Application

Weather Station

- Hardware setup
- Clock synchronization
- Google Spreadsheets

Hardware setup

- A sensor station
- A temperature sensor (LM35/TMP36)
- A GPIO ribbon cable for Raspberry Pi
- Male-to-female and female-to-female jumper wires
- A Raspberry Pi, a power adapter, an Ethernet cable, a personal computer

Clock Synchronization

- No Real Time Clock (RTC) for Pi
- Solution:
  - External RTC
  - Using Network Time Protocol (NTP)

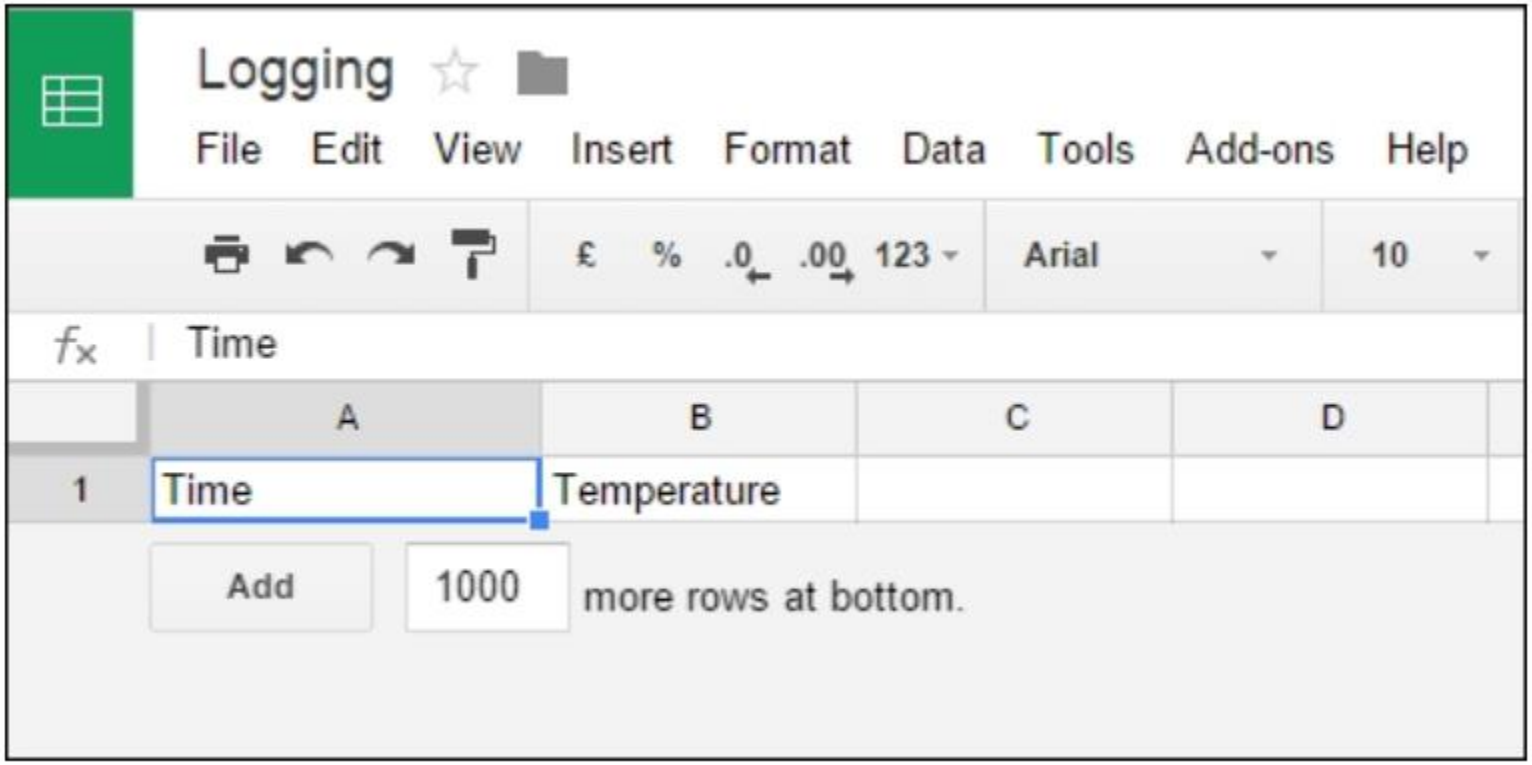
sudo nano /etc/rc.conf

DAEMONS=(!hwclock ntpd ntpdate)

