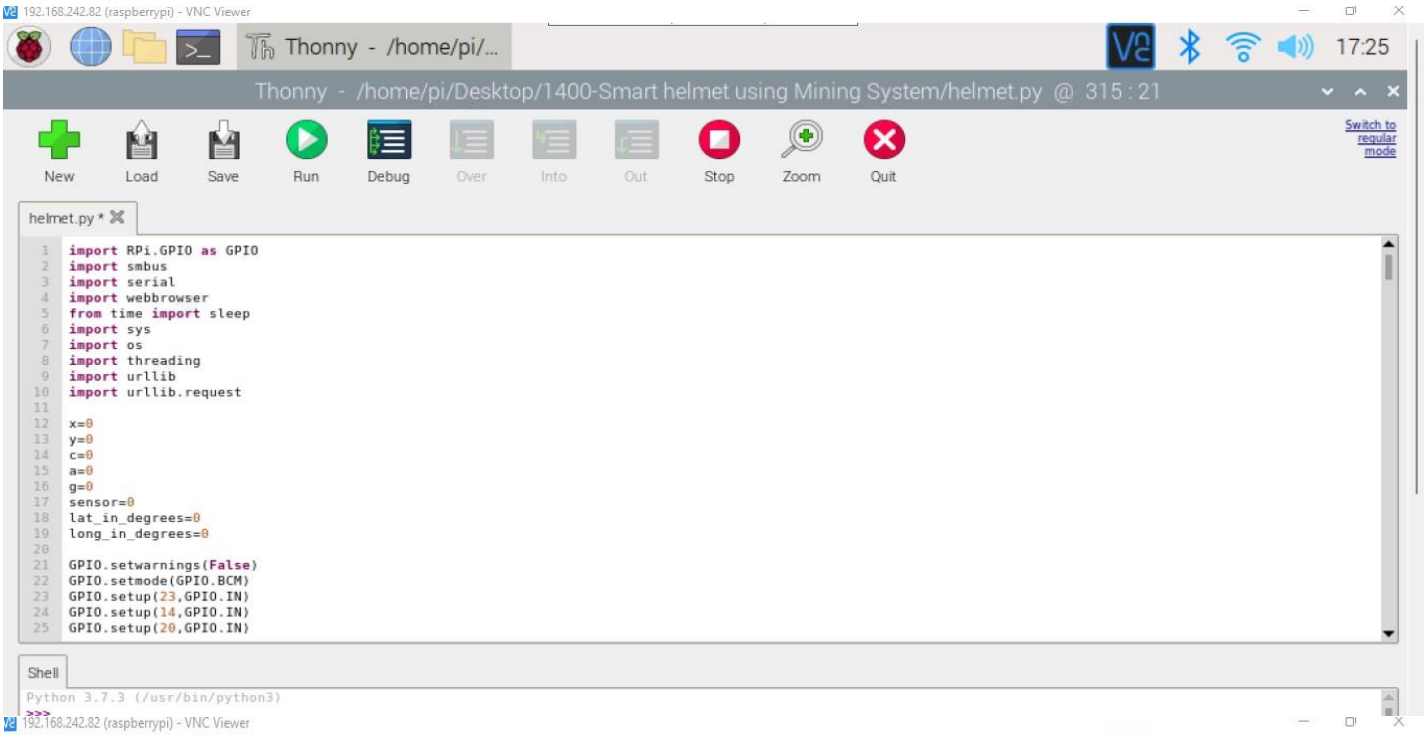
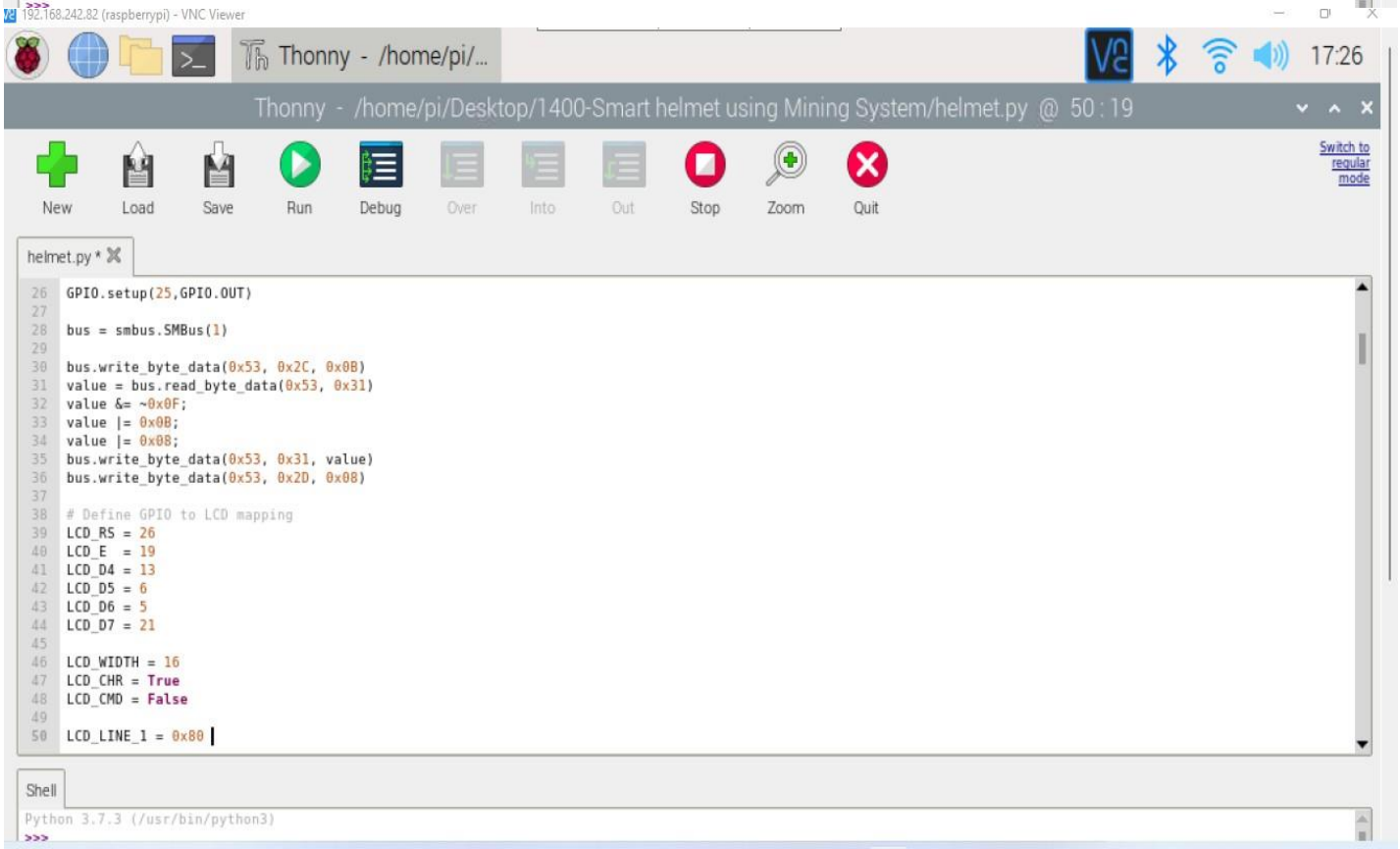
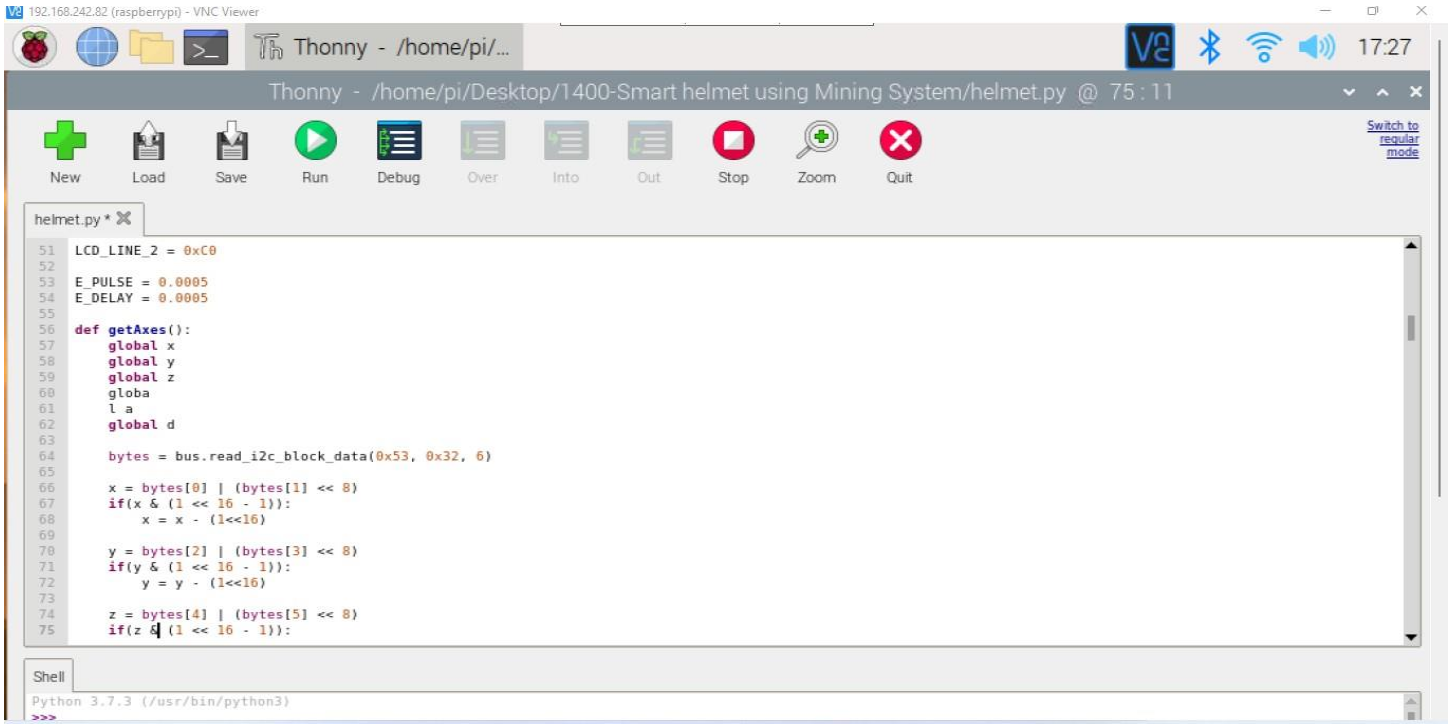
