Board ET-IOT HAT is particularly designed for using together with Board Raspberry PI by using I2C BUS Interface, so it can communicate to Board INPUT, OUTPUT or Board SENSORS in the format of I2C BUS Interface or 1-WIRE Interface effectively.

**SPECIFICATIONS OF BOARD ET-IOT HAT**

- Can be used with Board Raspberry PI by using I2C BUS Interface
- Connecting part of Circuit EEPROM (OPTION) can be used either as I2EEPROM or general EEPROM as required
- Part of Circuit RTC (DS3231) is used as Time Base of Board Raspberry, provided with BATTERY BACKUP
- Connecting position of I2C BUS can be connected to both 3.3V and 5V external devices
- Circuit I2C to 1-Wire (DS2482-100) can be connected to 1-Wire Device
- Connector External POWER SUPPLY 5VDC can provide power for the connected board; it can use power either from Board Raspberry PI or from External Board by setting JUMPER
- PCB Size: 6.5 x 5.6 cm
COMPONENTS OF BOARD ET-IOT HAT

- **No.1** Connector External POWER SUPPLY 5VDC provides power for the connected board when the energy source of Board Raspberry PI hasn’t got enough POWER SUPPLY to provide for the circuit.
- **No.2** JUMPER (JP1) chooses 5V energy source between from Board Raspberry PI (PI) or from External Board (EXT).
- **No.3** CONNECTOR I2C BUS WAFER 4PIN (CN 5) can be used with 5V System.
- **No.4** CONNECTOR I2C BUS RJ 16PIN (CN 4) can be used with 5V System.
- **No.5** CONNECTOR I2C BUS HEADER 4PIN can be used with 5V System.
- **No.6** Connecting Area of Circuit EEPROM (OPTION) can be used as either IDEEPPROM or general EEPROM as required.
- **No.7** JUMPER (JP2, JP3) chooses EEPROM Interface either to be IDEEPPROM (IDEEP Position) or general EEPROM (EEP Position).
- **No.8** BATTERY BACKUP (CR2032) is used for CIRCUIT RTC (DS3231).
- **No.9** CONNECTOR 1-WIRE WAFFER 3PIN (CN 7) can be used with 5V System.
- **No.10** CONNECTOR I2C BUS WAFER 5PIN (CN 6) can be used with 3.3V System.
HOW TO CONNECT BOARD ET-IOT HAT TO BOARD RASPBERRY PI

HOW TO SETUP VALUES OF BOARD RASPBERRY PI FOR USING WITH BOARD ET-IOT HAT

Board ET-IOT HAT uses I2C Bus Interface, so, it requires enabling the I2C BUS Interface on Board Raspberry PI first; please follow the instructions below. The example below shows how to connect Board ET-IOT HAT to Board Raspberry PI 3 Mode B and it runs by the Operating System of RASPBIAN STRETCH WITH DESKTOP Version April 2018 Release date: 2018-04-18

1. Connect Board ET-IOT HAT together with Board Raspberry PI and then provide POWER SUPPLY into Board Raspberry PI completely.
2. Choose Menu Preferences — Raspberry PI Configuration
3. Choose Tab **Interfaces**, choose **I2C** as **Enable** and then click **OK** as shown in the picture. Next, please reboot Board Raspberry Pi.

![Raspberry Pi Configuration](image1)

4. Next, open **Program Terminal** and type the Command `i2cdetect -y 1` as shown in the picture below.

![Program Terminal](image2)

Then, the Program starts scanning and looking at the actually connected I2C devices on Board ETT-IOT HAT; **DS2482-100** Address 18 and **DS3231** Address 68. If the program shows these values, it means that Board ETT-IOT HAT is ready to run.
HOW TO USE PROBE DS18B20 HAT TOGETHER WITH BOARD ET-IOT HAT

PROBE DS18B20 HAT uses Temperature Sensor “DS18B20” that is 1-Wire Interface so it requires connecting PROBE DS18B20 HAT at CONNECTOR 1-Wire WAFER 3PIN (CN7). This example shows how to write a program to communicate to 1-Wire device by using OWFS 1-Wire File System, please follow the instructions below:

1. Open **Terminal** and type the Command `sudo apt-get install owfs owshell` to install owfs and owshell.

![Image of Terminal output]

2. Type **Y** and press **ENTER** to confirm, and please wait until the installation process is complete.

![Image of Terminal output]

3. Create new folder name 1wire in /mnt by using the Command `sudo mkdir /mnt/1wire`

4. Open File owfs.conf to edit value by using the Command `sudo nano /etc/owfs.conf`, next type the number sign (#) in front of the message `server: FAKE =DS18S20,DS2405` as shown in the picture below.
5. The following messages must be typed below the file:

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

Next, please press **CTRL+X** and **Y** to save the new values.
6. Open File fuse.conf to edit by using the Command `sudo nano /etc/fuse.conf`, next, remove the number sign (#) in front of the message `user_allow_other` and then press CTRL+X and y to save the new value.

7. Reboot Board Raspberry Pi. After rebooted completely, please type the Command `sudo owfs` in Program Terminal to stimulate owfs to run.

8. Type the Command `cd /mnt/1wire` to enter the Folder name 1wire, next, type the Command `ls` to see more details as shown in the picture below.

Then, user can see the Folder name 2BF4FC0205 that is the identity code of DS18B20 and each identity code is different.
9. Type the Command `cd 28FF4F07C21705` to enter the Folder name `28FF4F07C21705`, next, type the Command `ls` to see more details as shows in the picture below.

Then user can see folders that include details of DS18B20 and it is read out.

10. Test the operation by using the Command `cat temperatur` to read temperature value; it shows the temperature value of 26.375 degrees Celsius as shown in the picture below.

11. Referred to this example, it always types the Command `sudo owfs` every time user opens Board Raspberry Pi to stimulate `owfs` to run. Please use the Command `sudo crontab -e` and uses `owfs` auto runs every time user opens the Board as shown in the picture below.
12. Choose No.2 and press ENTER. Next, type the message `@reboot sudo -u root owfs` below the file and `owfs` always auto runs after rebooted completely. Then, press `CTRL+X` and `y` to save the new value.

13. Test the operation by writing program by Python Language (DS1820.py) to read and show the value on Program Terminal. Some part of program must be edited that is identity code of DS18B20 because each value is different; so, user needs to edit the value according to user's requirement. Please refer to the example, the identity code is 28FF4F0C2105.
14. Finally, save the Code Program and run the program in this case, user may go to Program Terminal and use the Command `python3 DS18B20.py` and the program starts running as shown in the picture below;