Google Spreadsheets

- Log in to your Gmail account or create a new one.
- Open or link Google Drive by logging onto http://www.drive.google.com with your Gmail ID.
- In the user interface of Google Drive, on the left-side panel, click on the New button and select Spreadsheet from the drop-down menu. You will be redirected to a new tab.
- In the new tab, name the spreadsheet Logging. Avoid blank spaces in the name to reduce errors, as we are going to use this name in our code.
- Click on the Add-ons drop-down menu from one of the spreadsheet menus, and then check out Get Add-ons…. In the search bar, type Remove Blank Rows. Click on the free button and install it.
- In the first row of the spreadsheet, enter the column titles as Time and Temperature.
- Click on the Add-ons menu again and then on the Remove Blank Rows add-on. Check out the Delete or hide blank rows option. You will be able to see the sidebar with some options. Click on the top-left corner of the spreadsheet to select all rows and columns. Then, in the sidebar, select the All row cells must be blank option and delete all blank rows. It should look like what is shown in the following screenshot.
- Close the tab, and we are ready to code in the RasPi.

Google Logging



Install Gspred library

git clone https://github.com/burnash/gspread.git

cd gspread

python setup.py install

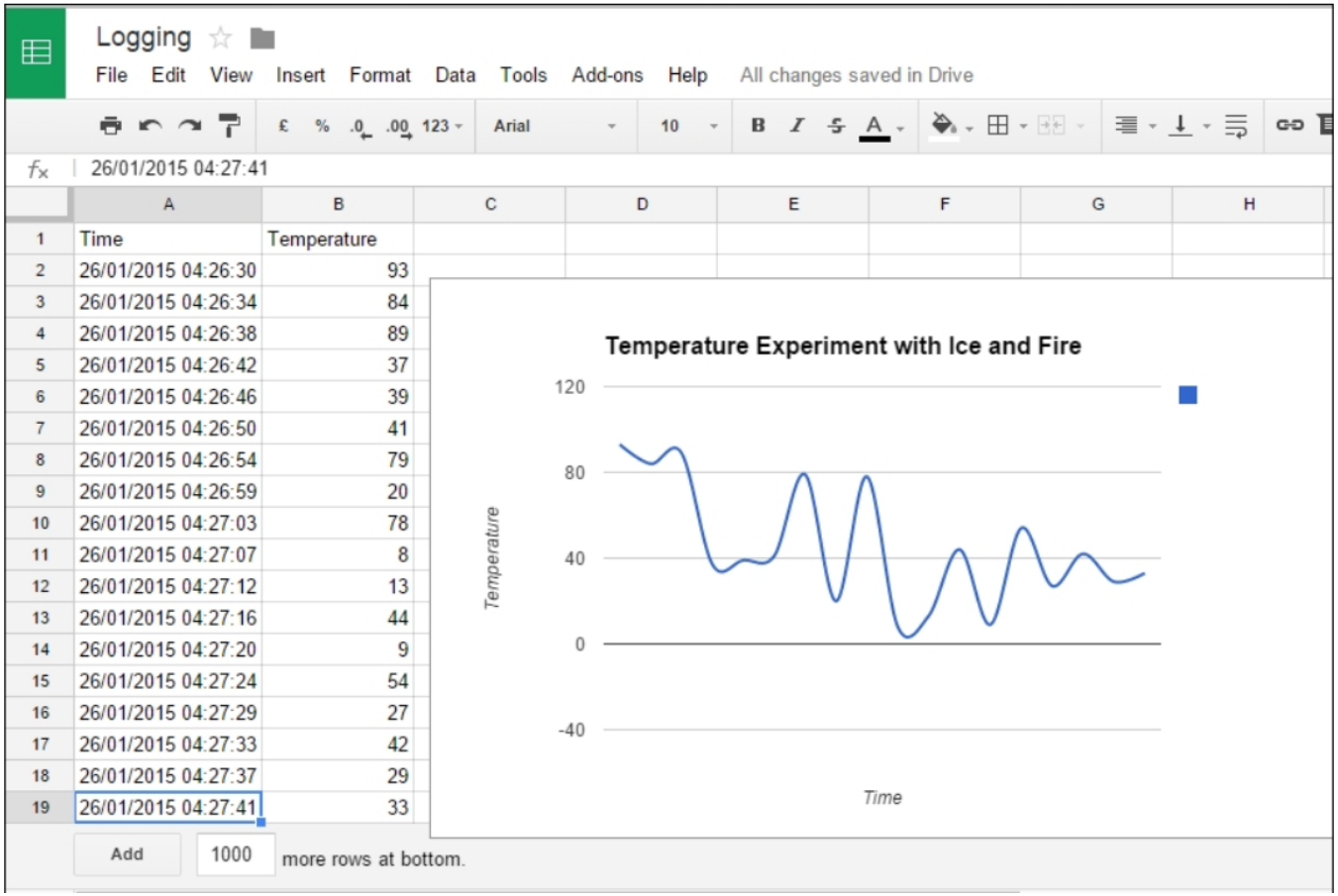
Recall MCP3008 Library

```
#start the SPI bus by opening the spi port
spi = spidev.SpiDev()
spi.open(0,0)
#function to read the channels of MCP3008
def readadc(channel):
    value = spi.xfer2([1,(8+channel)<<4,0])
    read = ((value[1]&3) << 8) + value[2]
    return read
```