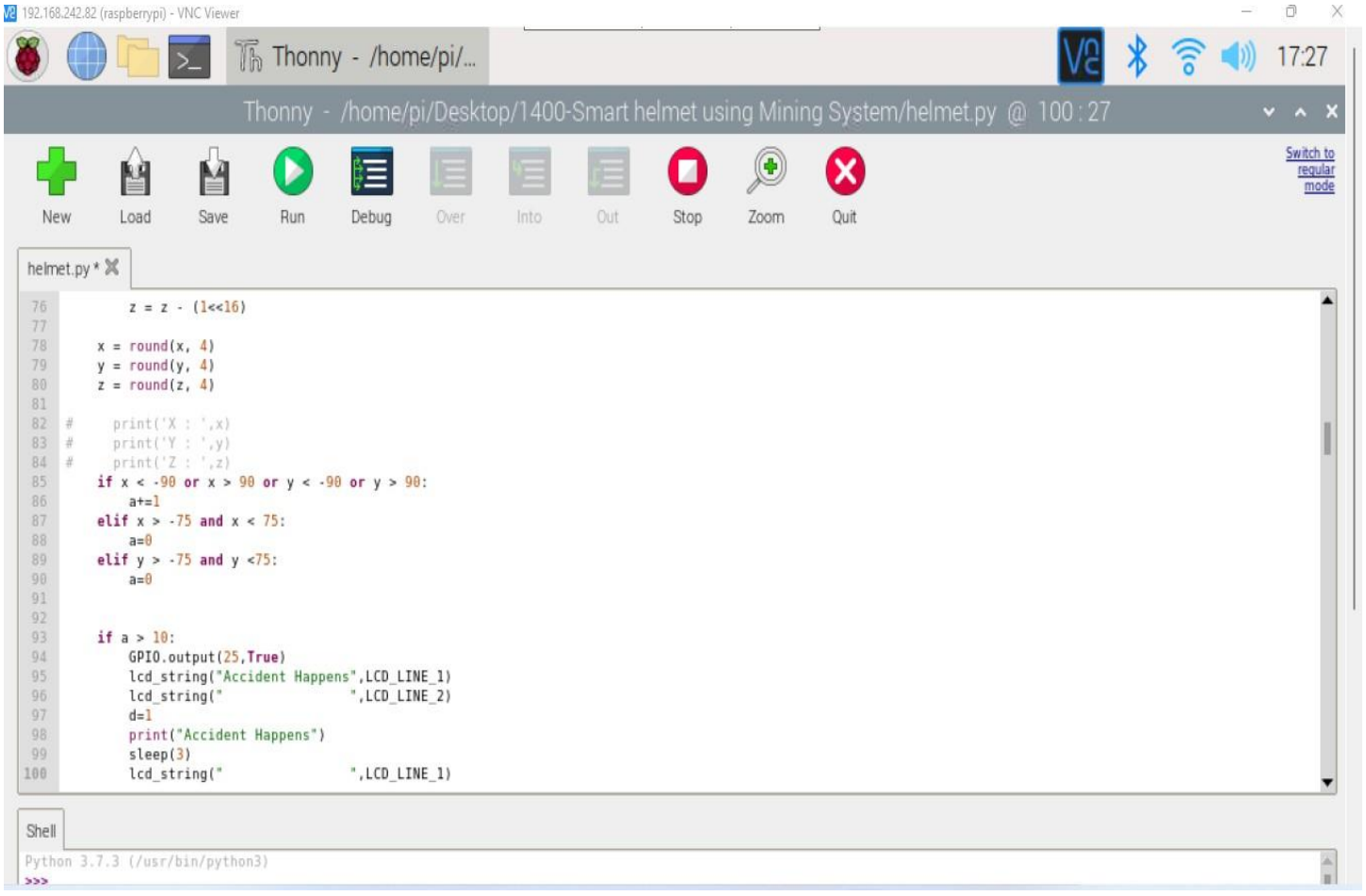
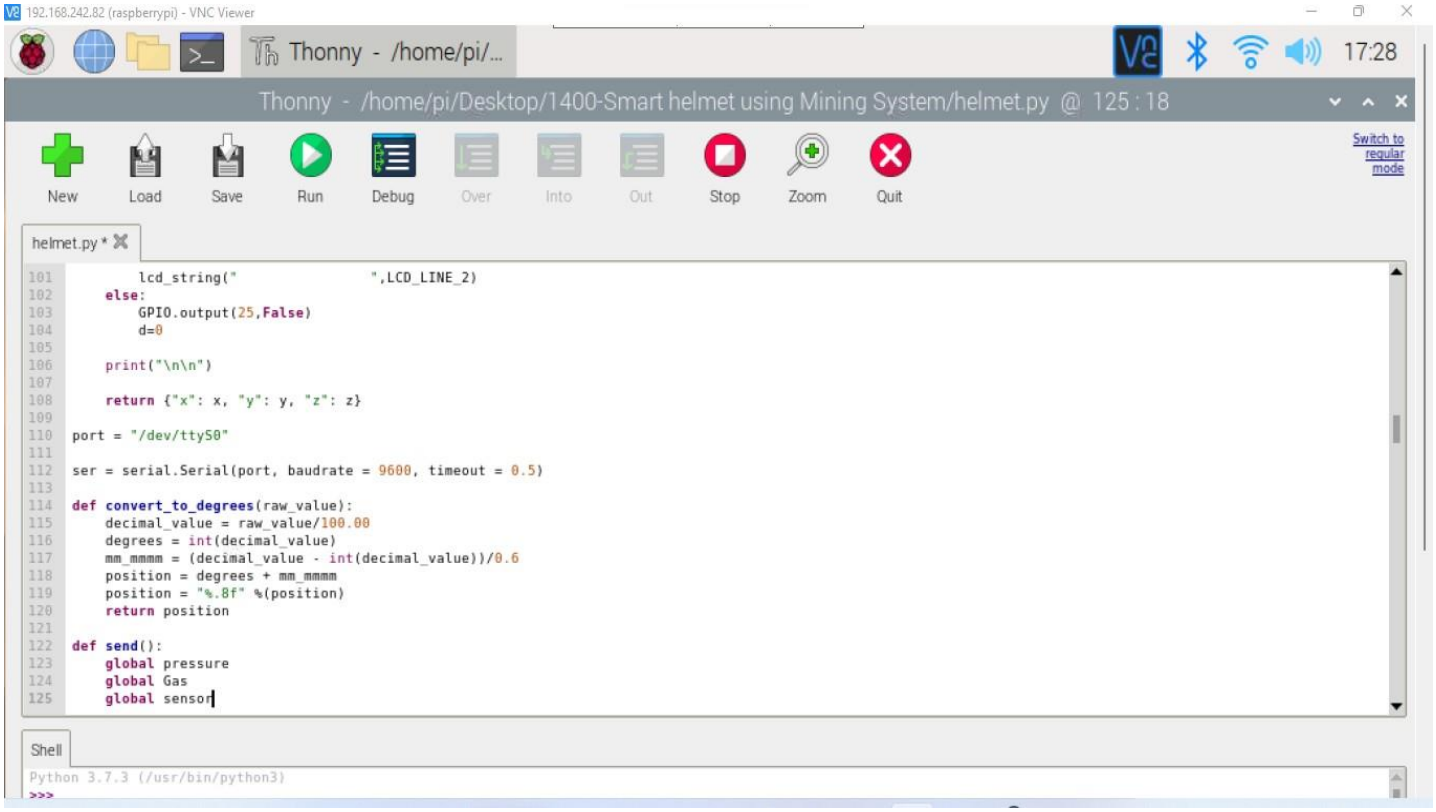
CODE

