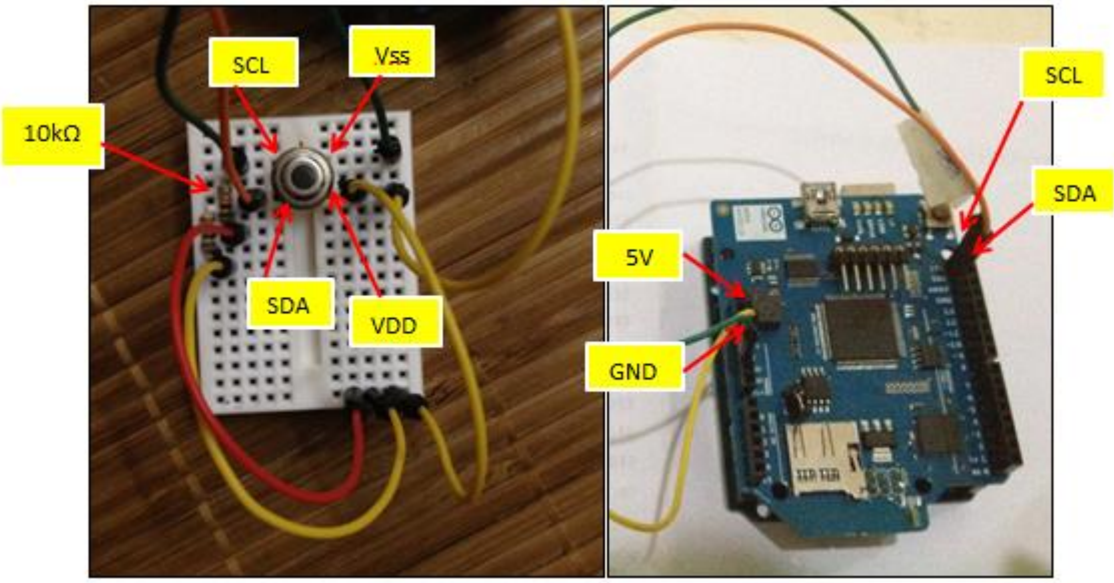
Hardware

1. Setup according the picture below.



2. However, MAKE SURE Wi-Fi shield firmware is already updated, if not follow below guide to install firmware:

Upgrading the firmware on the devices is a two steps process:

- First part is the HDG104 firmware which named "wifi_dnld.elf". The H&D module doesn't have static memory so you'll upload its firmware to AT32UC3 controller, then the AT32UC3 will transfer the firmware into the HDG104 module's dedicated flash memory.
- Second part is the firmware for the HDG104 is uploaded then it is ready to upload the Wi-Fi shield firmware for the AT32UC3. The "wifiHD.elf" is the file that contains the application for the controller.

Part 1

Download the DFU programmer depends on what operating system use by the user from http://www.atmel.com/tools/FLIP.aspx. Once done, the updated firmware is needed which located with Arduino software in the program files(x86)/hardware/avr/arduino/firmwares/wifishield folder.

Part 2

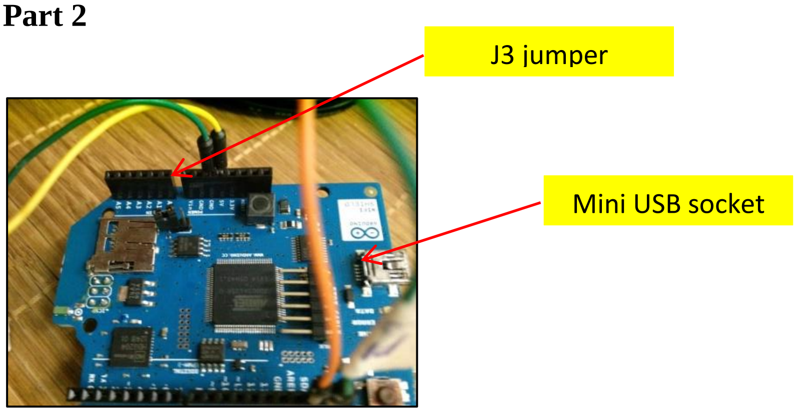


Figure 4.6: *Important components on updating Wi-Fi firmware*

The Flip software which is provided by Atmel is purposely for program the device using DFU mode (Device Firmware Update). Flip supplies a utility called batch that will be used to make the upgrade. Once Flip installed, CMD or command prompt is open and reach the following path by typing the installation directory cd C:\Program Files (x86)\Atmel\Flip 3.4.5\bin depends on location of the software. The J3 jumper is connected by putting the shield in programming mode then plug in mini USB socket. On the CMD type:

```
batchisp.exe -device AT32UC3A1256 -hardware usb -operation erase f
memory            flash            blankcheck            loadbuffer
/Arduino/hardware/avr/arduino/firmwares/wifishield/binary/wifi_dnld
.ef    program verify start reset 0
```

