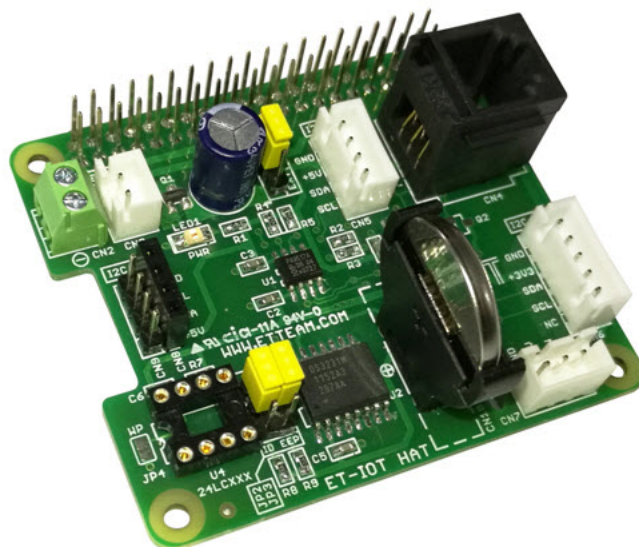


## ET-IOT HAT

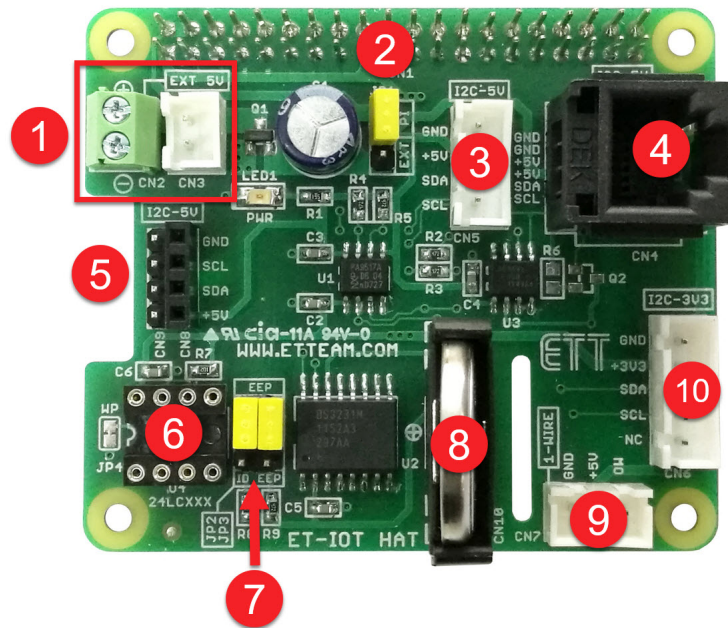


ET-IOT HAT เป็นบอร์ดที่ออกแบบมาใช้งานร่วมกับบอร์ด Raspberry Pi โดยใช้ในการเชื่อมต่อแบบ I2C Bus เพื่อให้สามารถเชื่อมต่อกับบอร์ด INPUT , OUTPUT หรือบอร์ดเซ็นเซอร์ ต่าง ๆ ที่ใช้ในการเชื่อมต่อแบบ I2C Bus หรือ 1-Wire

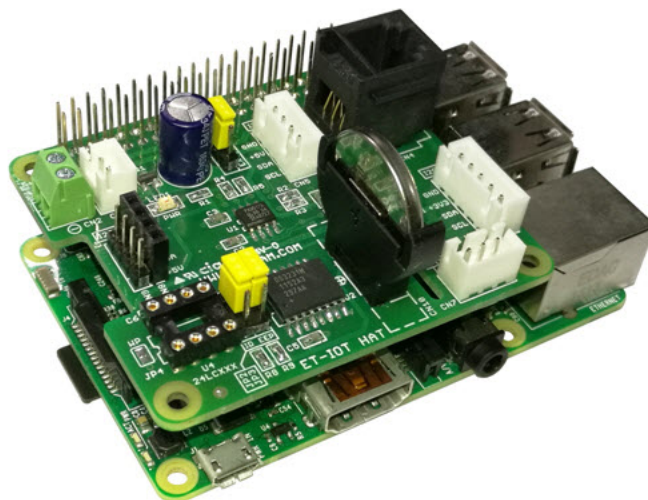
**คุณสมบัติของบอร์ด ET-IOT HAT**

- สามารถใช้งานร่วมกับบอร์ด Raspberry Pi โดยใช้ในการเชื่อมต่อแบบ I2C Bus
- มีส่วนเชื่อมต่อของวงจร EEPROM (OPTION) โดยสามารถเลือกใช้เป็น ID EEPROM หรือ EEPROM ธรรมดาได้
- มีส่วนของวงจร RTC (DS3231) สำหรับใช้เป็นฐานเวลาของบอร์ด Raspberry Pi พร้อมแบตเตอรี่ Backup
- มีจุดเชื่อมต่อ I2C Bus สำหรับเชื่อมต่อกับอุปกรณ์ภายนอกทั้งระบบ 3.3V และ 5V
- มีวงจร I2C to 1-Wire ( DS2482-100 ) สำหรับเชื่อมต่อกับอุปกรณ์ที่ใช้การเชื่อมต่อแบบ 1-Wire
- มีขั้วต่อแหล่งจ่าย 5VDC จากภายนอก สำหรับจ่ายให้กับบอร์ดที่มาเชื่อมต่อ โดยสามารถเลือกว่าจะใช้ไฟจากบอร์ด Raspberry Pi หรือจากภายนอก โดยใช้จัมเปอร์
- ขนาด PCB 6.5 x 5.6 cm.