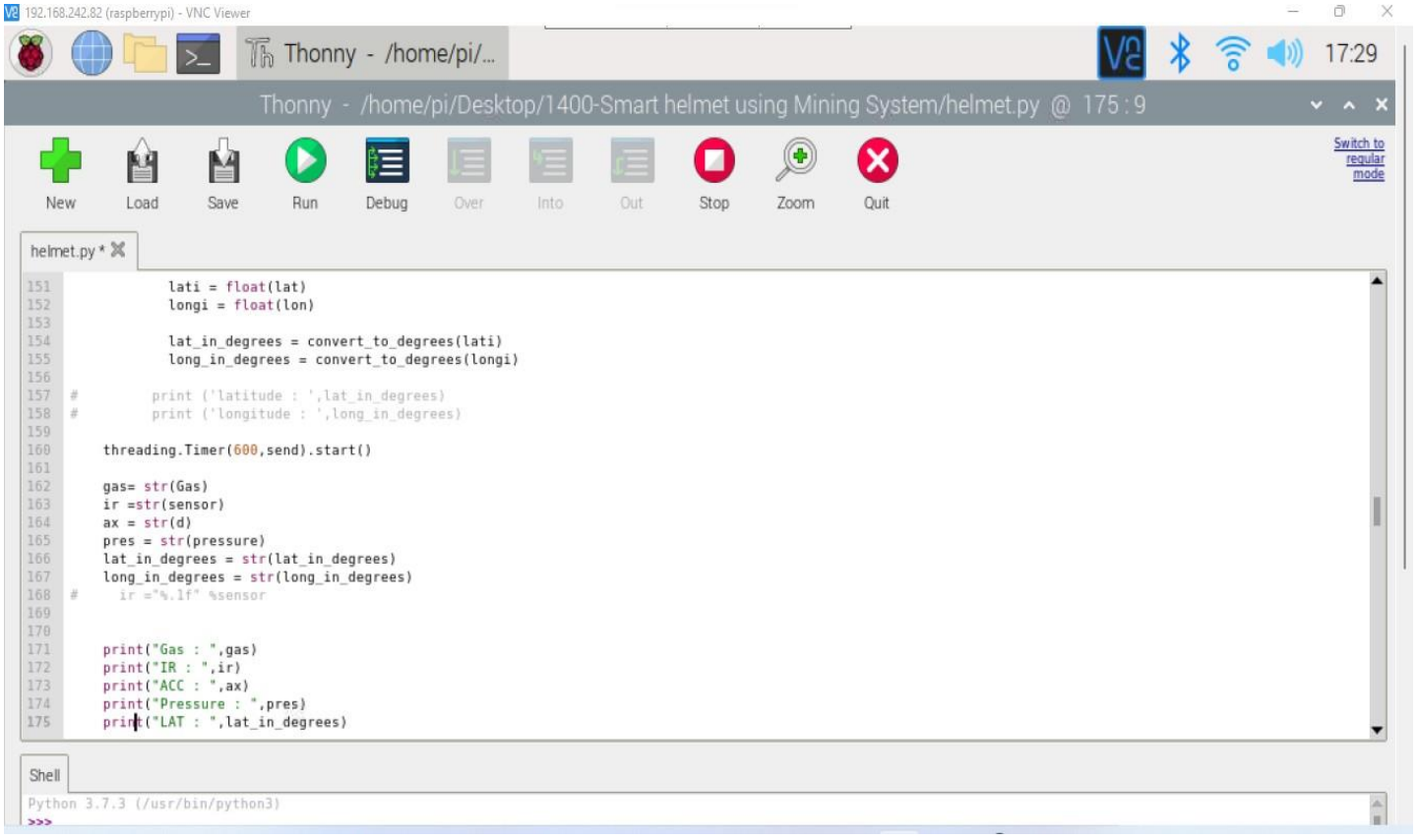

192.168.242.82 (raspberrypi) - VNC Viewer
Thonny - /home/pi/...
17:25
Thonny - /home/pi/Desktop/1400-Smart helmet using Mining System/helmet.py @ 315 : 21
New
Load
Save
Run
Debug
Over
Into
Out
Stop
Zoom
Quit
Switch to regular mode
helmet.py *
1 import RPi.GPIO as GPIO
2 import smbus
3 import serial
4 import webbrowser
5 from time import sleep
6 import sys
7 import os
8 import threading
9 import urllib
10 import urllib.request
11
12 x=0
13 y=0
14 c=0
15 a=0
16 g=0
17 sensor=0
18 lat_in_degrees=0
19 long_in_degrees=0
20
21 GPIO.setwarnings(False)
22 GPIO.setmode(GPIO.BCM)
23 GPIO.setup(23,GPIO.IN)
24 GPIO.setup(14,GPIO.IN)
25 GPIO.setup(20,GPIO.IN)
Shell
Python 3.7.3 (/usr/bin/python3)
>>>

