

GoPiGo3 Cloud Data with ThingSpeak

Contents

GoPiGo3 Cloud Data with ThingSpeak	1
ThingSpeak Sensor Data Example	1
Create ThingSpeak Account	1
Setup ThingSpeak Channel	1
Create ThingSpeak Python Program	2
Upload the Sensor Data	4
What's Next?	4

ThingSpeak Sensor Data Example

<http://www.billthecomputerguy.com/gopigo>

<http://www.billthecomputerguy.com/monitoring>

Create ThingSpeak Account

ThingSpeak.com is a free cloud service that can be used to collect and display data from the GoPiGo3. You can create a maximum of 4 channels with 8 data fields per channel.

1. Go to www.thingspeak.com Create a free account.
2. Go to **My Profile**. Edit and change your time zone to your local time zone.

Setup ThingSpeak Channel

1. Logon to your **ThinkSpeak** account.
2. Click **New Channel** to create a new channel. Give it a name.
3. Field 1: **Distance Sensor** Click **Save Channel**.
4. Click the **API Keys** tab. Copy the **Write API Key**. We will use this key to upload data to this channel.

Create ThingSpeak Python Program

We are going to upload Distance Sensor Data to our ThingSpeak channel.

Create the following file to hold your **Write API Key** for the channel you are using. You can keep multiple API keys in this file, just be sure to give each one a different name.

```
1 # thingspeak_api_key.py
2 # ThingSpeak channel write api keys
3
4 THINGSPEAK_API_KEY = 'insert your api key here'
```

```

1  #!/usr/bin/env python3
2  """
3      Name:      thingspeak_distance_sensor.py
4      Author:   William A Loring
5      Created:  10/17/21 Revised:
6      Purpose:  Example of uploading data to a ThingSpeak Channel
7  """
8  # This uses the EasyGoPiGo3 library
9  # https://gopigo3.readthedocs.io/en/master/api-basic/easygopigo3.html#easygopigo3
10
11 # Import the time library for the sleep function
12 import time
13 import requests
14 from easygopigo3 import EasyGoPiGo3 # Import GoPiGo3 library
15
16 # Substitute your api key in this file for updating your ThingSpeak channel
17 import thingspeak_api_key
18 TS_KEY = thingspeak_api_key.THINGSPEAK_API_KEY
19
20 # ThingSpeak data dictionary
21 ts_data = {} # Thingspeak data dictionary
22
23 # Create an instance of the GoPiGo3 class
24 gpg = EasyGoPiGo3()
25
26 # Initialize a Distance Sensor object
27 my_distance_sensor = gpg.init_distance_sensor()
28
29
30 def main():
31     while True:
32         # =====
33         # field1: Read the distance sensor data into millimeters
34         mm = str(my_distance_sensor.read_mm())
35
36         # =====
37         # field2: Read the distance sensor data into inches
38         inches = str(my_distance_sensor.read_inches())
39
40         # Print the values of the sensor to the console for debugging
41         print("Distance Sensor Reading: " + inches + " inches " + mm + " mm")
42
43         # Send sensor data to ThingSpeak
44         thingspeak_send(mm, inches)
45
46         # 15 seconds is the minimum amount of time between uploads
47         # Sleep is set to 15 seconds for testing purposes
48         time.sleep(15)
49
50

```

```

51 def thingspeak_send(mm, inches):
52     """
53     Update the ThingSpeak channel using the requests library
54     """
55     print("Update Thingspeak Channel")
56
57     # Each field number corresponds to a field in ThingSpeak
58     params = {
59         "api_key": TS_KEY,
60         "field1": mm,
61         "field2": inches
62     }
63
64     # Update data on Thingspeak
65     ts_update = requests.get(
66         "https://api.thingspeak.com/update", params=params)
67
68     # Was the update successful?
69     if ts_update.status_code == requests.codes.ok:
70         print("Data Received!")
71     else:
72         print("Error Code: " + str(ts_update.status_code))
73
74     # Print ThngSpeak response to console
75     # ts_update.text is the thingspeak data entry number in the channel
76     print("ThingSpeak Channel Entry: " + ts_update.text)
77
78
79 # If a standalone program, call the main function
80 # Else, use as a module
81 if __name__ == '__main__':
82     main()

```

Upload the Sensor Data

Run the program. Move the GoPiGo around by hand or by running a remote control program simultaneously. Go to your ThingSpeak channel. Your data should show up almost immediately.

What's Next?

Work with other sensors to read and upload the data.