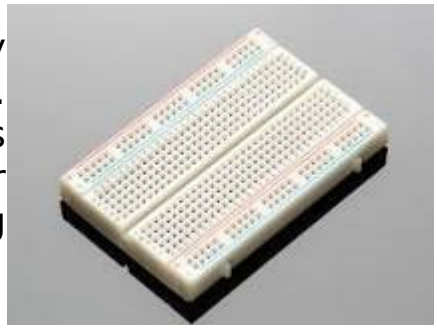


## RESISTOR:

A **resistor** is a passive two-terminal electrical component that implements electrical resistance as a circuit element. In electronic circuits, **resistors** are used to reduce current flow, adjust signal levels, to divide voltages, bias active elements, and terminate transmission lines, among other uses.

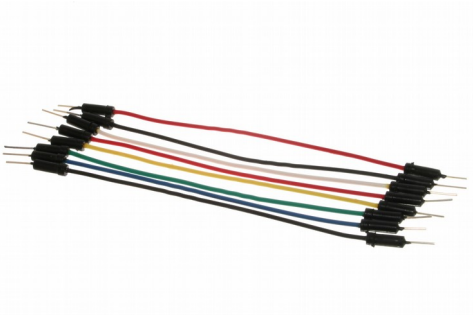
## BREADBOARD:

A **breadboard** is a solderless device for temporary prototype with electronics and test circuit designs. Most electronic components in electronic circuits can be interconnected by inserting their leads or terminals into the holes and then making connections through wires where appropriate.



## JUMPER WIRES:

**Jumper wires** are simply **wires** that have connector pins at each end, allowing them to be used to connect two points to each other without soldering. **Jumper wires** are typically used with breadboards and other prototyping tools in order to make it easy to change a circuit as needed.



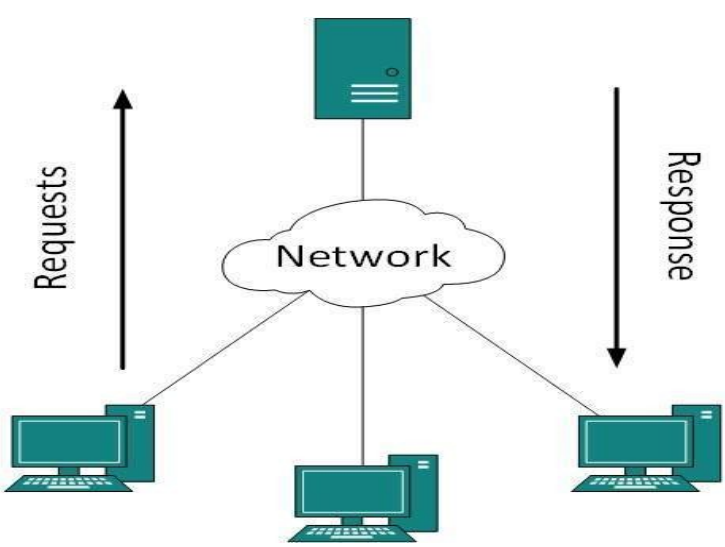
## SWITCH :

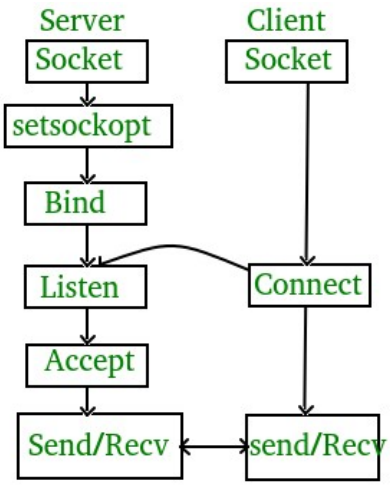
A simple switch mechanism for controlling some aspect of a machine or a process. Buttons are typically made out of hard material, usually plastic or metal. The surface is usually flat or shaped to accommodate the human finger or hand, so as to be easily depressed or pushed.

