|  |
| --- |
|  |
|  |  |

**9. Write a program to understand the use of Firebasae with Raspberry Pie to control sensors.**

**Raspberry-PI\_SHT31\_Firebase**

In this project I am going to post the real time data of temperature and humidity sensor to the Firebase database. I am using SHT31 Sensor to communicate with raspberry pi using I2C connection adapters

**Important Features:-**

The Firebase Realtime Database is a cloud-hosted database.

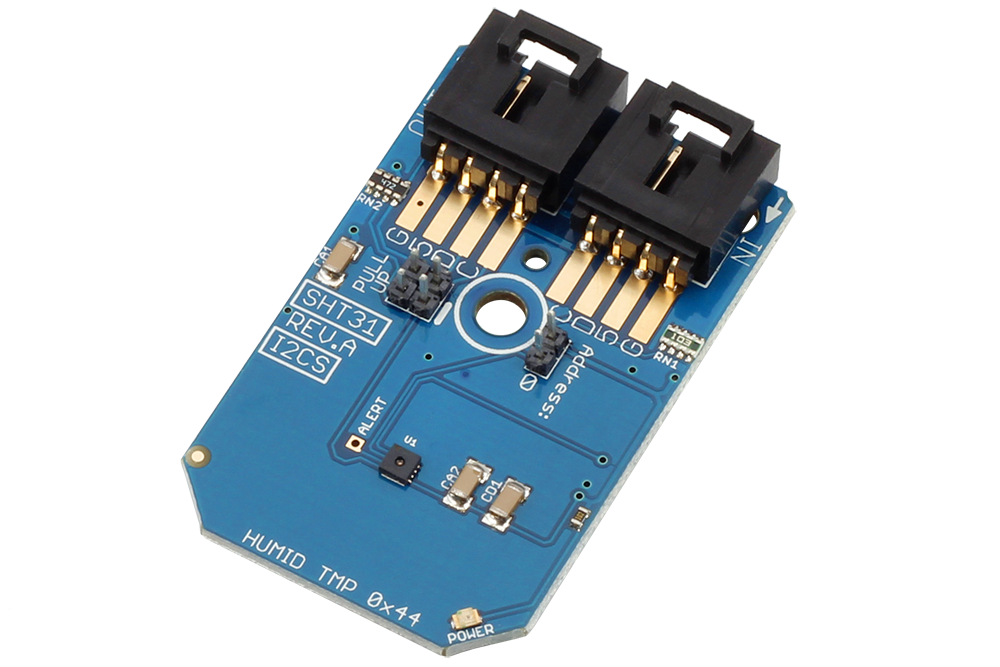
Single client sensor data is Collaborate with cross-platform apps like IOS, Android, and JavaScript SDKs as well as many IoT hardware with ease.

The 2 wired I2C protocol we are be able to extract the precised, calibrated and linear data of sensor. Using I2C connection protocol we can connect around 255 sensors at same time.

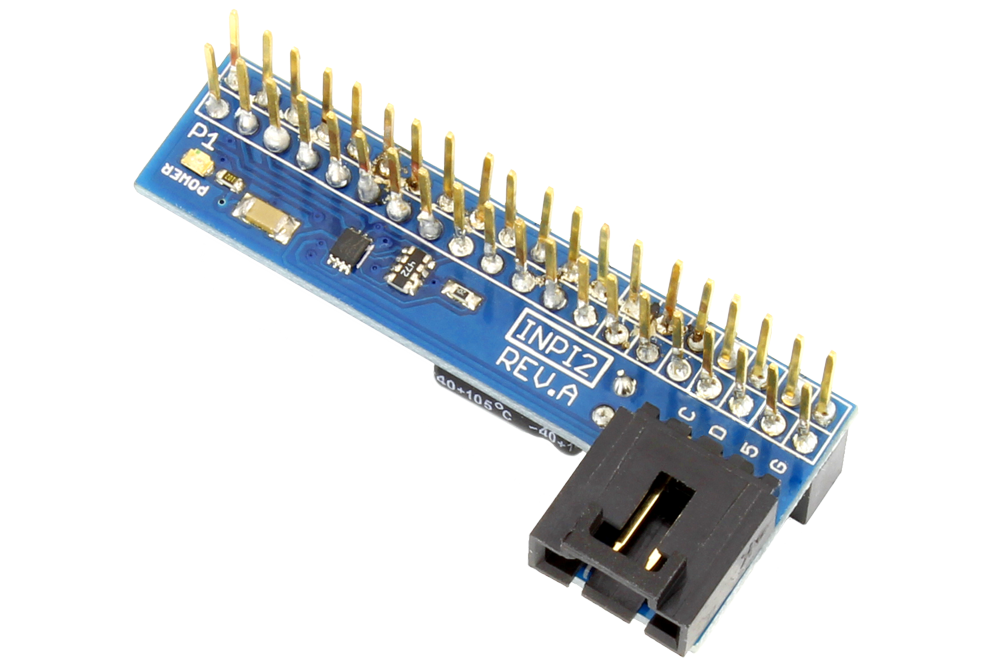
Data of sensor stores with specified key-value pairs to change the behavior and appearance of your app without requiring any download an update.