The HDG104 Wi-Fi module firmware can be downloaded inside the data flash

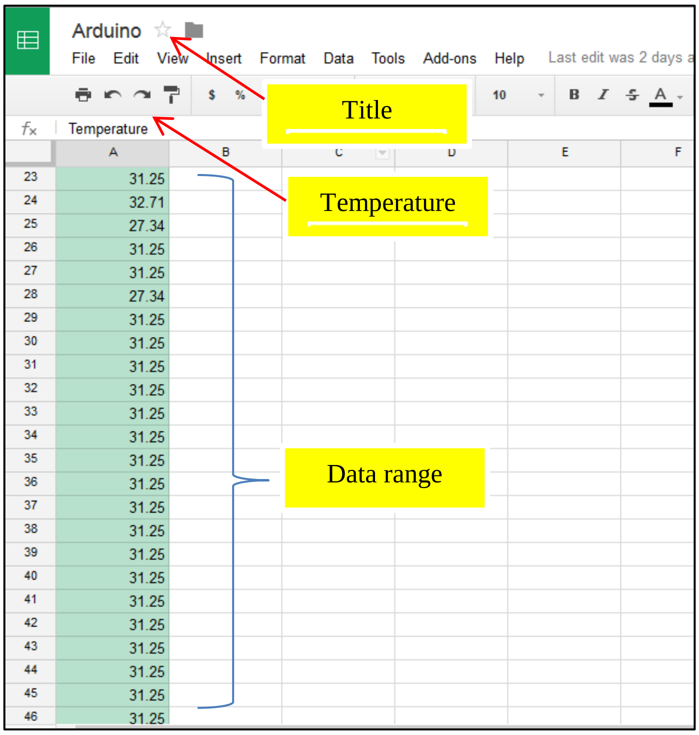
using this command:

```
batchisp.exe -device AT32UC3A1256 -hardware usb -operation erase f
memoryflashblankcheckloadbuffer/Arduino/hardware/avr/arduino/firmwa
res/wifishield/binary/wifiHD.elfprogram verify start reset 0
```

Once the upgrade is done the J3 jumper can be remove for normal mode and successful updating will prompt with a blue led light on Wi-Fi shield then restart the shield and now can start programmed inside the microprocessor.

Google Drive –Google Sheets

1. Create new Google sheet
2. To create chart automatically follow below instruction:
3. Create new Google Sheets inside Google Drive.
4. Create all important part as label below figure showing on setup for spreadsheet. Title and column name as shown is important which the information fill in the Temboo server. The data range is only updated if the sensor starting to uploading data to Google Drive. This all data inside will be used for generate a graph and temperature textview to the android application.

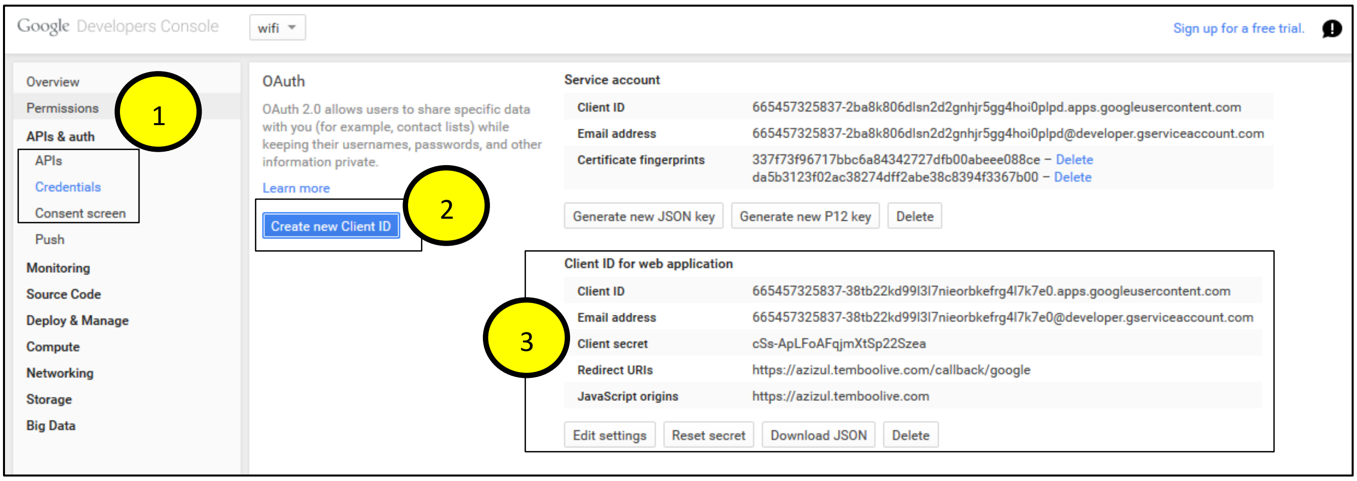


fx Temperature

	A
23	31.25
24	32.71
25	27.34
26	31.25
27	31.25
28	27.34
29	31.25
30	31.25
31	31.25
32	31.25
33	31.25
34	31.25
35	31.25
36	31.25
37	31.25
38	31.25
39	31.25
40	31.25
41	31.25
42	31.25
43	31.25
44	31.25
45	31.25
46	31.25