# Client and Server model

In client-server model, any process can act as Server or Client. It is not the type of machine, size of the machine, or its computing power which makes it server; it is the ability of serving request that makes a machine a server.

An application program is known as a client program, running on the local machine that requests for a service from an application program known as a server program, running on the remote machine.

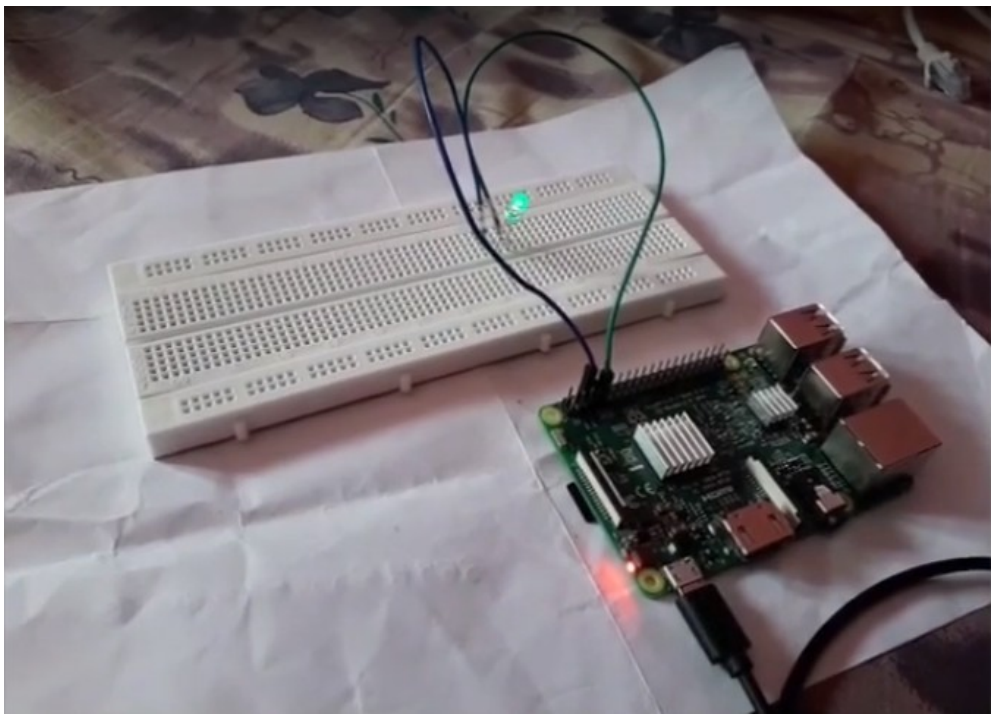



A client program runs only when it requests for a service from the server while the server program runs all time as it does not know when its service is required.

A server provides a service for many clients notjust for a single client. Therefore, we can say that client-server follows the many-to-one relationship. Many clients can use the service of one server.

# 1. Blinking of LED with Server Client Programming



To blink a LED through raspberry pi GPIO Pins on server side, when client sends data to the client.