## ส่วนประกอบของบอร์ด ET-IOT HAT



- หมายเลข 1 ขั้วต่อแหล่งจ่าย 5V จากภายนอก สำหรับจ่ายให้กับบอร์ดที่มาเชื่อมต่อ ในกรณีที่แหล่งจ่ายของบอร์ด Raspberry Pi ไม่สามารถจ่ายได้เพียงพอ
- หมายเลข 2 จัมเปอร์ (JP1) สำหรับเลือกแหล่งจ่าย 5V ว่าจะใช้จากบอร์ด Raspberry Pi (PI) หรือ จากภายนอก (EXT)
- หมายเลข 3 คอนเน็คเตอร์เชื่อมต่อ I2C Bus แบบ Wafer 4 Pin (CN5) สามารถใช้กับระบบที่ใช้แรงดัน 5V
- หมายเลข 4 คอนเน็คเตอร์เชื่อมต่อ I2C Bus แบบ RJ11 6 Pin (CN4) สามารถใช้กับระบบที่ใช้แรงดัน 5V
- หมายเลข 5 คอนเน็คเตอร์เชื่อมต่อ I2C Bus แบบ Header 4 Pin สามารถใช้กับระบบที่ใช้แรงดัน 5V
- หมายเลข 6 ส่วนเชื่อมต่อของวงจร EEPROM (OPTION) โดยสามารถเลือกใช้เป็น ID EEPROM หรือ EEPROM ธรรมดาได้
- หมายเลข 7 จัมเปอร์ (JP2, JP3) สำหรับเลือกการเชื่อมต่อ EEPROM ว่าจะใช้เป็น ID EEPROM (ตำแหน่ง ID EEP) หรือ EEPROM ธรรมดา (ตำแหน่ง EEP)
- หมายเลข 8 แบตเตอรี่ Backup (CR2032) สำหรับวงจร RTC (DS3231)
- หมายเลข 9 คอนเน็คเตอร์เชื่อมต่อ 1-Wire แบบ Wafer 3 Pin (CN7) สามารถใช้กับระบบที่ใช้แรงดัน 5V
- หมายเลข 10 คอนเน็คเตอร์เชื่อมต่อ I2C Bus แบบ Wafer 5 Pin (CN6) สามารถใช้กับระบบที่ใช้แรงดัน 3.3V

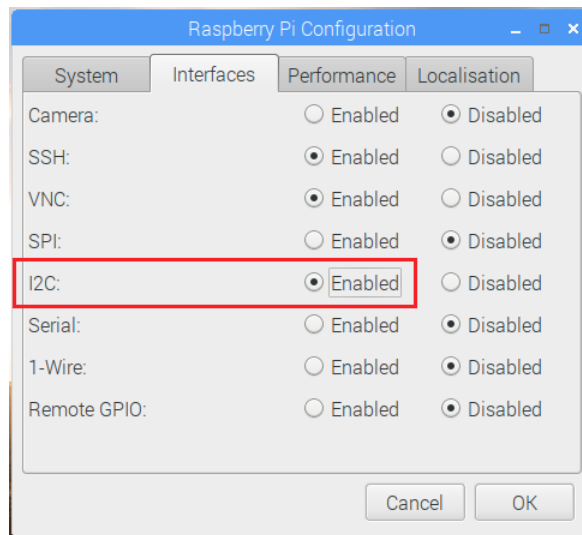
แสดงการต่อใช้งานบอร์ด ET-IOT HAT ร่วมกับบอร์ด Raspberry Piการตั้งค่าบอร์ด Raspberry Pi เพื่อให้ใช้งานร่วมกับ ET-IOT HAT

เนื่องจากบอร์ด ET-IOT HAT ใช้การเชื่อมต่อแบบ I2C Bus ดังนั้นจำเป็นต้องเปิดการเชื่อมต่อ I2C Bus บนบอร์ด Raspberry Pi ก่อน โดยขั้นตอนมีดังนี้ ซึ่งตัวอย่างนี้ทดลองกับบอร์ด Raspberry Pi 3 Mode B และระบบปฏิบัติการ RASPBIAN STRETCH WITH DESKTOP Version: April 2018 Release date: 2018-04-18

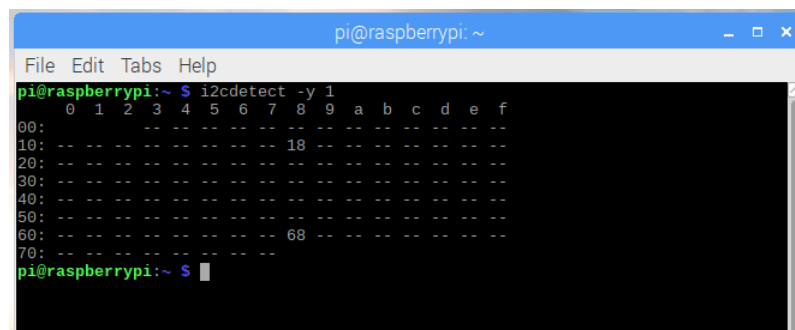
1. เชื่อมต่อบอร์ด ET-IOT HAT กับ บอร์ด Raspberry Pi จากนั้นจ่ายไฟเข้าบอร์ด Raspberry Pi
2. เลือกที่เมนู Preferences --- Raspberry Pi Configuration