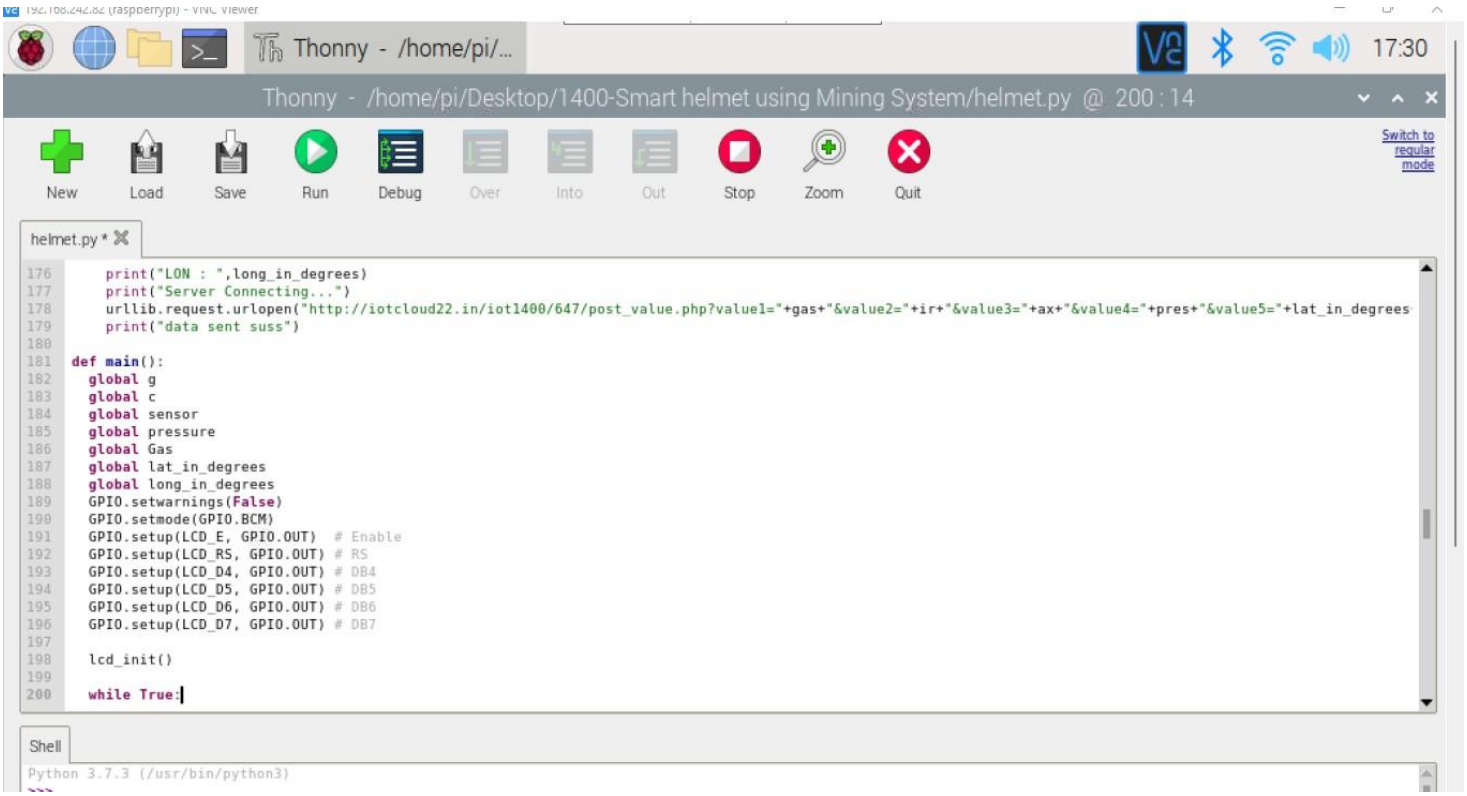

192.168.242.82 (raspberrypi) - VNC Viewer
Thonny - /home/pi/...
17:26
Thonny - /home/pi/Desktop/1400-Smart helmet using Mining System/helmet.py @ 50 : 19
New
Load
Save
Run
Debug
Over
Into
Out
Stop
Zoom
Quit
Switch to regular mode
helmet.py *
26 GPIO.setup(25,GPIO.OUT)
27
28 bus = smbus.SMBus(1)
29
30 bus.write_byte_data(0x53, 0x2C, 0x0B)
31 value = bus.read_byte_data(0x53, 0x31)
32 value &= ~0x0F;
33 value |= 0x0B;
34 value |= 0x08;
35 bus.write_byte_data(0x53, 0x31, value)
36 bus.write_byte_data(0x53, 0x2D, 0x08)
37
38 # Define GPIO to LCD mapping
39 LCD_RS = 26
40 LCD_E = 19
41 LCD_D4 = 13
42 LCD_D5 = 6
43 LCD_D6 = 5
44 LCD_D7 = 21
45
46 LCD_WIDTH = 16
47 LCD_CHR = True
48 LCD_CMD = False
49
50 LCD_LINE_1 = 0x80
Shell
Python 3.7.3 (/usr/bin/python3)
>>>

