IoT PROJECT
IoT Controlled Smart Home
REPORT

Sri Aditya Deevi
B-TECH ECE(AVIONICS)
Indian Institute of Space science and Technology (IIST)

7 June, 2020

BASIC DESCRIPTION

The problem statement (basic functional description) is to build an auxillary circuit that will enable us to control a normal household bulb with the help of *Google Assistant.*

Example : If we say "*OK Google, Lights ON*" , the bulb should turn on.

COMPONENTS REQUIRED

- Arduino UNO with cable

- Male and Female Jumper Connectors

- 9V batteries

- ESP8266 WiFi Module

- Wire Straps

- Soldering Iron

- Dual Channel 5VDC Relay

- 230V bulb along with holder

- Encoder Wheel

- Clip Connectors
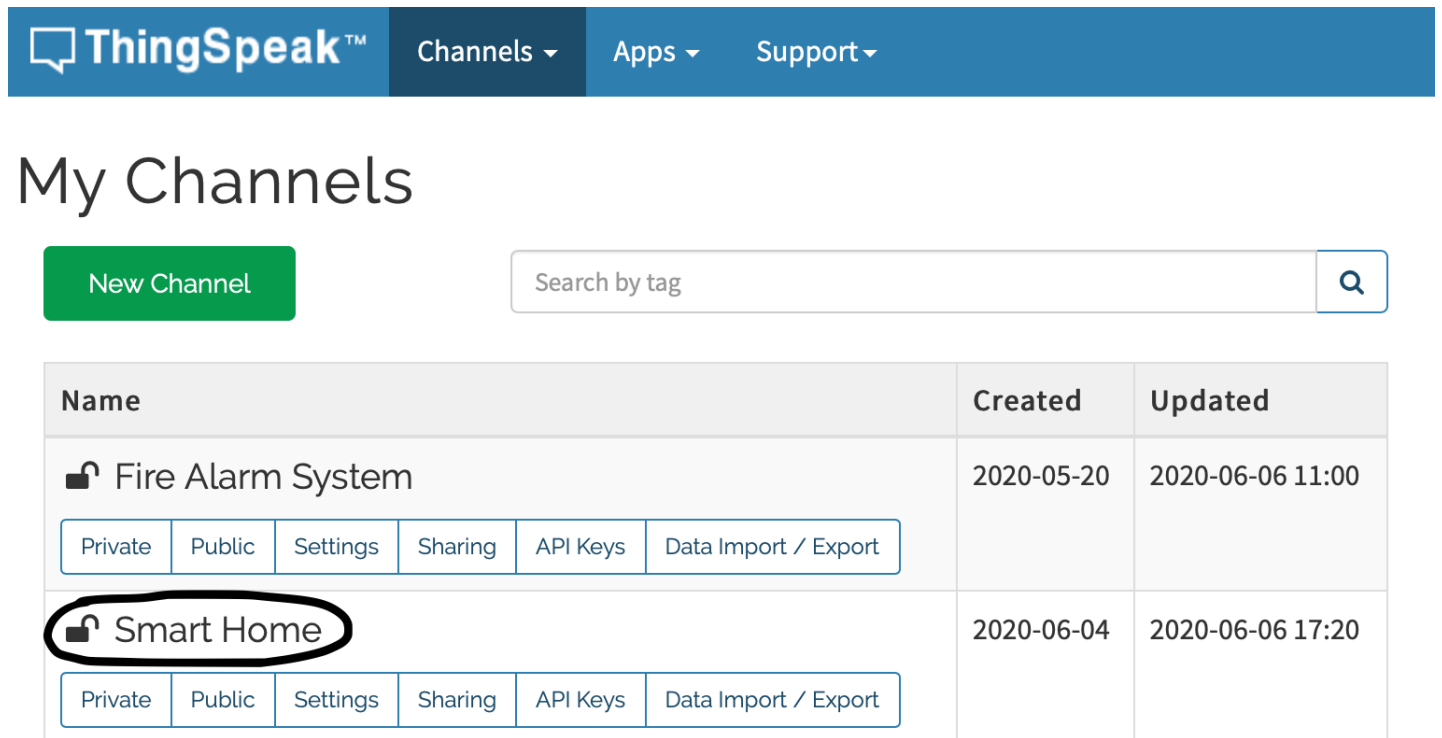
- Barrel Jack

- Mini Breadbook

- Glue Gun
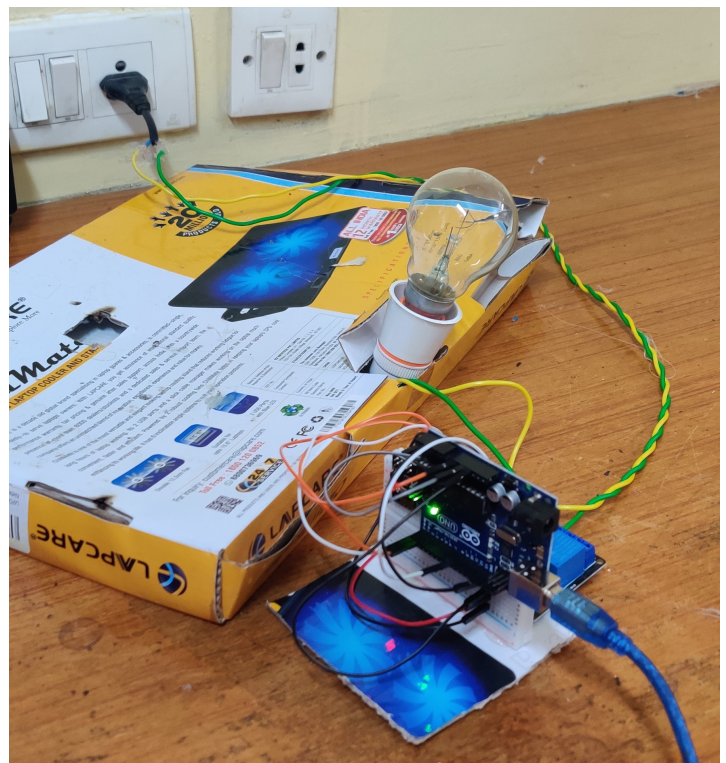
Final Model

OUTLINE OF PROCEDURE

1. A public channel in ThingSpeak was created and Field-1 was chosen to be *"Light Bulb"* for the smart bulb.

2. The read API key , write API key , Channel ID and Field number were noted for future use.

3. Two applets were created in IFTTT(If This Then That) corresponding to the Switch Off and Switch On actions (*This* : Google Assistant and *That* : WebHooks Web Request).