3. เลือกที่แท็บ Interface และเลือกหัวข้อ I2C เป็น Enable และคลิก OK ดังรูป จากนั้นให้ Reboot บอร์ด Raspberry Pi



4. จากนั้นให้เปิดโปรแกรม Terminal และพิมพ์คำสั่ง `i2cdetect -y 1` ดังรูป

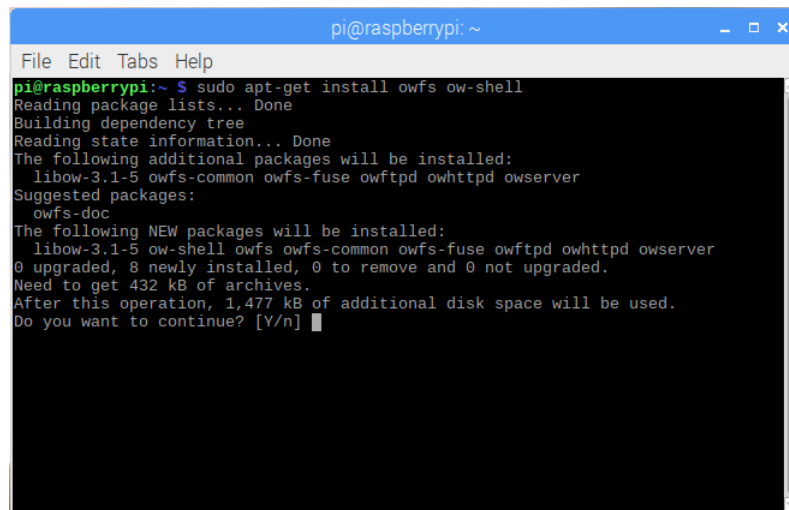
The image shows a terminal window on a Raspberry Pi. The command `i2cdetect -y 1` has been executed, resulting in a grid of dashes representing the I2C bus scan. The grid shows a '18' at the 10:00 position and a '68' at the 60:00 position, indicating the presence of devices at those addresses. The terminal prompt is `pi@raspberrypi:~$`.

หลังจากนั้นโปรแกรมจะสแกนหาอุปกรณ์ I2C ที่เชื่อมต่ออยู่ ซึ่งบนบอร์ด ET-IOT HAT อยู่ 2 ตัว คือ DS2482-100 Address 18 และ DS3231 Address 68 ถ้าแสดงแบบนี้แสดงว่าบอร์ด ET-IOT HAT พร้อมใช้งานแล้ว

ตัวอย่างการใช้งาน PROBE DS18B20 HAT ร่วมกับ ET-IOT HAT

PROBE DS18B20 HAT จะใช้เซ็นเซอร์วัดอุณหภูมิแบบ DS18B20 ซึ่งจะใช้การเชื่อมต่อแบบ 1-Wire ดังนั้นการใช้งานให้เอา PROBE DS18B20 HAT ไปต่อที่คอนเน็คเตอร์เชื่อมต่อ 1-Wire แบบ Wafer 3 Pin ( CN7) โดยในตัวอย่างนี้จะใช้ OWFS 1-Wire File System เป็นตัวช่วยในการเขียนโปรแกรมเพื่อติดต่อกับอุปกรณ์ที่ใช้การเชื่อมต่อแบบ 1-Wire ซึ่งมีขั้นตอนดังนี้

1. เปิดโปรแกรม Terminal และพิมพ์คำสั่ง `sudo apt-get install owfs ow-shell` เพื่อติดตั้ง owfs และ ow-shell

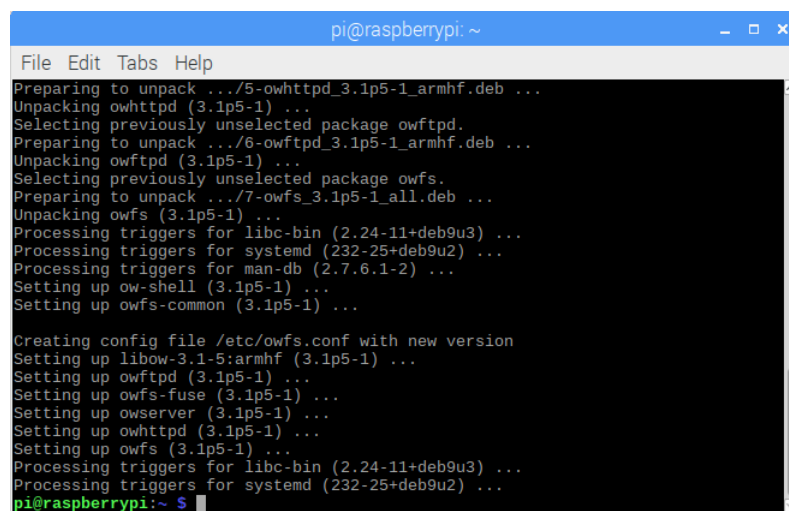


```

pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi:~ $ sudo apt-get install owfs ow-shell
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libow-3.1-5 owfs-common owfs-fuse owftpd owhttpd owserver
Suggested packages:
  owfs-doc
The following NEW packages will be installed:
  libow-3.1-5 ow-shell owfs owfs-common owfs-fuse owftpd owhttpd owserver
0 upgraded, 8 newly installed, 0 to remove and 0 not upgraded.
Need to get 432 kB of archives.
After this operation, 1,477 kB of additional disk space will be used.
Do you want to continue? [Y/n]