**Hardware Requirements:**

1) LED
2) Breadboard
3) Raspberry Pi

**Server Side Program**

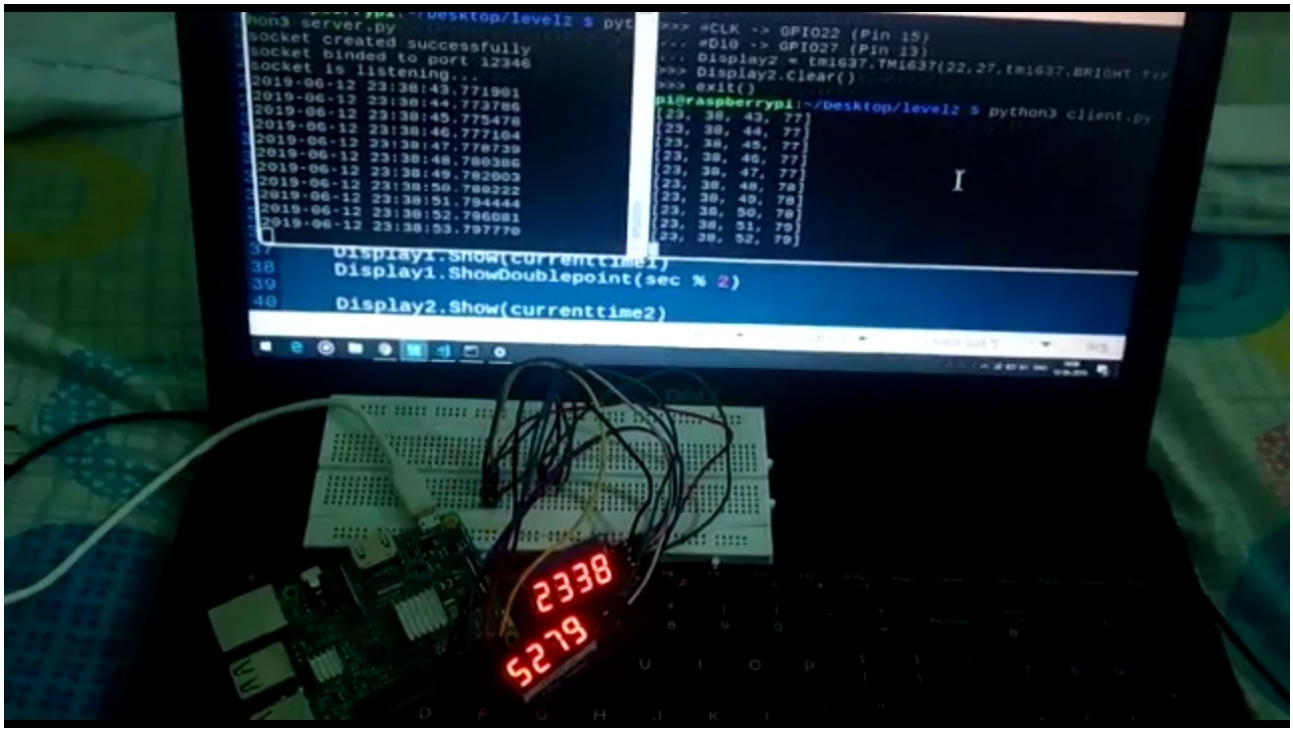
```
import socket
import RPi.GPIO as gp
import time
#setup
gp.setmode(gp.BOARD)
gp.setup(7,gp.OUT)
port = 12345

with socket.socket() as s:
     print("socket created successfully")
     port = 12346
     s.bind(('',port))
```

```
     print("socket binded to port %s"%(port))
     s.listen(5)
     print ("socket is listening...")
     c, addr = s.accept()
     while True:
          data = c.recv(6)
          if ('EOF' in data.decode('utf-8')):
               break
          print(data.decode('utf-8'))
          gp.output(7, True)
          time.sleep(1)
          gp.output(7, False)
          time.sleep(2)
gp.cleanup()
```

**Cient Side Program**

```
import socket
with socket.socket() as c:
     c.connect(('', 12346))
     with open('data.txt', 'r') as lines:
          for line in lines:
               c.sendall(line.encode('utf-8'))
          c.sendall('EOF'.encode('utf-8'))
```

# 2. Displaying of Time sent by server on Client Side




To display time using 2 x 4 digit 7 segment display through raspberry pi on client side. Each Digit on display has seven segments in it and each segment has one LED inside it to display the numbers by lightning up the corresponding segments.