Figure 4.16: *Setup new Google sheet inside the Google Drive*

5. As we are using platform for storage for communication between Google and hardware, it is required to have the Google Developer Console active. All Google API settings only accessible inside the developer console and the Google Drive API must be enable before proceed any process. Below shows steps with numbering that must be follows accordingly to complete the Google authentication part. First step, Consent screen tab, the email address and product name fields must be fill out by user before proceed create the Client ID. Step two need to configure by creating new Client ID which is a web application by specify the callback URL as the Authorized Redirect URL into https://azizul.temboolive.com/callback/google. The result will produce Client ID and Client Secret that will be useful later.



OAuth

OAuth 2.0 allows users to share specific data with you (for example, contact lists) while keeping their usernames, passwords, and other information private.

Service account

Client ID	665457325837-2ba8k806dlsn2d2gnhjr5gg4hoi0plpd.apps.googleusercontent.com
Email address	665457325837-2ba8k806dlsn2d2gnhjr5gg4hoi0plpd@developer.gserviceaccount.com
Certificate fingerprints	337f73f96717bbc6a84342727dfb00abeee088ce – Delete da5b3123f02ac38274dff2abe38c8394f3367b00 – Delete

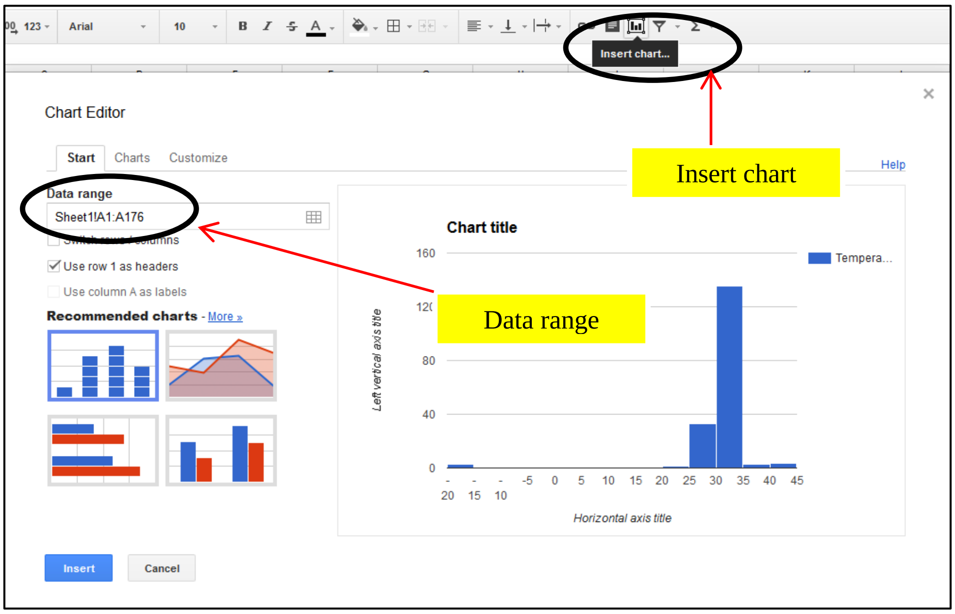
Client ID for web application

Client ID	665457325837-38tb22kd99l3l7nieorbkefrg4l7k7e0.apps.googleusercontent.com
Email address	665457325837-38tb22kd99l3l7nieorbkefrg4l7k7e0@developer.gserviceaccount.com
Client secret	cSs-ApLFoAFqjmXtSp22Szea
Redirect URIs	https://azizul.temboolive.com/callback/google
JavaScript origins	https://azizul.temboolive.com

Figure 4.17: *Steps on creating consent screen and new Client ID in developer console.*

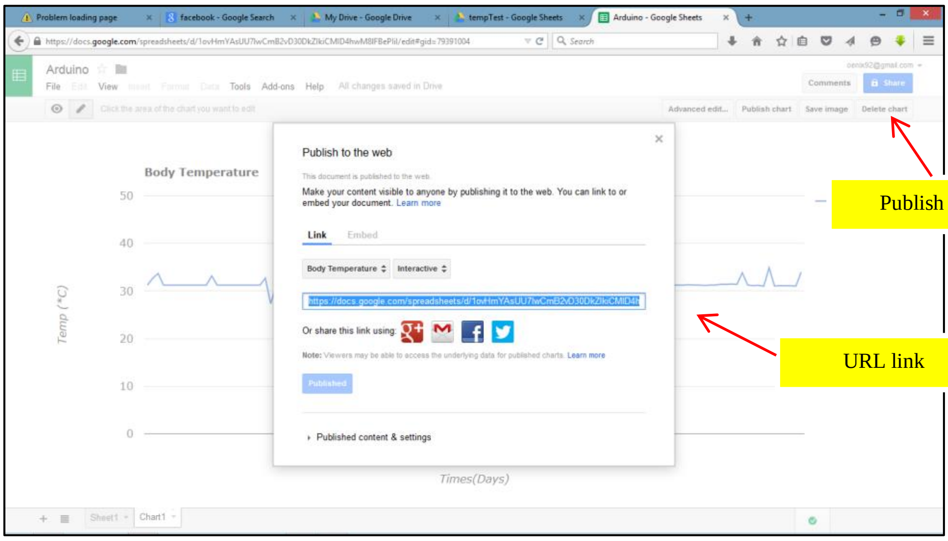
6. To publish chart by getting URL into webview Android application, follow this steps:
7. Here are all steps needed to publish the chart from user Google Drive into web. Before proceed on publishing, the chart must be create based on the column and row data