```

2. พิมพ์ Y และกด ENTER เพื่อยืนยัน และรอจนการติดตั้งเสร็จ



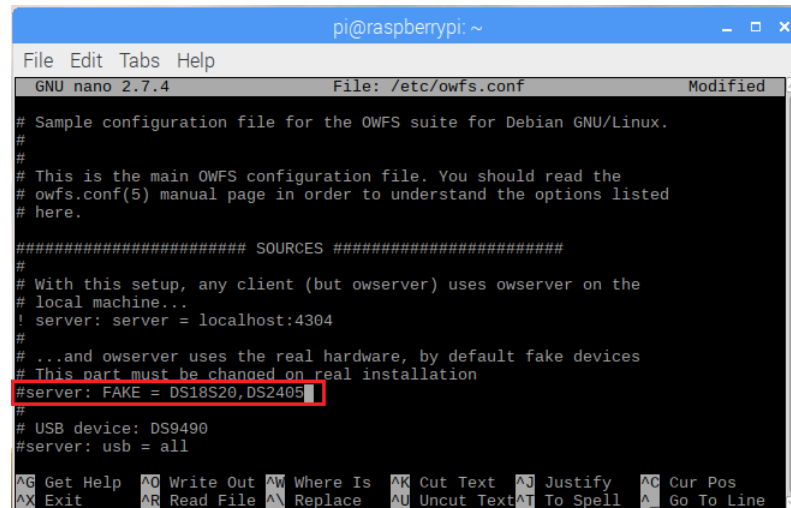
```

pi@raspberrypi: ~
File Edit Tabs Help
Preparing to unpack .../5-owhttpd_3.1p5-1_armhf.deb ...
Unpacking owhttpd (3.1p5-1) ...
Selecting previously unselected package owftpd.
Preparing to unpack .../6-owftpd_3.1p5-1_armhf.deb ...
Unpacking owftpd (3.1p5-1) ...
Selecting previously unselected package owfs.
Preparing to unpack .../7-owfs_3.1p5-1_all.deb ...
Unpacking owfs (3.1p5-1) ...
Processing triggers for libc-bin (2.24-11+deb9u3) ...
Processing triggers for systemd (232-25+deb9u2) ...
Processing triggers for man-db (2.7.6.1-2) ...
Setting up ow-shell (3.1p5-1) ...
Setting up owfs-common (3.1p5-1) ...

Creating config file /etc/owfs.conf with new version
Setting up libow-3.1-5:armhf (3.1p5-1) ...
Setting up owftpd (3.1p5-1) ...
Setting up owfs-fuse (3.1p5-1) ...
Setting up owserver (3.1p5-1) ...
Setting up owhttpd (3.1p5-1) ...
Setting up owfs (3.1p5-1) ...
Processing triggers for libc-bin (2.24-11+deb9u3) ...
Processing triggers for systemd (232-25+deb9u2) ...
pi@raspberrypi:~ $

```

3. ทำการสร้างโฟลเดอร์ 1wire ใน /mnt โดยใช้คำสั่ง `sudo mkdir /mnt/1wire`
4. ทำการเปิดไฟล์ `owfs.conf` เพื่อทำการแก้ไขโดยใช้คำสั่ง `sudo nano /etc/owfs.conf` จากนั้นให้พิมพ์เครื่องหมาย # เพิ่มหน้าข้อความ  
server: FAKE = DS18S20,DS2405 ดังรูป



```

pi@raspberrypi: ~
File Edit Tabs Help
GNU nano 2.7.4 File: /etc/owfs.conf Modified
# Sample configuration file for the OWFS suite for Debian GNU/Linux.
#
# This is the main OWFS configuration file. You should read the
# owfs.conf(5) manual page in order to understand the options listed
# here.
##### SOURCES #####
# With this setup, any client (but owserver) uses owserver on the
# local machine...
! server: server = localhost:4304
# ...and owserver uses the real hardware, by default fake devices
# This part must be changed on real installation
#server: FAKE = DS18S20,DS2405
# USB device: DS9490
#server: usb = all
AG Get Help AO Write Out AW Where Is AK Cut Text AJ Justify AC Cur Pos
AX Exit AR Read File AL Replace AU Uncut Text AT To Spell A Go To Line