**Hardware Requirements**

1. Raspberry Pi
2. 4 digit 7 Segment Display
3. Jumper wires (Female to Female)

Here, We used 2* 4 digits-7 segments LED display with TM1637 controller

**Connect your 4 digit 7 segment display with Raspberry Pi's GPIO Pins.**

| TM1637 Board Pin | Function | RPI Physical | Raspberry Function |
|---|---|---|---|
| **GND** | Ground | 14 | GND |
| **VCC** | + 5V Power | 4 | 5V |
| **DI0** | Data In | 18 | GPIO 24 |
| **CLK** | Clock | 16 | GPIO 24 |

**Server Side Program**

```
import socket
import datetime
from time import sleep

with socket.socket() as s:
  print("socket created successfully")
  port = 12346
  s.bind(('',port))
  print("socket binded to port %s"%(port))
  s.listen(1)
  print ("socket is listening...")
  c, addr = s.accept()
  while True:
    now = datetime.datetime.now()
    print(now)
              data   =   str(now.hour).zfill(2)   +   ':'   +
str(now.minute).zfill(2) + ':' + str(now.second).zfill(2) +
':' + str(now.microsecond // 1000).zfill(2)[:2]
    sleep(1)
    c.send(data.encode('utf-8'))
  s.close()
```

**Client Side Program**

```
import sys
import time
import datetime
import RPi.GPIO as GPIO
import tm1637

import socket
import json

#CLK -> GPIO23 (Pin 16)
#Di0 -> GPIO24 (Pin 18)
Display1 = tm1637.TM1637(23,24,tm1637.BRIGHT_TYPICAL)
Display1.Clear()
Display1.SetBrightnes(1)

#CLK -> GPIO22 (Pin 15)
#Di0 -> GPIO27 (Pin 13)
Display2 = tm1637.TM1637(22,27,tm1637.BRIGHT_TYPICAL)
Display2.Clear()
Display2.SetBrightnes(1)
```

```
with socket.socket() as s:
  s.connect(('localhost', 12346))
  while True:
    data = s.recv(11).decode('utf-8')
    unpacked_time = list(map(int, data.split(':')))
    hr = unpacked_time[0]
    mint = unpacked_time[1]
    sec = unpacked_time[2]
    ms = unpacked_time[3]
    print(unpacked_time[0:4])

    currenttime1=[int(hr/10),hr%10,int(mint/10),mint%10]

    currenttime2=[int(sec/10),sec%10,int(ms/10),(ms)%10]

    Display1.Show(currenttime1)
    Display1.ShowDoublepoint(sec % 2)

    Display2.Show(currenttime2)
```

## Script functions

The clock script uses the following functions, defined in tm1637.py:

**Display.Clear() -** Clears the display if individual LEDs are still active.

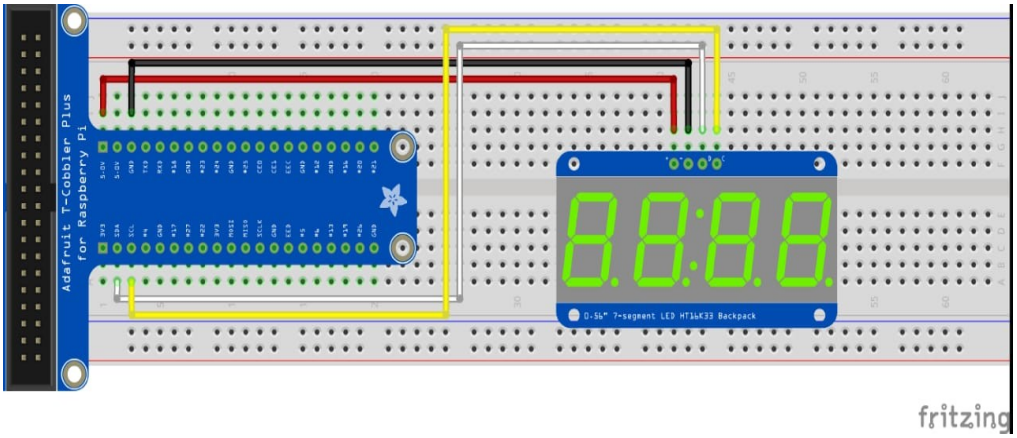
**Display.SetBrightness(x)** - After this you can adjust the brightness of the display, at least 0 and maximum 7.