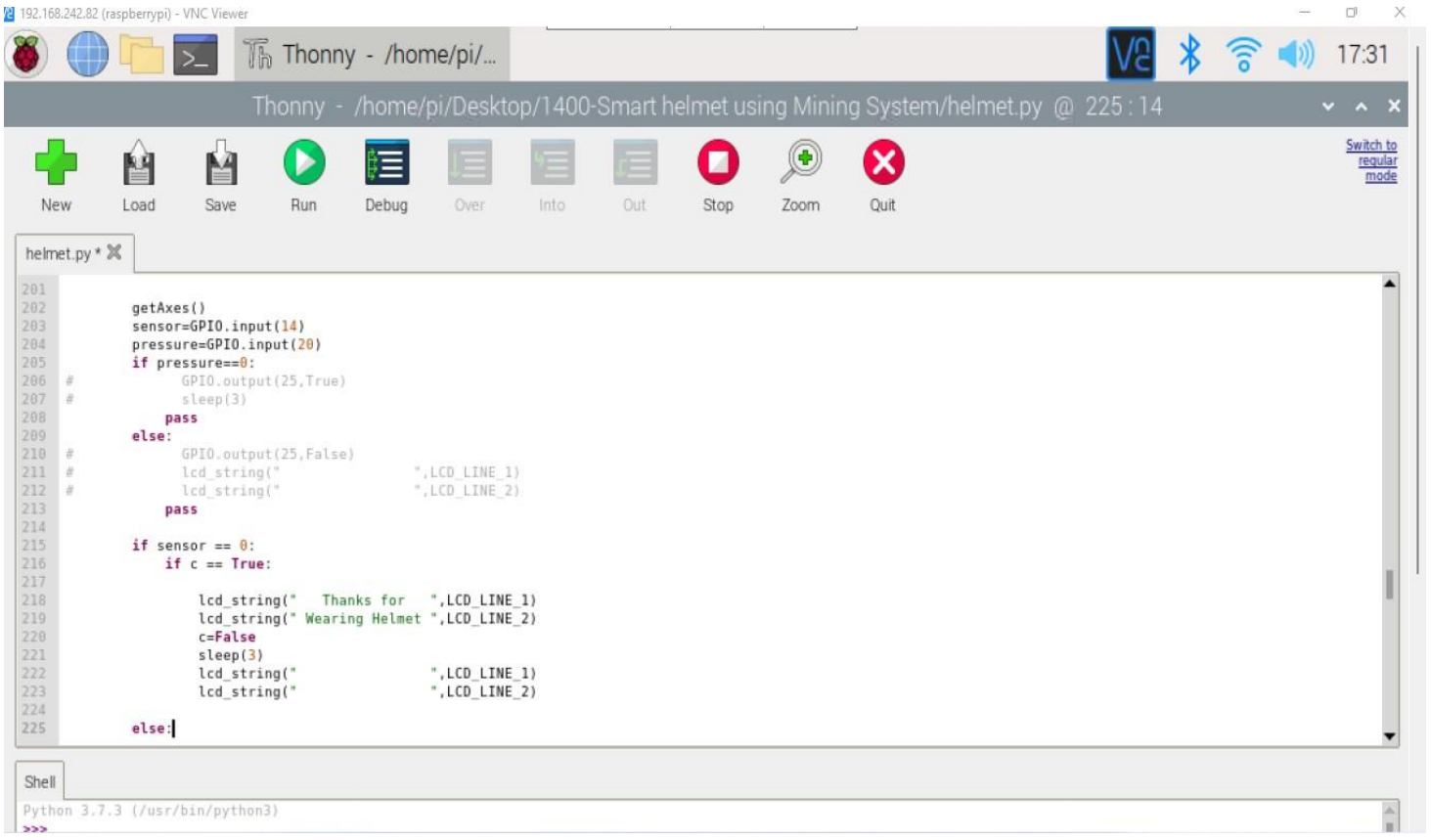

192.168.242.82 (raspberrypi) - VNC Viewer
Thonny - /home/pi/...
17:27
Thonny - /home/pi/Desktop/1400-Smart helmet using Mining System/helmet.py @ 75 : 11
New
Load
Save
Run
Debug
Over
Into
Out
Stop
Zoom
Quit
Switch to regular mode
helmet.py *
51 LCD_LINE_2 = 0xC0
52
53 E_PULSE = 0.0005
54 E_DELAY = 0.0005
55
56 def getAxes():
57 global x
58 global y
59 global z
60 globa
61 l a
62 global d
63
64 bytes = bus.read_i2c_block_data(0x53, 0x32, 6)
65
66 x = bytes[0] | (bytes[1] << 8)
67 if(x & (1 << 16 - 1)):
68 x = x - (1<<16)
69
70 y = bytes[2] | (bytes[3] << 8)
71 if(y & (1 << 16 - 1)):
72 y = y - (1<<16)
73
74 z = bytes[4] | (bytes[5] << 8)
75 if(z & (1 << 16 - 1)):
Shell
Python 3.7.3 (/usr/bin/python3)
>>>

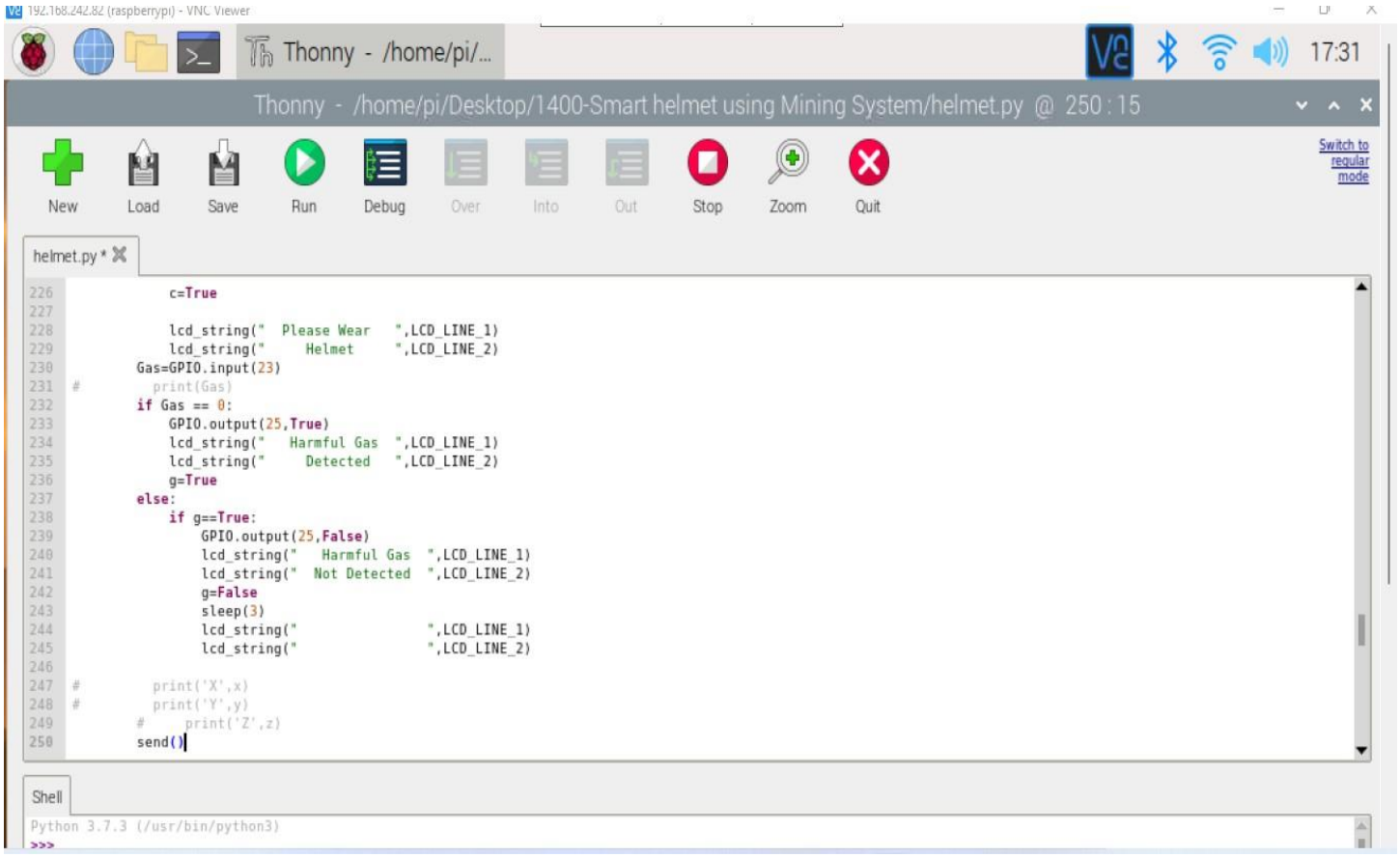

192.168.242.82 (raspberrypi) - VNC Viewer
Thonny - /home/pi/...
17:27
Thonny - /home/pi/Desktop/1400-Smart helmet using Mining System/helmet.py @ 100 : 27
New
Load
Save
Run
Debug
Over
Into
Out
Stop
Zoom
Quit
Switch to regular mode
helmet.py *
76 z = z - (1<<16)
77
78 x = round(x, 4)
79 y = round(y, 4)
80 z = round(z, 4)
81
82 # print('X : ',x)
83 # print('Y : ',y)
84 # print('Z : ',z)
85 if x < -90 or x > 90 or y < -90 or y > 90:
86 a+=1
87 elif x > -75 and x < 75:
88 a=0
89 elif y > -75 and y <75:
90 a=0
91
92
93 if a > 10:
94 GPIO.output(25,True)
95 lcd_string("Accident Happens",LCD_LINE_1)
96 lcd_string(" ",LCD_LINE_2)
97 d=1
98 print("Accident Happens")
99 sleep(3)
100 lcd_string(" ",LCD_LINE_1)
Shell
Python 3.7.3 (/usr/bin/python3)
>>>