inside the Google spreadsheet as shown in Figure 4.9. The data range must be add accordingly on how many data need to updated inside the chart and user can pick type of chart using recommended chart as shown below.

8. 

9. **Figure 4.9**: *Panel insert chart Google spreadsheet based on data range.*

10. After this step completed, the publishing part as shown in Figure 4.10 can be proceed. This will provide URL link that will be insert into the java programming part in Android studio as the webview layout explained in programming part. However, due to the certain cloud security issues, preventing from abusing activity updating only allow for every 5 minutes.



Publish to the web

This document is published to the web.

Make your content visible to anyone by publishing it to the web. You can link to or embed your document. Learn more

Link Embed

Body Temperature ⇕ Interactive ⇕

https://docs.google.com/spreadsheets/d/1ovHmYAsUU7lwCmB2vD30DkZlkiCMID4h

Or share this link using:

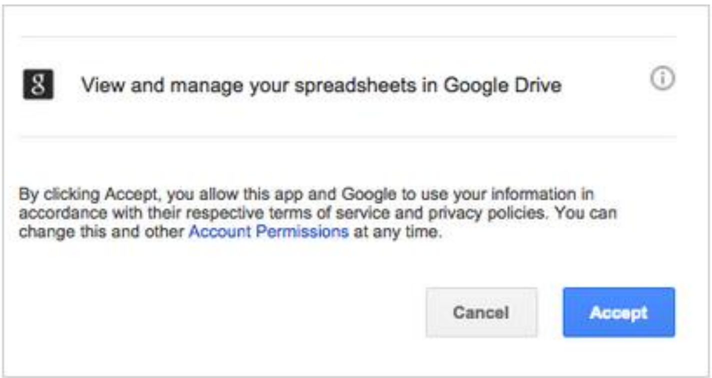
Note: Viewers may be able to access the underlying data for published charts. Learn more

Published

▸ Published content & settings

Cloud server – Temboo

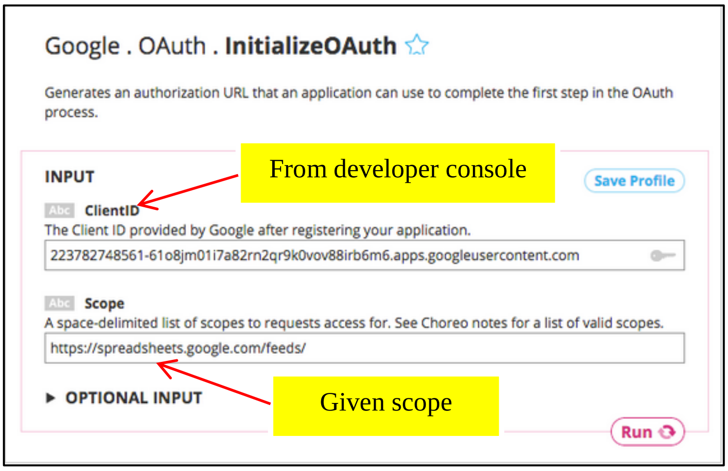
1. The next step for authentication purpose the user must undergo OAuth choreos which are initializeOAuth and finalizeOAuth. InitializeOuth is a choreos generates an authorization URL that an application can use to complete the first step in the OAuth process. A unique and private callback URL for your account is registered with Temboo so that when redirect your user to the AuthorizationURL returned by the choreo, the authorization code generated by Google is stored temporarily for easy retrieval in the finalization step of the OAuth process. OAuth process can be complete by passing the CallbackID (returned by this Choreo), to the FinalizeOAuth Choreo. It also resulting URL link for click and redirect to grant account permission. FinalizeOAuth is a Choreo completes the OAuth process by retrieving an access token for a user. The Choreo uses the CallbackID input to retrieve the callback data that Temboo collects and when the app's user authorizes the request (in this case, an authorization code). Then, the Choreo makes a request to retrieve an access token or refresh token from Google. With the refresh token given, it enables the choreo appendrow to take place.



View and manage your spreadsheets in Google Drive

By clicking Accept, you allow this app and Google to use your information in accordance with their respective terms of service and privacy policies. You can change this and other Account Permissions at any time.