**Display.Show(x,x,x,x)** - Show the actual 4 digits (digits), x can be 0 to 9.

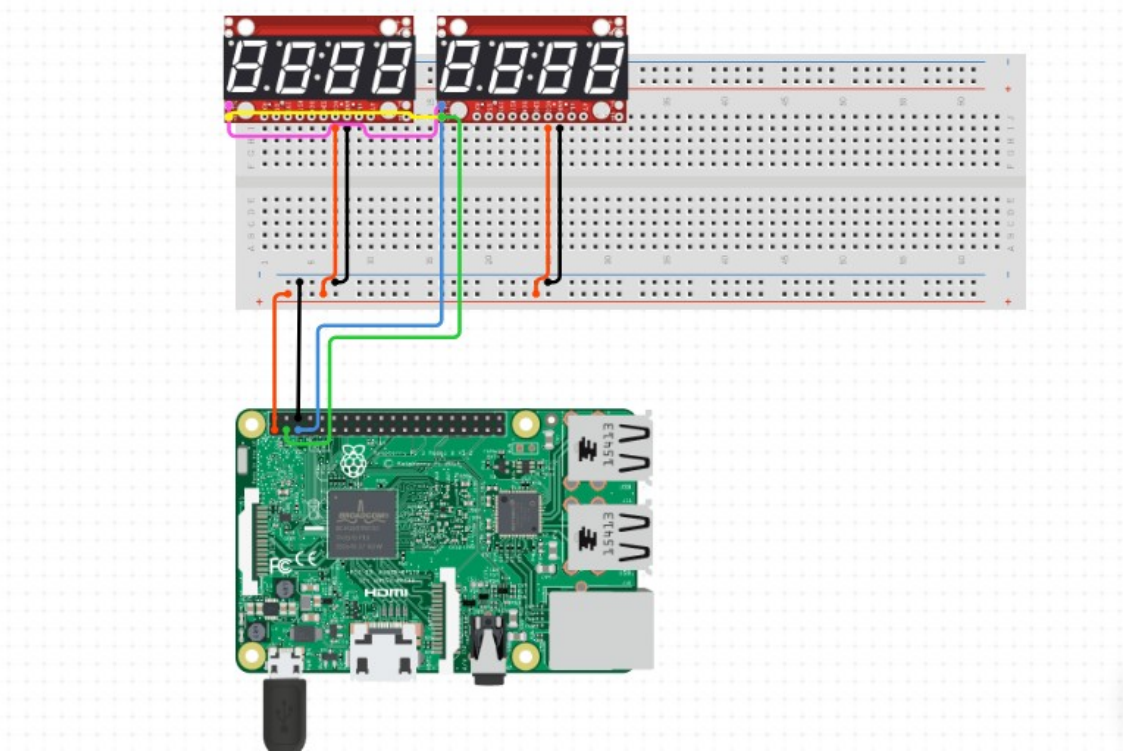
**Display.ShowDoublePoint(status)** - Controlling the ':' between the second and third digit, true(1) = on / false (0) = off.