```

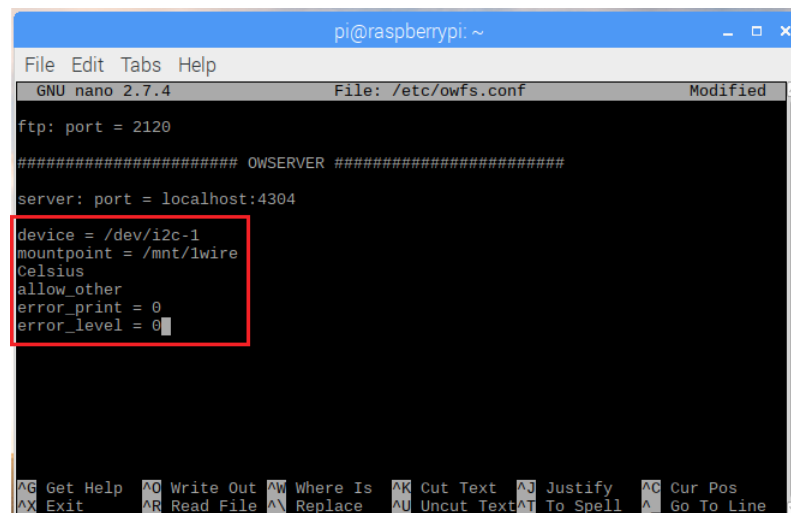
5. พิมพ์ข้อความต่อไปนี้ด้านล่างของไฟล์

```

device = /dev/i2c-1
mountpoint = /mnt/1wire
Celsius
allow_other
error_print = 0
error_level = 0

```

จากนั้นทำการบันทึก โดยการกด CTRL+X และกด y เพื่อบันทึก



```

pi@raspberrypi: ~
File Edit Tabs Help
GNU nano 2.7.4 File: /etc/owfs.conf Modified
ftp: port = 2120
##### OWSERVER #####
server: port = localhost:4304
device = /dev/i2c-1
mountpoint = /mnt/1wire
Celsius
allow_other
error_print = 0
error_level = 0
AG Get Help AO Write Out AW Where Is AK Cut Text AJ Justify AC Cur Pos
AX Exit AR Read File AL Replace AU Uncut Text AT To Spell A Go To Line

```

6. ทำการเปิดไฟล์ fuse.conf เพื่อทำการแก้ไขโดยใช้คำสั่ง `sudo nano /etc/fuse.conf` จากนั้นเอาเครื่องหมาย # หน้าข้อความ `user_allow_other` ออกจากนั้นทำการบันทึก โดยการกด CTRL+X และกด y เพื่อบันทึก

```

pi@raspberrypi: ~
File Edit Tabs Help
GNU nano 2.7.4 File: /etc/fuse.conf Modified
# /etc/fuse.conf - Configuration file for Filesystem in Userspace (FUSE)
# Set the maximum number of FUSE mounts allowed to non-root users.
# The default is 1000.
#mount_max = 1000
# Allow non-root users to specify the allow_other or allow_root mount options.
user_allow_other
AG Get Help  AO Write Out  AW Where Is  AK Cut Text  AJ Justify  AC Cur Pos
AX Exit      AR Read File  AL Replace  AU Uncut Text  AT To Spell  AA Go To Line

```

7. จากนั้นให้ทำการ Reboot บอร์ด Raspberry Pi หลังจาก Reboot เสร็จเรียบร้อยแล้วให้ทดลองพิมพ์คำสั่ง `sudo owfs` ในโปรแกรม Terminal เพื่อให้ owfs เริ่มทำงาน

```

pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi:~ $ sudo owfs
pi@raspberrypi:~ $

```

8. พิมพ์คำสั่ง `cd /mnt/1wire/` เพื่อเข้าไปที่โฟลเดอร์ 1wire จากนั้นพิมพ์คำสั่ง `ls` เพื่อดูรายละเอียดดังรูป

```

pi@raspberrypi: /mnt/1wire
File Edit Tabs Help
pi@raspberrypi:~ $ sudo owfs
pi@raspberrypi:~ $ cd /mnt/1wire/
pi@raspberrypi:/mnt/1wire $ ls
28.FF4F07C21705 bus.0 settings statistics system
alarm bus.1 simultaneous structure uncached
pi@raspberrypi:/mnt/1wire $

```

หลังจากนั้นจะเห็นโฟลเดอร์ชื่อ `28.FF4F07C21705` ซึ่งจะเป็นรหัสประจำตัวของ DS18B20 ซึ่งแต่ละตัวรหัสนี้จะไม่ซ้ำกัน

9. พิมพ์คำสั่ง `cd 28.FF4F07C21705` เพื่อเข้าไปที่โฟลเดอร์ `28.FF4F07C21705` จากนั้นพิมพ์คำสั่ง `ls` เพื่อดูรายละเอียดดังรูป

```

pi@raspberrypi: /mnt/1wire/28.FF4F07C21705
File Edit Tabs Help
pi@raspberrypi:~ $ sudo owfs
pi@raspberrypi:~ $ cd /mnt/1wire/
pi@raspberrypi:/mnt/1wire $ ls
28.FF4F07C21705 bus.0 settings statistics system
alarm bus.1 simultaneous structure uncached
pi@raspberrypi:/mnt/1wire $ cd 28.FF4F07C21705
pi@raspberrypi:/mnt/1wire/28.FF4F07C21705 $ ls
address family locator r_locator temperature11 templov
alias fasttemp power scratchpad temperature12 tempres
crc8 id r_address temperature temperature9 type
updates latesttemp r_id temperature10 temphigh
pi@raspberrypi:/mnt/1wire/28.FF4F07C21705 $

```

หลังจากนั้นจะเห็นโฟลเดอร์ต่างๆ ซึ่งจะเป็นรายละเอียดต่างๆ ของ DS18B20 ซึ่งเราสามารถที่จะอ่านค่าออกมาได้