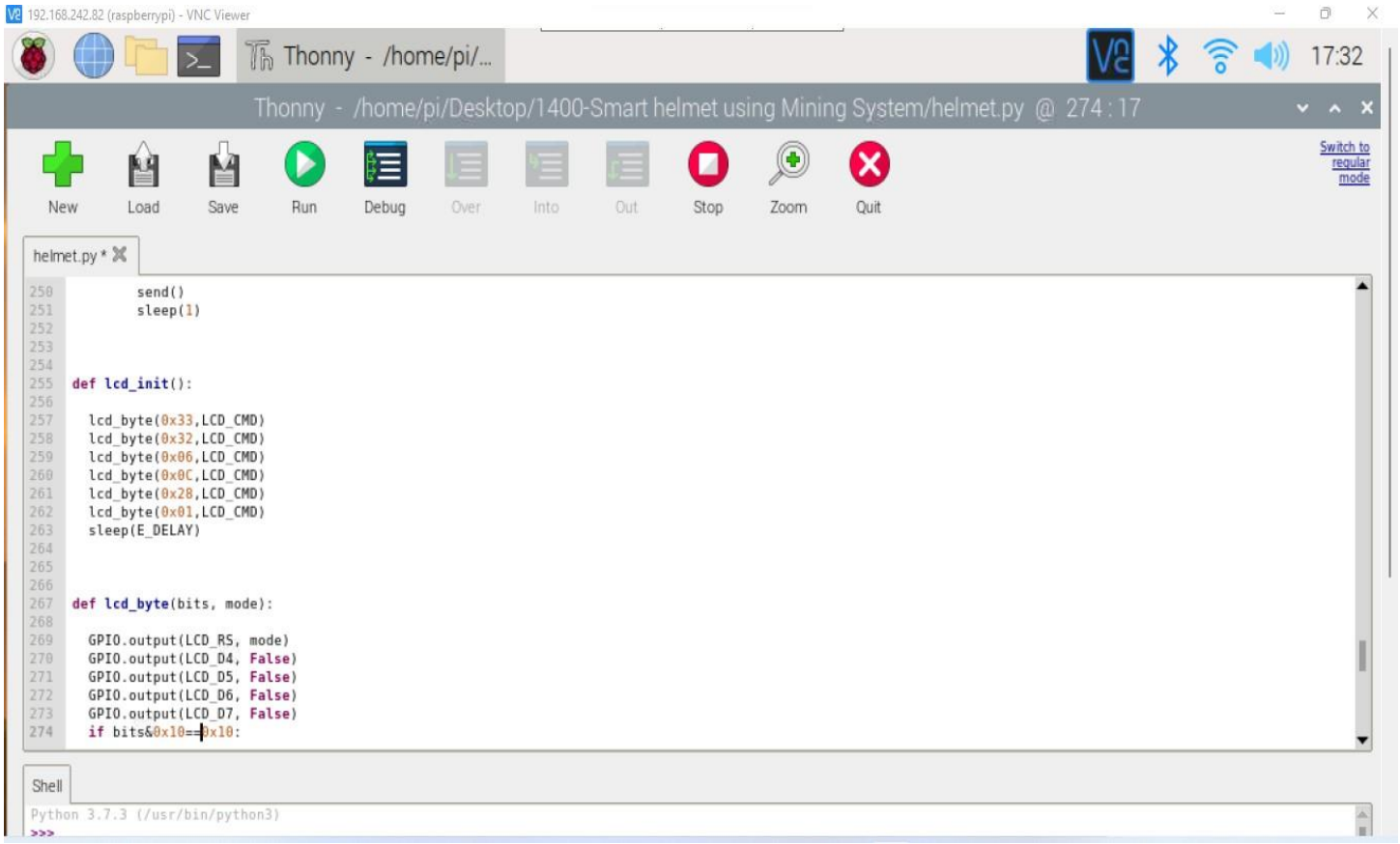

192.168.242.82 (raspberrypi) - VNC Viewer
Thonny - /home/pi/...
17:28
Thonny - /home/pi/Desktop/1400-Smart helmet using Mining System/helmet.py @ 125 : 18
New
Load
Save
Run
Debug
Over
Into
Out
Stop
Zoom
Quit
Switch to regular mode
helmet.py *
lcd_string(" ",LCD_LINE_2)
else:
GPIO.output(25,False)
d=0
print("\n\n")
return {"x": x, "y": y, "z": z}
port = "/dev/ttyS0"
ser = serial.Serial(port, baudrate = 9600, timeout = 0.5)
def convert_to_degrees(raw_value):
decimal_value = raw_value/100.00
degrees = int(decimal_value)
mm_mmmm = (decimal_value - int(decimal_value))/0.6
position = degrees + mm_mmmm
position = "%.8f" %(position)
return position
def send():
global pressure
global Gas
global sensor
Shell
Python 3.7.3 (/usr/bin/python3)
>>>

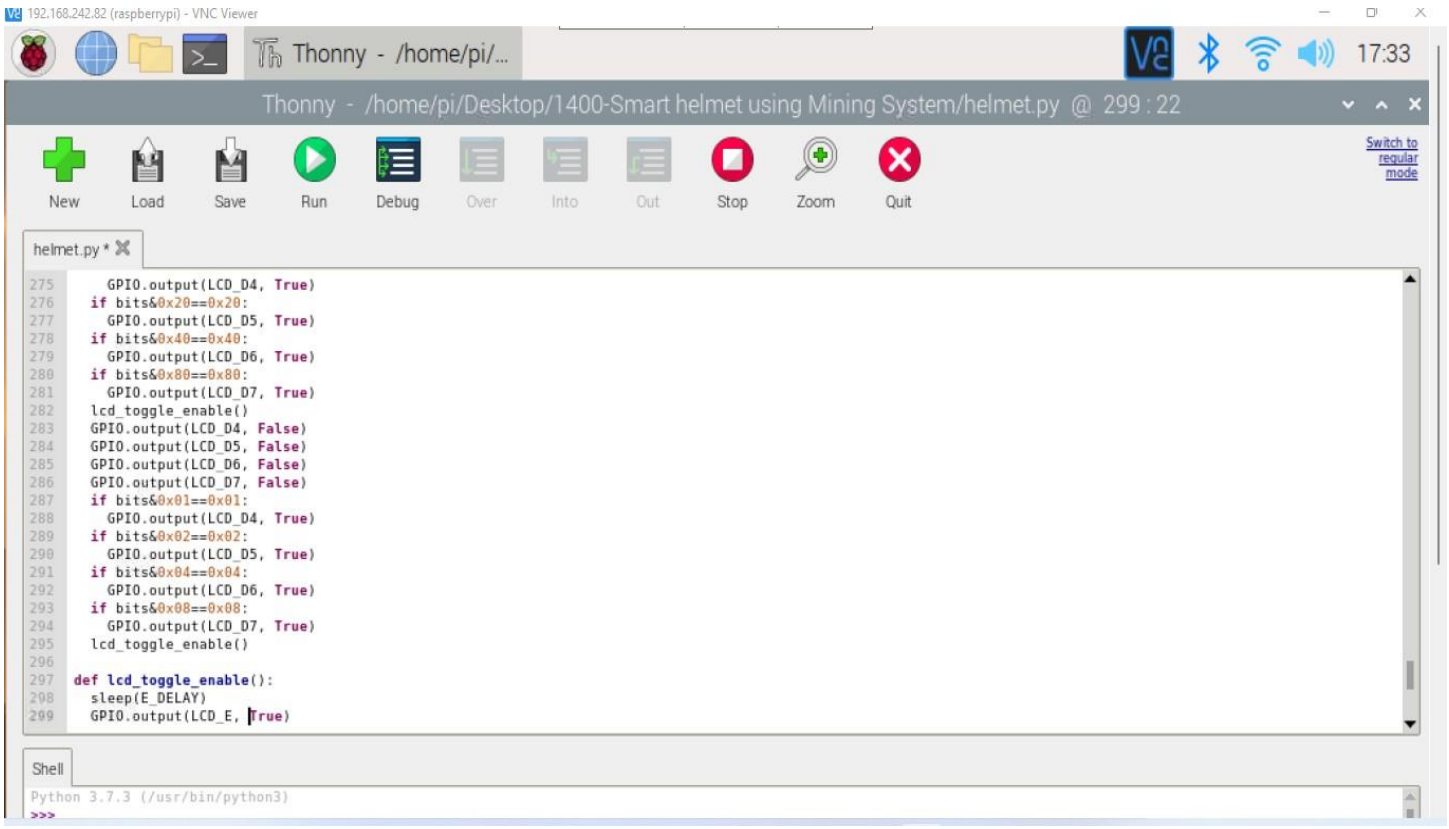

192.168.242.82 (raspberrypi) - VNC Viewer
Thonny - /home/pi/...
17:29
Thonny - /home/pi/Desktop/1400-Smart helmet using Mining System/helmet.py @ 175 : 9
New
Load
Save
Run
Debug
Over
Into
Out
Stop
Zoom
Quit
Switch to regular mode
helmet.py *
lati = float(lat)
longi = float(lon)
lat_in_degrees = convert_to_degrees(lati)
long_in_degrees = convert_to_degrees(longi)
# print ('latitude : ',lat_in_degrees)
# print ('longitude : ',long_in_degrees)
threading.Timer(600,send).start()
gas= str(Gas)
ir =str(sensor)
ax = str(d)
pres = str(pressure)
lat_in_degrees = str(lat_in_degrees)
long_in_degrees = str(long_in_degrees)
# ir ="%.1f" %sensor
print("Gas : ",gas)
print("IR : ",ir)
print("ACC : ",ax)
print("Pressure : ",pres)
print("LAT : ",lat_in_degrees)
Shell
Python 3.7.3 (/usr/bin/python3)
>>>