Code

```
import os
import spidev
import glob
import time
import sys
import datetime
import gspread

#start the SPI bus by opening the spi port
spi = spidev.SpiDev()
spi.open(0,0)

# Enter your account details (Your Gmail ID and Password) as shown here
email = 'gajjar.rushi@gmail.com'
password = 'raspberrypi'

#Name of Spreadsheet created in Google Drive
spreadsheet = 'Logging'

#Putting the exception call in python to attempt for logging in Gmail
try:
  ret = gspread.login(email,password)
except:
  print('Oops! Check Internet Connection or Login Credentials')
  sys.exit() #open the spreadsheet by either of these two options
  worksheet = ret.open(spreadsheet).sheet1

#or with the spreadsheet key
#worksheet = ret.open_by_key('1eQth-TY4FXFKChB5RFPhelQ6zn47NWDESh13WkXGQAk')
#prefer First Option
def readadc(channel):
  value = spi.xfer2([1,(8+channel)<<4,0])
  read = ((value[1]&3) << 8) + value[2]
  return read

while True:
  #Get data from Channel 0, TMP36 Temperature Sensor
  val = readadc(0)
  temperature = ((val * 330)/float(1023))-50
  values = [datetime.datetime.now(), temperature]
  worksheet.append_row(values)
  time.sleep(5)
```

Output



	A	B
1	Time	Temperature
2	26/01/2015 04:26:30	93
3	26/01/2015 04:26:34	84
4	26/01/2015 04:26:38	89
5	26/01/2015 04:26:42	37
6	26/01/2015 04:26:46	39
7	26/01/2015 04:26:50	41
8	26/01/2015 04:26:54	79
9	26/01/2015 04:26:59	20
10	26/01/2015 04:27:03	78
11	26/01/2015 04:27:07	8
12	26/01/2015 04:27:12	13
13	26/01/2015 04:27:16	44
14	26/01/2015 04:27:20	9
15	26/01/2015 04:27:24	54
16	26/01/2015 04:27:29	27
17	26/01/2015 04:27:33	42
18	26/01/2015 04:27:37	29
19	26/01/2015 04:27:41	33

Questions?