Import sys

Sys.path.append(‘/home/pi/py/adafruit-raspberry-pi-python-code/adafruit\_charLCDplate’)

From time import sleep

From adafruit\_charLCDplate import adafruit\_charLCDplate

Import mcp3008

From controlmypi import controlmypi

Import logging

Import datatime

Import pickle

From genericpath import exists

Import smtplib

Lcd = adafruit\_charcLCDplate()

Pickle\_file=’/home/pi/py/moisture/moisk.pkl’

Def on\_msg(conn, key, value):

Pass

Def append\_chart\_point(chart,point):

If len(chart) >= 48:

Del chart[0]

Chart.append(point)

Return chart

Def load(default):

If not exists(pickle\_file):

Return default

Pkl\_file=open(pickle\_file,’rb’)

Data=pickle.load(pkl\_file)

Pkl\_file.close()

Return data

Def update\_lcd(m):

Try:

Lcd.home()

Lcd.messge(“moisture level:\n%d%% “%m)

If m<15:

Lcd.backlight(lcd.red)

Elif m<50:

Lcd.backlight(lcd.yellow)

Else:

Lcd.backlight(lcd.green)

Except IOerror as e:

Print e

Def send\_gmail(from\_name,sender,password,recipient,subject,body):

Senddate=datetime.datatime.strftime(datatime.datatime.now(), ‘%Y-%m-%d’)

Msg=”date:%s\r\nFrom: %s<%S>\r\nTo: %s\r\nSubject: %s\r\nX-Mailer: My-Mail\r\n\r\n”% (senddate,from\_name, sender, recipient, subject)

Server=smtplib.SMTP(‘smtp.gmail.com:587’)

Server.login(sender,password)

Server.login(server,password)

Server.sendmail(sender,recipient,msg+body)

Server.quit()

Logging.basicConfig(level=logging.INFO)

P=[

[ [‘G’, ‘moist’, ‘level’,0,0,100],[‘LC’, ‘chart1’, ‘Time’, ‘Value’,0,100]],

]

C1=load([])

Readings=[]

Conn=ControlMyPi(‘you@gmail.com’,’password’,’moisture’,moisture moniter’,p, on\_msg)

Delta=datatime.timedelta(minutes=30)

Next\_time=datatime.datatime.now()

Delta\_email=datatime.timedelta(days=1)

Next\_email\_time=datatime.datatime.now()

If conn.start\_control();

Try:

While true:

Dt= datetime.datatime.now()

M=mcp3008.read\_pct(5)

Readings.append(m)

To\_update={‘moisture’:m}

#update the chart?

If dt>next\_time:

#take the average the reading list to smooth a graph a little

Avg=int(round(sum(reading)/len(reading)))

Reading=[]

Cl=append\_chart\_point\_point(cl,[dt.strftime(‘%H:%M’),avg])

Save(cl)

Next\_time=dt+delta

To\_update[‘chart1’]=cl

Conn.update\_status(to\_update)

#send an email?

If dt> next\_email\_time:

Next\_email\_time=dt+delta\_email

If m<40:

Send\_gmail(‘Your Name’,’you@gmail.com’,’password’,’recipient@gmail.com’,’Moisture sensor level’,’the the is now: %s’ %m)

Sleep(30)

Finally:

Conn.stop\_control().