Figure 4.18: *Granting permission on using Google account.*



Google . OAuth . **InitializeOAuth**

Generates an authorization URL that an application can use to complete the first step in the OAuth process.

INPUT

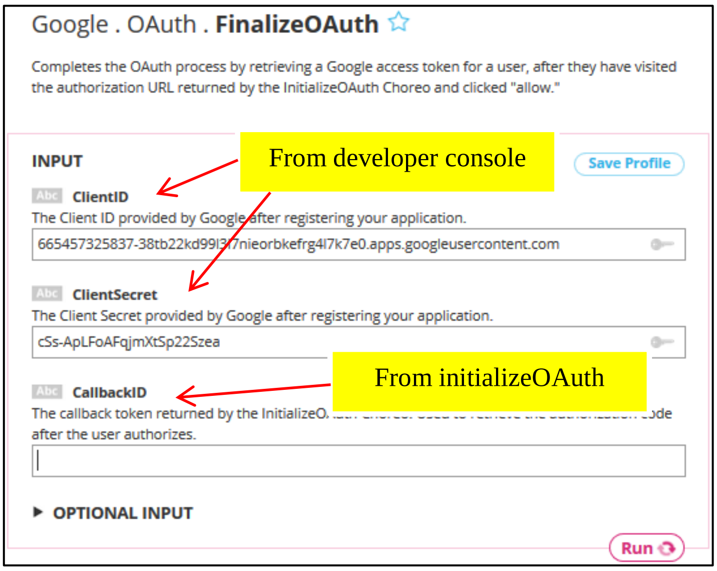
ClientID
The Client ID provided by Google after registering your application.
223782748561-61o8jm01i7a82rn2qr9k0vov88irb6m6.apps.googleusercontent.com

Scope
A space-delimited list of scopes to requests access for. See Choreo notes for a list of valid scopes.
https://spreadsheets.google.com/feeds/

From developer console

Given scope

Figure 4.19: *Steps on creating initializeOAuth.*



Google . OAuth . **FinalizeOAuth**

Completes the OAuth process by retrieving a Google access token for a user, after they have visited the authorization URL returned by the InitializeOAuth Choreo and clicked "allow."

INPUT

ClientID
The Client ID provided by Google after registering your application.
665457325837-38tb22kd99l3l7nieorbkefrg4l7k7e0.apps.googleusercontent.com

ClientSecret
The Client Secret provided by Google after registering your application.
cSs-ApLFoAFqjmXtSp22Szea

CallbackID
The callback token returned by the InitializeO ode after the user authorizes.

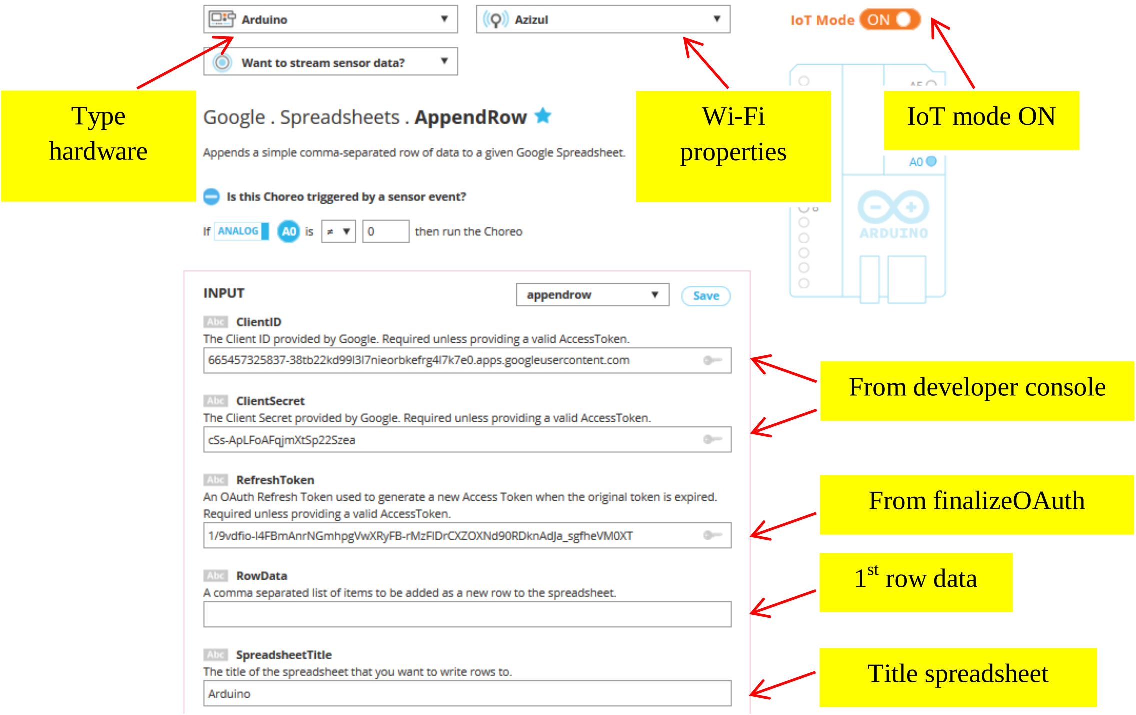
From developer console

From initializeOAuth

Figure 4.20: *Steps on creating finalizeOAuth.*

2. This step is the most important part whereby it will upload the data automatically into the Google Spreadsheet created as shown in figure 4.16 using appendrow choreo.