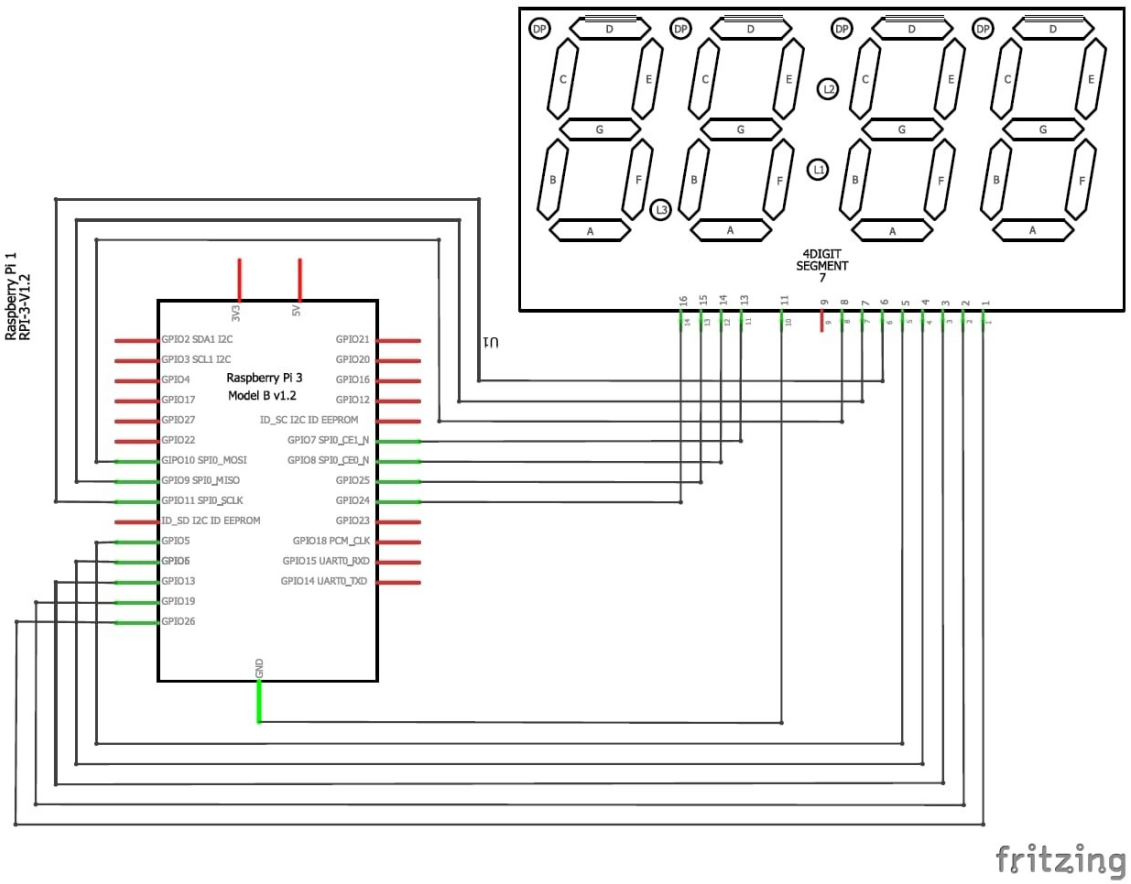
## CONNECTIONS:

### 4 Bit 7 Segment Display




### 8 Bit 7 Segment Display




DP
D
C
E
G
B
F
A
L1
L2
L3
4DIGIT
SEGMENT
7
16 15 14 13 11 9 8 7 6 5 4 3 2 1
Raspberry Pi 1
RPI-3-V1.2
3V3
5V
GPIO2 SDA1 I2C
GPIO3 SCL1 I2C
GPIO4
GPIO17
GPIO27
GPIO22
GIPO10 SPI0_MOSI
GPIO9 SPI0_MISO
GPIO11 SPI0_SCLK
ID_SD I2C ID EEPROM
GPIO5
GPIO6
GPIO13
GPIO19
GPIO26
Raspberry Pi 3
Model B v1.2
GPIO21
GPIO20
GPIO16
GPIO12
ID_SC I2C ID EEPROM
GPIO7 SPI0_CE1_N
GPIO8 SPI0_CE0_N
GPIO25
GPIO24
GPIO23
GPIO18 PCM_CLK
GPIO15 UART0_RXD
GPIO14 UART0_TXD
GND
U1
fritzing

# REFERENCES


http://www.microcontroller.it/english/Tutorials/Elettronica/componenti/TM1637.htm

https://www.raspberrypi.org/

https://www.youtube.com/

https://en.wikipedia.org/wiki/Raspberry_Pi