10. ทดลองใช้คำสั่ง `cat temperature` เพื่ออ่านค่าอุณหภูมิด้รูป จากรูปจะได้ค่าอุณหภูมิคือ 26.375 องศาเซลเซียส

```

pi@raspberrypi: /mnt/1wire/28.FF4F07C21705
File Edit Tabs Help
pi@raspberrypi:~ $ sudo owfs
pi@raspberrypi:~ $ cd /mnt/1wire/
pi@raspberrypi:/mnt/1wire $ ls
28.FF4F07C21705 bus.0 settings statistics system
alarm bus.1 hwm1000000 structure uncached
pi@raspberrypi:/mnt/1wire $ cd 28.FF4F07C21705
pi@raspberrypi:/mnt/1wire/28.FF4F07C21705 $ ls
address family locator r_locator temperature11 tempow
alias fasttemp power scratchpad temperature12 tempres
crc8 id r_address temperature temperature9 type
7777 latesttemp r_id temperature10 temphigh
pi@raspberrypi:/mnt/1wire/28.FF4F07C21705 $ cat temperature
26.375pi@raspberrypi:/mnt/1wire/28.FF4F07C21705 $

```

11. จากตัวอย่างนี้ทุกครั้งที่เปิดบอร์ด Raspberry Pi จะต้องคอยมาพิมพ์คำสั่ง `sudo owfs` ทุกครั้งเพื่อให้ `owfs` ทำงาน ถ้าผู้ใช้ต้องการให้ `owfs` ทำงานอัตโนมัติเมื่อเปิดบอร์ดก็สามารถทำได้โดยใช้คำสั่ง `sudo crontab -e` ดังรูป

```

pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi:~ $ sudo crontab -e
no crontab for root - using an empty one

Select an editor. To change later, run 'select-editor'.
 1. /bin/ed
 2. /bin/nano <---- easiest
 3. /usr/bin/vim.tiny

Choose 1-3 [2]:

```

12. กดเลือกหมายเลข 2 และ ENTER จากนั้นให้พิมพ์ข้อความ `@reboot sudo -u root owfs` ด้านล่างของไฟล์เพื่อให้ `owfs` ทำงานอัตโนมัติหลังจาก Reboot จากนั้นทำการบันทึก โดยการกด CTRL+X และกด y เพื่อบันทึก

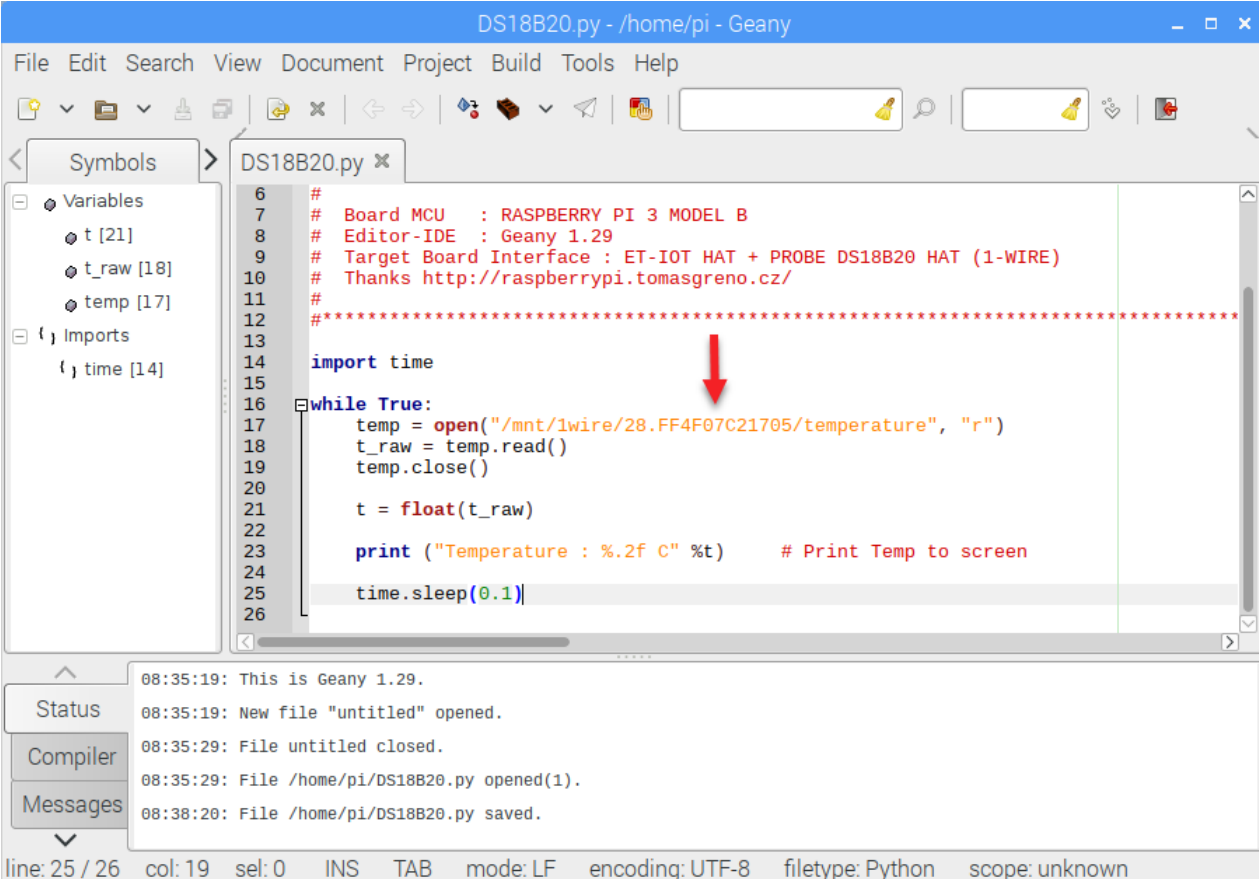
```

pi@raspberrypi: ~
File Edit Tabs Help
GNU nano 2.7.4 File: /tmp/crontab.FLersR/crontab Modified
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow command
@reboot sudo -u root owfs