   Make sure appendrow choreo is in IoT mode and the type of hardware with Wi-Fi properties must be add accordingly to prevent any error while running the program inside Arduino.



Arduino ▼

Want to stream sensor data? ▼

Azizul ▼

Google . Spreadsheets . **AppendRow** ★

Appends a simple comma-separated row of data to a given Google Spreadsheet.

Is this Choreo triggered by a sensor event?

If ANALOG A0 is ≠ ▼ 0 then run the Choreo

INPUT appendrow ▼ Save

ClientID
The Client ID provided by Google. Required unless providing a valid AccessToken.
665457325837-38tb22kd99l3l7nieorbkefrg4l7k7e0.apps.googleusercontent.com

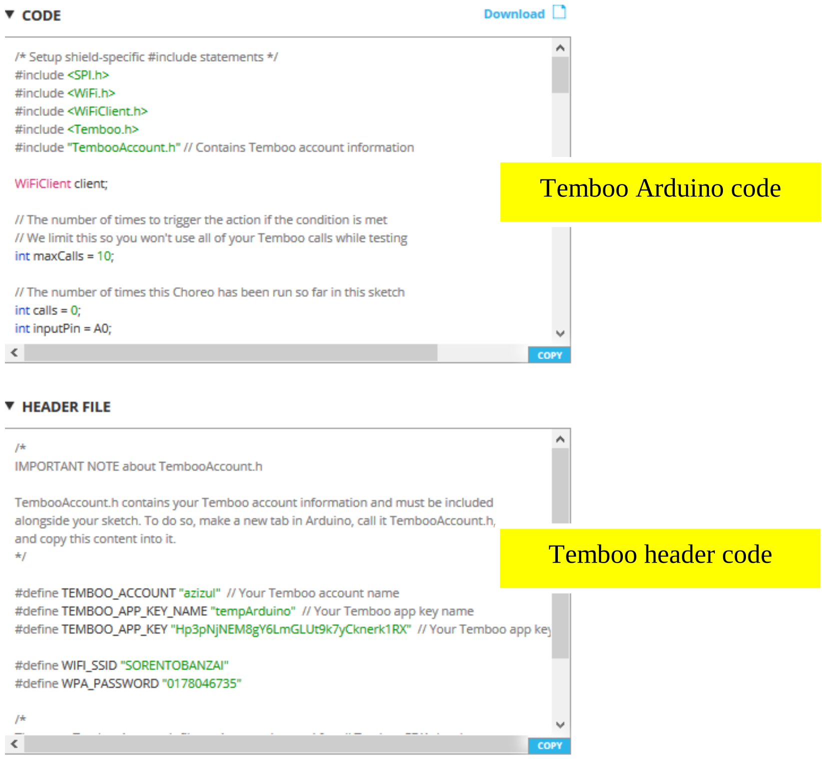
ClientSecret
The Client Secret provided by Google. Required unless providing a valid AccessToken.
cSs-ApLFoAFqjmXtSp22Szea

RefreshToken
An OAuth Refresh Token used to generate a new Access Token when the original token is expired. Required unless providing a valid AccessToken.
1/9vdfio-l4FBmAnrNGmhpgVwXRyFB-rMzFlDrCXZOXNd90RDknAdJa_sgfheVM0XT

RowData
A comma separated list of items to be added as a new row to the spreadsheet.

SpreadsheetTitle
The title of the spreadsheet that you want to write rows to.
Arduino

Figure 4.21: *Steps on generate code appendrow choreo*



▼ **CODE**

```
/* Setup shield-specific #include statements */
#include <SPI.h>
#include <WiFi.h>
#include <WiFiClient.h>
#include <Temboo.h>
#include "TembooAccount.h" // Contains Temboo account information

WiFiClient client;

// The number of times to trigger the action if the condition is met
// We limit this so you won't use all of your Temboo calls while testing
int maxCalls = 10;

// The number of times this Choreo has been run so far in this sketch
int calls = 0;
int inputPin = A0;
```