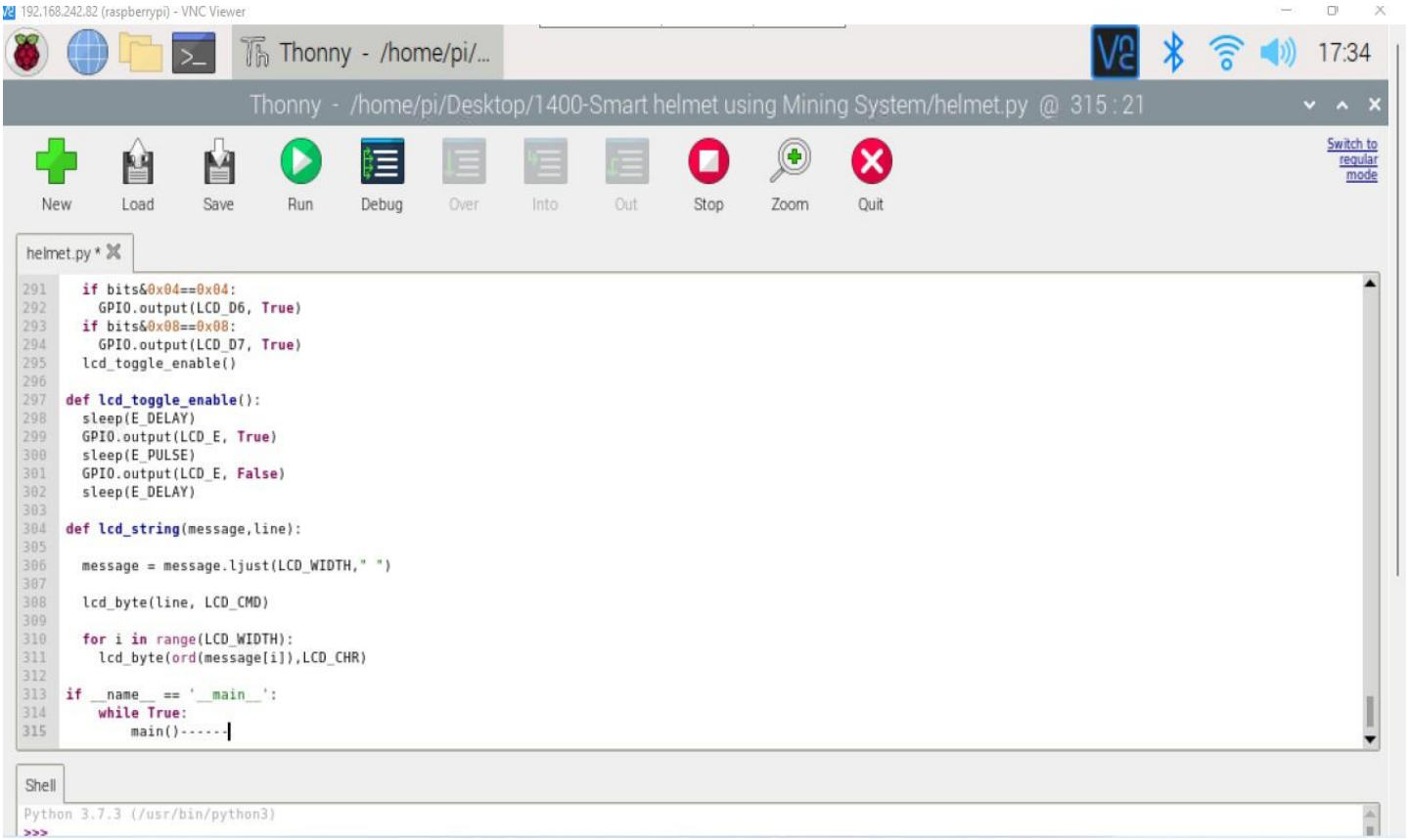

Thonny - /home/pi/...
17:30
Thonny - /home/pi/Desktop/1400-Smart helmet using Mining System/helmet.py @ 200 : 14
New
Load
Save
Run
Debug
Over
Into
Out
Stop
Zoom
Quit
Switch to regular mode
helmet.py *
print("LON : ",long_in_degrees)
print("Server Connecting...")
urllib.request.urlopen("http://iotcloud22.in/iot1400/647/post_value.php?value1="+gas+"&value2="+ir+"&value3="+ax+"&value4="+pres+"&value5="+lat_in_degrees
print("data sent suss")
def main():
global g
global c
global sensor
global pressure
global Gas
global lat_in_degrees
global long_in_degrees
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)
GPIO.setup(LCD_E, GPIO.OUT) # Enable
GPIO.setup(LCD_RS, GPIO.OUT) # RS
GPIO.setup(LCD_D4, GPIO.OUT) # DB4
GPIO.setup(LCD_D5, GPIO.OUT) # DB5
GPIO.setup(LCD_D6, GPIO.OUT) # DB6
GPIO.setup(LCD_D7, GPIO.OUT) # DB7
lcd_init()
while True:
Shell
Python 3.7.3 (/usr/bin/python3)


192.168.242.82 (raspberrypi) - VNC Viewer
Thonny - /home/pi/...
17:31
Thonny - /home/pi/Desktop/1400-Smart helmet using Mining System/helmet.py @ 225 : 14
New
Load
Save
Run
Debug
Over
Into
Out
Stop
Zoom
Quit
Switch to regular mode
helmet.py *
getAxes()
sensor=GPIO.input(14)
pressure=GPIO.input(20)
if pressure==0:
# GPIO.output(25,True)
# sleep(3)
pass
else:
# GPIO.output(25,False)
# lcd_string(" ",LCD_LINE_1)
# lcd_string(" ",LCD_LINE_2)
pass
if sensor == 0:
if c == True:
lcd_string(" Thanks for ",LCD_LINE_1)
lcd_string(" Wearing Helmet ",LCD_LINE_2)
c=False
sleep(3)
lcd_string(" ",LCD_LINE_1)
lcd_string(" ",LCD_LINE_2)
else:
Shell
Python 3.7.3 (/usr/bin/python3)


192.168.242.82 (raspberrypi) - VNC Viewer
Thonny - /home/pi/...
17:31
Thonny - /home/pi/Desktop/1400-Smart helmet using Mining System/helmet.py @ 250 : 15
New
Load
Save
Run
Debug
Over
Into
Out
Stop
Zoom
Quit
Switch to regular mode
helmet.py *
c=True
lcd_string(" Please Wear ",LCD_LINE_1)
lcd_string(" Helmet ",LCD_LINE_2)
Gas=GPIO.input(23)
# print(Gas)
if Gas == 0:
GPIO.output(25,True)
lcd_string(" Harmful Gas ",LCD_LINE_1)
lcd_string(" Detected ",LCD_LINE_2)
g=True
else:
if g==True:
GPIO.output(25,False)
lcd_string(" Harmful Gas ",LCD_LINE_1)
lcd_string(" Not Detected ",LCD_LINE_2)
g=False
sleep(3)
lcd_string(" ",LCD_LINE_1)
lcd_string(" ",LCD_LINE_2)
# print('X',x)
# print('Y',y)
# print('Z',z)
send()
Shell
Python 3.7.3 (/usr/bin/python3)
>>>