```

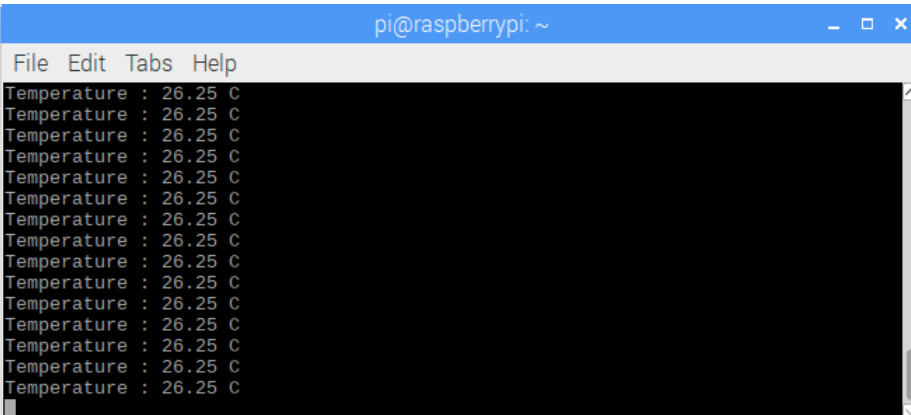


13. ทดลองเขียนโปรแกรมภาษา Python (DS18B20.py) เพื่ออ่านค่ามาแสดงที่โปรแกรม Terminal ซึ่งโปรแกรมจะมีส่วนที่ต้องแก้ไขคือรหัสประจำตัวของ DS18B20 ซึ่งแต่ละตัวจะไม่เหมือนกัน ต้องทำการแก้ไขให้ตรงกับของผู้ใช้เอง จากตัวอย่างนี้คือรหัส 28.FF4F07C21705