▼ **HEADER FILE**

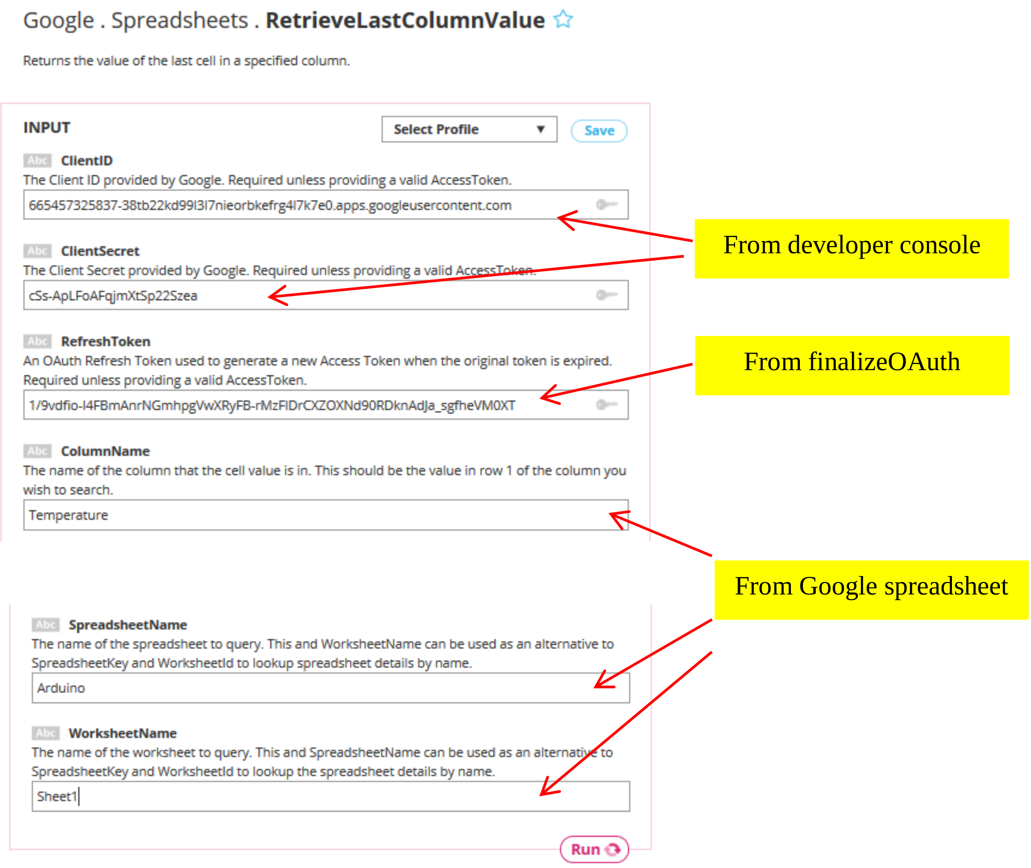
```
/*
IMPORTANT NOTE about TembooAccount.h

TembooAccount.h contains your Temboo account information and must be included
alongside your sketch. To do so, make a new tab in Arduino, call it TembooAccount.h,
and copy this content into it.
*/

#define TEMBOO_ACCOUNT "azizul"  // Your Temboo account name
#define TEMBOO_APP_KEY_NAME "tempArduino"  // Your Temboo app key name
#define TEMBOO_APP_KEY "Hp3pNjNEM8gY6LmGLUt9k7yCknerk1RX"  // Your Temboo app key

#define WIFI_SSID "SORENTOBANZAI"
#define WPA_PASSWORD "0178046735"

/*
```

Figure 4.22: *Automated code generated by Temboo*

3. After completed all steps, the program and header code generated is put inside the setup and loop function combine with the temperature sensor coding. With

this process, it should be successfully uploaded temperature data from MLX90614 to the Google Drive without any problem.

4. Figure 4.23 showing the all steps needed to complete the retreiveLastColumnvalue choreo. This choreo work by retrieving value uploaded inside the Google Drive and showing only the attest value into the Android application. The code generated in java programming and will be paste inside main activity Android application for textview of the current temperature monitored in real time.



Google . Spreadsheets . **RetrieveLastColumnValue**

Returns the value of the last cell in a specified column.

INPUT

ClientID
The Client ID provided by Google. Required unless providing a valid AccessToken.
665457325837-38tb22kd99l3l7nieorbkefrg4l7k7e0.apps.googleusercontent.com

ClientSecret
The Client Secret provided by Google. Required unless providing a valid AccessToken.
cSs-ApLFoAFqjmXtSp22Szea

RefreshToken
An OAuth Refresh Token used to generate a new Access Token when the original token is expired. Required unless providing a valid AccessToken.
1/9vdfio-l4FBmAnrNGmhpgVwXRyFB-rMzFlDrCXZOXNd90RDknAdJa_sgfheVM0XT

ColumnName
The name of the column that the cell value is in. This should be the value in row 1 of the column you wish to search.
Temperature

SpreadsheetName
The name of the spreadsheet to query. This and WorksheetName can be used as an alternative to SpreadsheetKey and WorksheetId to lookup spreadsheet details by name.
Arduino