4. The way this works is that if a user asks Google Assistant to switch On the lights, value of 1 is written into the channel and if a user asks Google Assistant to switch Off the lights, value of 0 is written into the channel.

5. Then the circuit is made according to the schematic and the following code is uploaded.

HELPER RESOURCES

Figure 1: ThingSpeak Interface

Channel Stats

Created: 2 days ago
Last entry: about 2 hours ago
Entries: 65
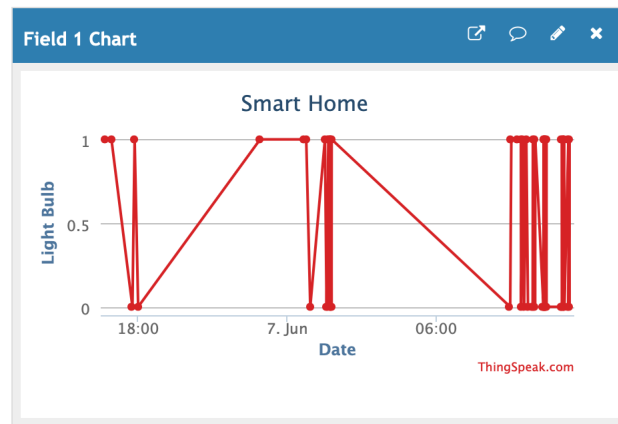
Figure 2: ThingSpeak Interface - Channel Summary

IFTTT   Home   🔍 Search

My Applets

If You say "Lights OFF", then Make a web request

by dsriaditya999

○ Connected

If You say "Light ON", then Make a web request

by dsriaditya999

○ Connected

Figure 3: IFTTT Interface

SOURCE CODE

```
1  /// Program for IoT controlled Smart Bulb
2
3  String test;
4  int receive_val;
5  char char_buf[50];
6
7  String sendCommand;
8  String Wifiname = "**********";
9  String password = "++++++++++";
10
11 int Bulb = 8;
12
13
14 void setup()
15 {
16 Serial.begin(9600);                              // Start Serial Communication
17 delay(1000);
18 Serial.println("AT");                            // AT Commands Start
19 pinMode(Bulb,OUTPUT);
20 while(!Serial.find("OK"))
21 {
22 }
23 Serial.println("AT+RST");
24 while(!Serial.find("OK"))
25 {
26 }
27 delay(1000);
28 Serial.println("AT+CWMODE=1");                   // Client Mode
29 while(!Serial.find("OK"))
30 {
31 }
32 Serial.println("AT+CWJAP=\""+Wifiname+"\",\""+password+"\"");
33 while(!Serial.find("OK"))
34 {
35 delay(6000);
36 Serial.println("AT+CWJAP=\""+Wifiname+"\",\""+password+"\"");
37 }
38
39 Serial.println("AT+CIPMUX=0");                   // Connected to only 1 Device
40 while(!Serial.find("OK")){
41 }
42 }
43
44
45 void loop()
46 {
47 sendCommand = "GET /channels/^^^^^^^^/fields/1/last.txt";        // Replace ^^^^^
      with the Public Channel's ID
48 Serial.println("AT+CIPSTART=\"TCP\",\"api.thingspeak.com\",80"); // Start TCP
      communication
49 while(!Serial.find("CONNECT"))
50 {
```

```
51 delay(5000);
52 Serial.println("AT+CIPSTART=\"TCP\",\"api.thingspeak.com\",80");
53 }
54 while(!Serial.find("OK"))
55 {
56
57 }
58
59 delay(20);
60 Serial.println("AT+CIPSEND="+String(sendCommand.length()+4));        // Sending Command's
      Length
61 while(!Serial.find("OK"))
62 {
63 delay(3000);
64 Serial.println("AT+CIPSEND="+String(sendCommand.length()+4));
65
66 }
67 while(!Serial.find(">"))
68 {
69
70 }
71 delay(100);
72 Serial.println(sendCommand);                                        // Command sent to
      retrieve last entry from the specified channel's Field-1
73 while(!Serial.find("SEND_OK"))
74 {
75 delay(3000);
76 Serial.println(sendCommand);
77
78 }
79
80 test = Serial.readString();
81 test.toCharArray(char_buf,50);
82 receive_val = int(char_buf[11]);
83 Serial.println(receive_val);                                        // Receive the ASCII
      value of the number in the last entry
84 if(receive_val == 48){                                              // ASCII Value of "0"
      is 48 (Relay is Active LOW)
85 digitalWrite(Bulb,HIGH);
86 }
87 else if(receive_val == 49){                                         // ASCII Value of "1" is
       49 (Relay is Active LOW)
88 digitalWrite(Bulb,LOW);
89 }
90
91
92 delay(2000);
93
94
95
96 }
```

SCHEMATIC

Connect Rx and Tx after uploading Code

Arduino UNO

Keyes_SRly

AC POWER SOURCE

9v