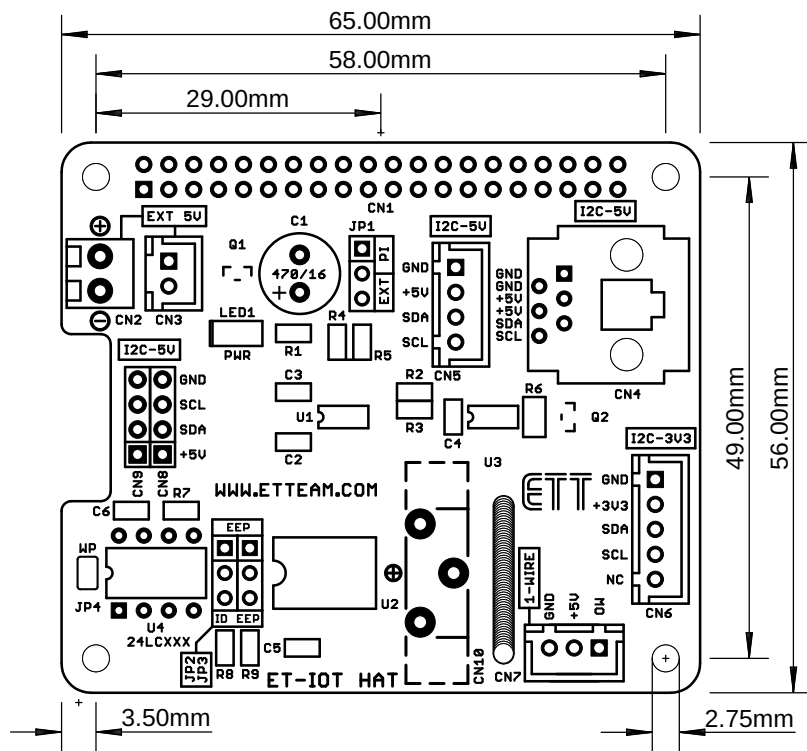


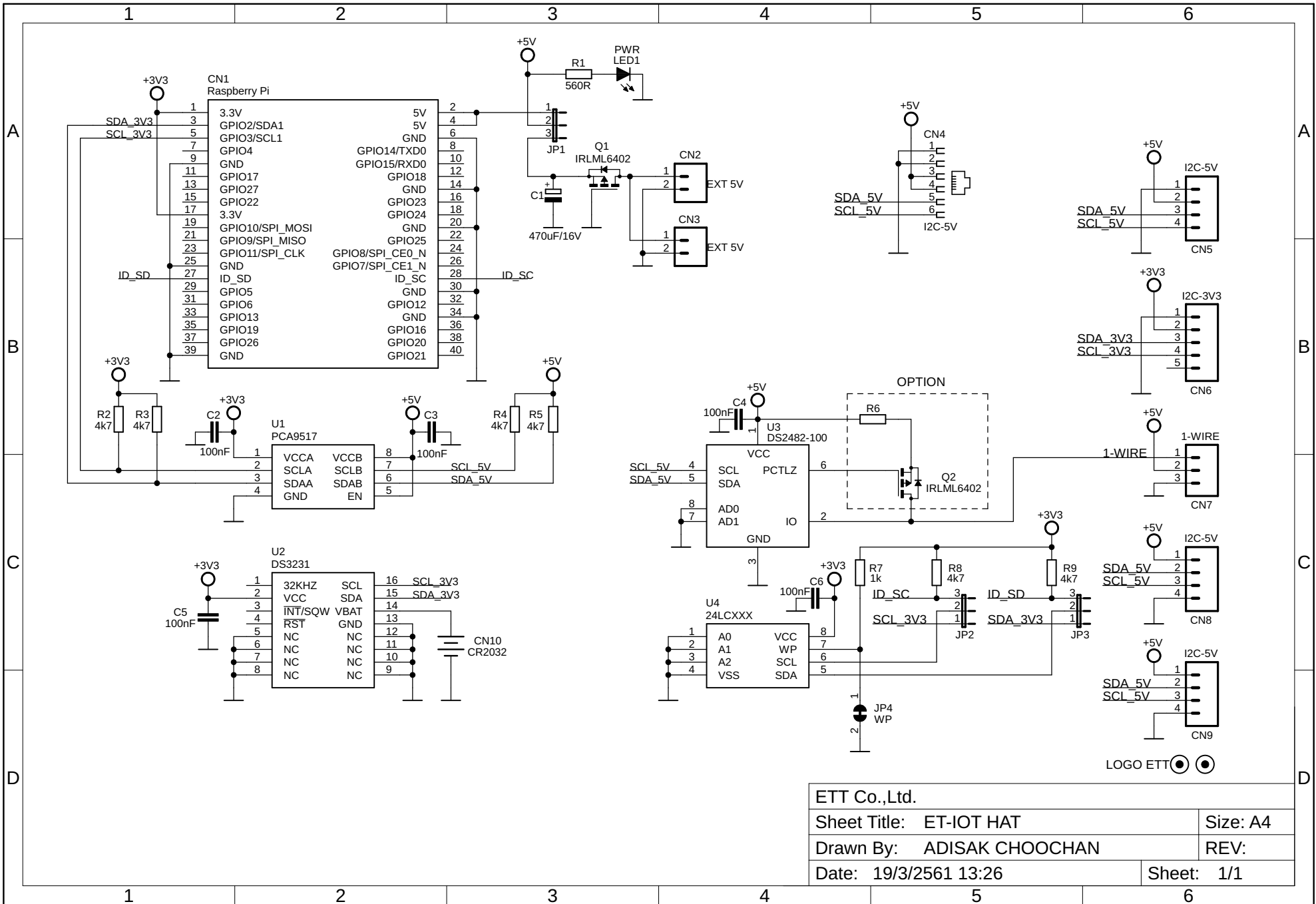
```
DS18B20.py - /home/pi - Geany
File Edit Search View Document Project Build Tools Help
Symbols
Variables
  t [21]
  t_raw [18]
  temp [17]
Imports
  time [14]
6 #
7 # Board MCU : RASPBERRY PI 3 MODEL B
8 # Editor-IDE : Geany 1.29
9 # Target Board Interface : ET-IOT HAT + PROBE DS18B20 HAT (1-WIRE)
10 # Thanks http://raspberrypi.tomasgrenoz.cz/
11 #
12 #*****
13
14 import time
15
16 while True:
17     temp = open("/mnt/1wire/28.FF4F07C21705/temperature", "r")
18     t_raw = temp.read()
19     temp.close()
20
21     t = float(t_raw)
22
23     print ("Temperature : %.2f C" %t) # Print Temp to screen
24
25     time.sleep(0.1)
26
08:35:19: This is Geany 1.29.
08:35:19: New file "untitled" opened.
08:35:29: File untitled closed.
08:35:29: File /home/pi/DS18B20.py opened(1).
08:38:20: File /home/pi/DS18B20.py saved.
line: 25 / 26 col: 19 sel: 0 INS TAB mode: LF encoding: UTF-8 filetype: Python scope: unknown
```

14. ทำการบันทึกโค้ดโปรแกรมและสั่งให้โปรแกรมทำงาน โดยอาจจะไปที่โปรแกรม Terminal แล้วใช้คำสั่ง `python3 DS18B20.py` ซึ่งจะเห็นโปรแกรมทำงานดังรูป



```
pi@raspberrypi: ~
File Edit Tabs Help
Temperature : 26.25 C
Temperature : 26.25 C
Temperature : 26.25 C
Temperature : 26.25 C
Temperature : 26.25 C
Temperature : 26.25 C
Temperature : 26.25 C
Temperature : 26.25 C
Temperature : 26.25 C
Temperature : 26.25 C
Temperature : 26.25 C
Temperature : 26.25 C
Temperature : 26.25 C
Temperature : 26.25 C
Temperature : 26.25 C
Temperature : 26.25 C
Temperature : 26.25 C
```





ETT Co.,Ltd.	
Sheet Title: ET-IOT HAT	Size: A4
Drawn By: ADISAK CHOOCHAN	REV:
Date: 19/3/2561 13:26	Sheet: 1/1