Devices Used in project

* Raspberry Pi 3 Model B ([Setup and Insatallation procedure for Newbie](https://github.com/varul29/Raspberry-PI-/blob/master/README.md))
* [SHT31](https://store.ncd.io/product/sht31-humidity-and-temperature-sensor-%C2%B12rh-%C2%B10-3c-i2c-mini-module/) I2C mini module sensor



Raspberry Pi [I2C adapter](https://store.ncd.io/product/i2c-shield-for-raspberry-pi-3-pi2-with-inward-facing-i2c-port/)



* I2C cable

***Connect the Raspberry Pi GPIO pins with I2C adapter***

***Using I2C cable, connect SHT31 sensor 'IN' port (available on sensor) with I2C adapter***

**Working Of Code:-**

We are importing the the libraries packages of I2C, Firebase in starting.

**import smbus**

**import time**

**from firebase import firebase**

Detects the pull up then initialize the I2C register, measurement commands, repeatability measurement and write the I2C data

**bus = smbus.SMBus(1)**

**bus.write\_i2c\_block\_data(0x44, 0x2C, [0x06])**

Read data from 16 Bits I2C frame of specified address registers

**data = bus.read\_i2c\_block\_data(0x44, 0x00, 6)**

Combine the 2 Bytes data (MSB & LSB available) by using shift register value

**temp = data[0] \* 256 + data[1]**

With help of SHT31 datasheet, calclate the temperature in celcius as well as in Farenhiet divided by the resolutin specified by user, subtrcted 1.

**temp = data[0] \* 256 + data[1] #shifting data[0]**

**cTemp = -45 + (175 \* temp / 65535.0)**

**fTemp = -49 + (315 \* temp / 65535.0)**

**humidity = 100 \* (data[3] \* 256 + data[4]) / 65535.0**

Finally print the data after the conversion

**print "Temperature in Celsius is : %.2f C" %cTemp**

**print "Temperature in Fahrenheit is : %.2f F" %fTemp**

**print "Relative Humidity is : %.2f %%RH" %humidity**

**Python Code:-**

#We are using Python code to send the real time data of SHT31(temperature and humidity sensor) I2C module to Firebase

#Importing I2C, Time, Firebase library in python code.

import smbus

import time

from firebase import firebase

# Get I2C bus

bus = smbus.SMBus(1) #If I2C library detects the pull up then initialize the register using sensors

# SHT31 address, 0x44(68)

# Send measurement command, 0x2C(44)

# 0x06(06) High repeatability measurement

bus.write\_i2c\_block\_data(0x44, 0x2C, [0x06])

time.sleep(0.5)

# SHT31 address, 0x44(68)

#6 bytes ->

#0,1-> 2 bytes for MSB and LSB for Temperature, 2 -> 1 byte for CRC , 3,4-> 2 byte for MSB and LSB for humidity, 5 -> 1byte for CRC.

# Temp MSB, Temp LSB, Temp CRC, Humidity MSB, Humidity LSB, Humidity CRC

# Read data back from 0x00(00), 6 bytes

data = bus.read\_i2c\_block\_data(0x44, 0x00, 6)

# Convert the data

temp = data[0] \* 256 + data[1] #shifting data[0] to left side and adding data[1] xisting in right side

cTemp = -45 + (175 \* temp / 65535.0) #formula mentioned in datasheet

fTemp = -49 + (315 \* temp / 65535.0) #formula mentioned in datasheet

humidity = 100 \* (data[3] \* 256 + data[4]) / 65535.0

# Print the readings and show it RPI OS CLI

print "Temperature in Celsius is : %.2f C" %cTemp

print "Temperature in Fahrenheit is : %.2f F" %fTemp

print "Relative Humidity is : %.2f %%RH" %humidity

time.sleep(5) #5milliseconds

#store the Host ID(provided in firebase database) in variable where you want to send the real time sensor data.

firebase= firebase.FirebaseApplication('HOST ID')

#store the readings in variable and convert it into string and using firbase.post then data will be posted to databse of firebase

result = firebase.post('Project Name',{'cTemp':str(cTemp),'ftemp':str(fTemp), 'humidity':str(humidity)})

print(result)

**Output in Firebase:-**

Output readings of SHT31 sensor in database will be appear in form of Parent, Child name and its value

temperature-sensor-bb4bd (PARENT)

{

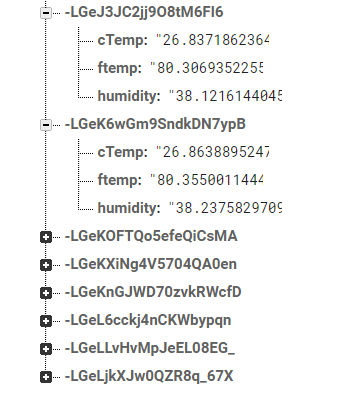
LGeJ3JC2jj9O8tM6FI6 (Secured Random Authentication Value)

cTemp: "26.8371862364" (CHILD NAME)

ftemp: "80.3069352255" (CHILD NAME)

humidity: "38.1216144045" (CHILD NAME)

}



**Post the data in Firebase**

We are syncing the real time database for JSON data. Data will be stored in form JSON object and synchronized in realtime to every connected client. For creating project for storing sensor data in console database, check the documentation

Connecting to the Host ID(provided in firebase database)

firebase= firebase.FirebaseApplication('host id mentioned in databse of firebase')