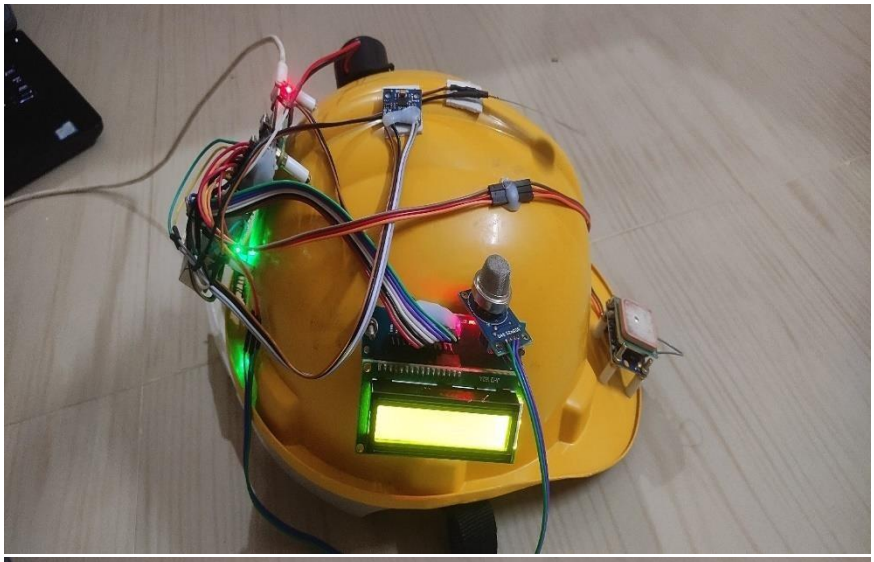

192.168.242.82 (raspberrypi) - VNC Viewer
Thonny - /home/pi/...
17:32
Thonny - /home/pi/Desktop/1400-Smart helmet using Mining System/helmet.py @ 274 : 17
New
Load
Save
Run
Debug
Over
Into
Out
Stop
Zoom
Quit
Switch to regular mode
helmet.py *
send()
sleep(1)
def lcd_init():
lcd_byte(0x33,LCD_CMD)
lcd_byte(0x32,LCD_CMD)
lcd_byte(0x06,LCD_CMD)
lcd_byte(0x0C,LCD_CMD)
lcd_byte(0x28,LCD_CMD)
lcd_byte(0x01,LCD_CMD)
sleep(E_DELAY)
def lcd_byte(bits, mode):
GPIO.output(LCD_RS, mode)
GPIO.output(LCD_D4, False)
GPIO.output(LCD_D5, False)
GPIO.output(LCD_D6, False)
GPIO.output(LCD_D7, False)
if bits&0x10==0x10:
Shell
Python 3.7.3 (/usr/bin/python3)
>>>

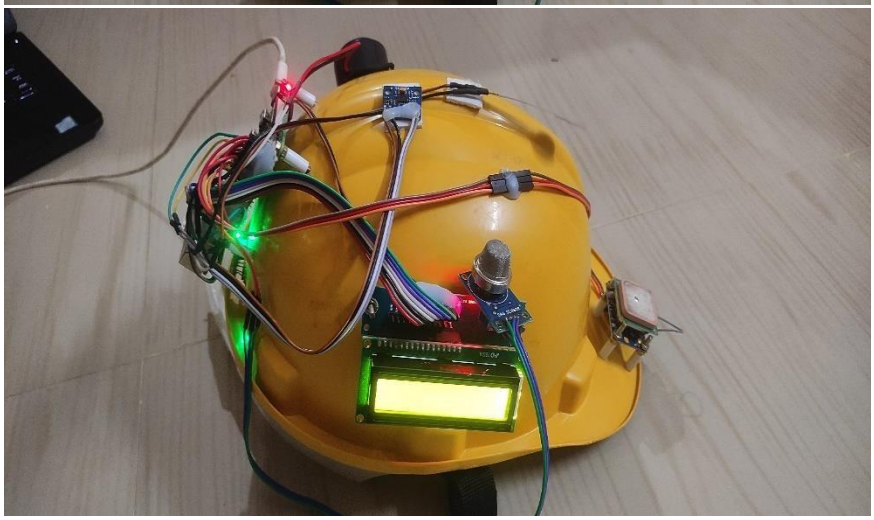

192.168.242.82 (raspberrypi) - VNC Viewer
Thonny - /home/pi/...
17:33
Thonny - /home/pi/Desktop/1400-Smart helmet using Mining System/helmet.py @ 299 : 22
New
Load
Save
Run
Debug
Over
Into
Out
Stop
Zoom
Quit
Switch to regular mode
helmet.py *
GPIO.output(LCD_D4, True)
if bits&0x20==0x20:
GPIO.output(LCD_D5, True)
if bits&0x40==0x40:
GPIO.output(LCD_D6, True)
if bits&0x80==0x80:
GPIO.output(LCD_D7, True)
lcd_toggle_enable()
GPIO.output(LCD_D4, False)
GPIO.output(LCD_D5, False)
GPIO.output(LCD_D6, False)
GPIO.output(LCD_D7, False)
if bits&0x01==0x01:
GPIO.output(LCD_D4, True)
if bits&0x02==0x02:
GPIO.output(LCD_D5, True)
if bits&0x04==0x04:
GPIO.output(LCD_D6, True)
if bits&0x08==0x08:
GPIO.output(LCD_D7, True)
lcd_toggle_enable()
def lcd_toggle_enable():
sleep(E_DELAY)
GPIO.output(LCD_E, True)
Shell
Python 3.7.3 (/usr/bin/python3)
>>>

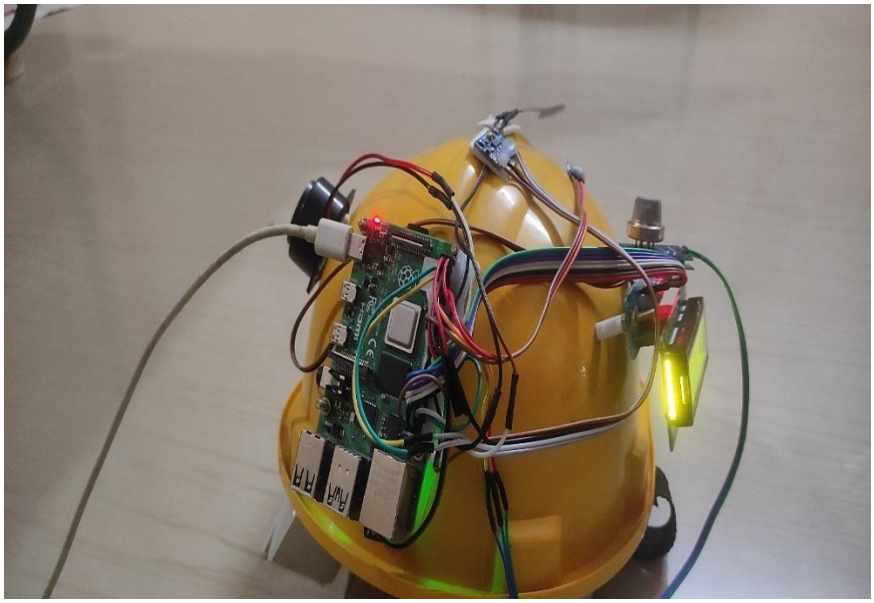

192.168.242.82 (raspberrypi) - VNC Viewer
Thonny - /home/pi/...
17:34
Thonny - /home/pi/Desktop/1400-Smart helmet using Mining System/helmet.py @ 315 : 21
New
Load
Save
Run
Debug
Over
Into
Out
Stop
Zoom
Quit
Switch to regular mode
helmet.py *
if bits&0x04==0x04:
GPIO.output(LCD_D6, True)
if bits&0x08==0x08:
GPIO.output(LCD_D7, True)
lcd_toggle_enable()
def lcd_toggle_enable():
sleep(E_DELAY)
GPIO.output(LCD_E, True)
sleep(E_PULSE)
GPIO.output(LCD_E, False)
sleep(E_DELAY)
def lcd_string(message,line):
message = message.ljust(LCD_WIDTH," ")
lcd_byte(line, LCD_CMD)
for i in range(LCD_WIDTH):
lcd_byte(ord(message[i]),LCD_CHR)
if __name__ == '__main__':
while True:
main()------
Shell
Python 3.7.3 (/usr/bin/python3)
>>>

SCREENSHOT

HELMET IMAGE







BEFORE ACCIDENT DETECTED

AFTER ACCIDENT DETECTED