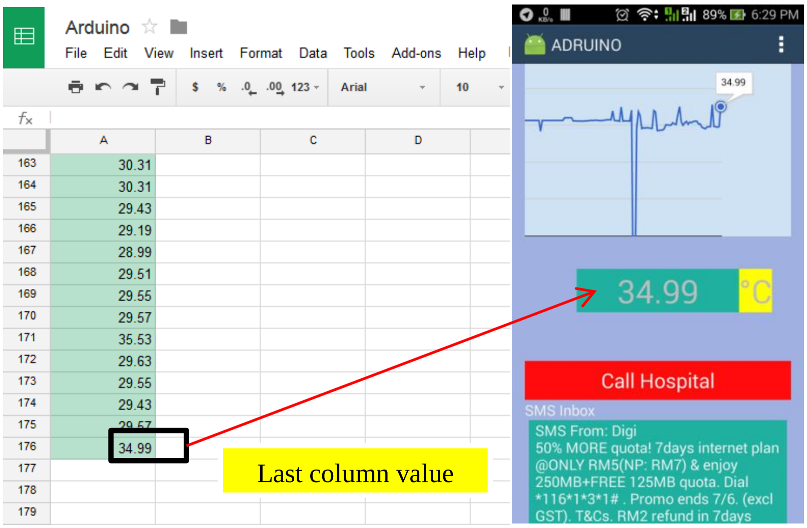
WorksheetName
The name of the worksheet to query. This and SpreadsheetName can be used as an alternative to SpreadsheetKey and WorksheetId to lookup the spreadsheet details by name.
Sheet1

From developer console

From finalizeOAuth

From Google spreadsheet

Figure 4.23: *Step on RetreiveLastColumnvalue choreo.*



	A
163	30.31
164	30.31
165	29.43
166	29.19
167	28.99
168	29.51
169	29.55
170	29.57
171	35.53
172	29.63
173	29.55
174	29.43
175	29.57
176	34.99
177	
178	
179	

89% 6:29 PM

34.99 °C

Call Hospital

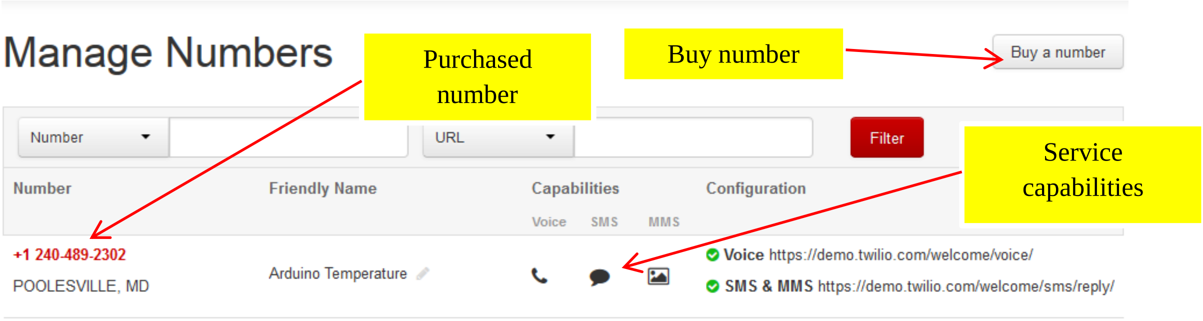
SMS Inbox

SMS From: Digi
50% MORE quota! 7days internet plan @ONLY RM5(NP: RM7) & enjoy 250MB+FREE 125MB quota. Dial *116*1*3*1# . Promo ends 7/6. (excl GST). T&Cs. RM2 refund in 7days

5. **Figure 4.24**: *Value monitored in real time by Android application*

6. The alert which is sending SMS directly to user mobile phones based on this API Twillio.

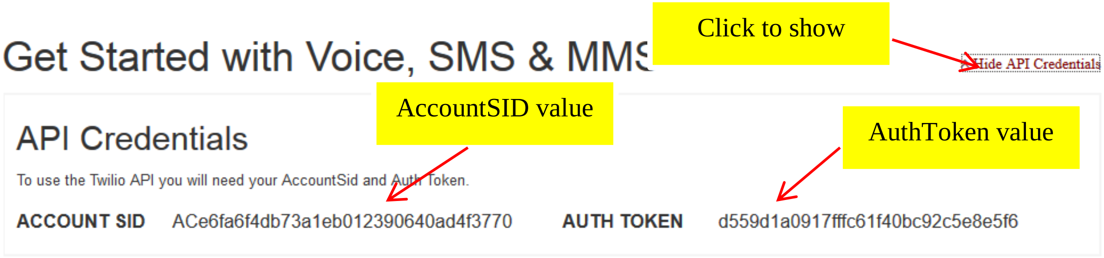
7. The user must buy a number as shown below; this purchased number must be containing capabilities on sending SMS.



Manage Numbers

Number	Friendly Name	Capabilities Voice SMS MMS	Configuration
+1 240-489-2302 POOLESVILLE, MD	Arduino Temperature	☎ ● ▨	✓ Voice https://demo.twilio.com/welcome/voice/ ✓ SMS & MMS https://demo.twilio.com/welcome/sms/reply/

Figure 4.25: *Purchasing a number with SMS capabilities*



Get Started with Voice, SMS & MMS

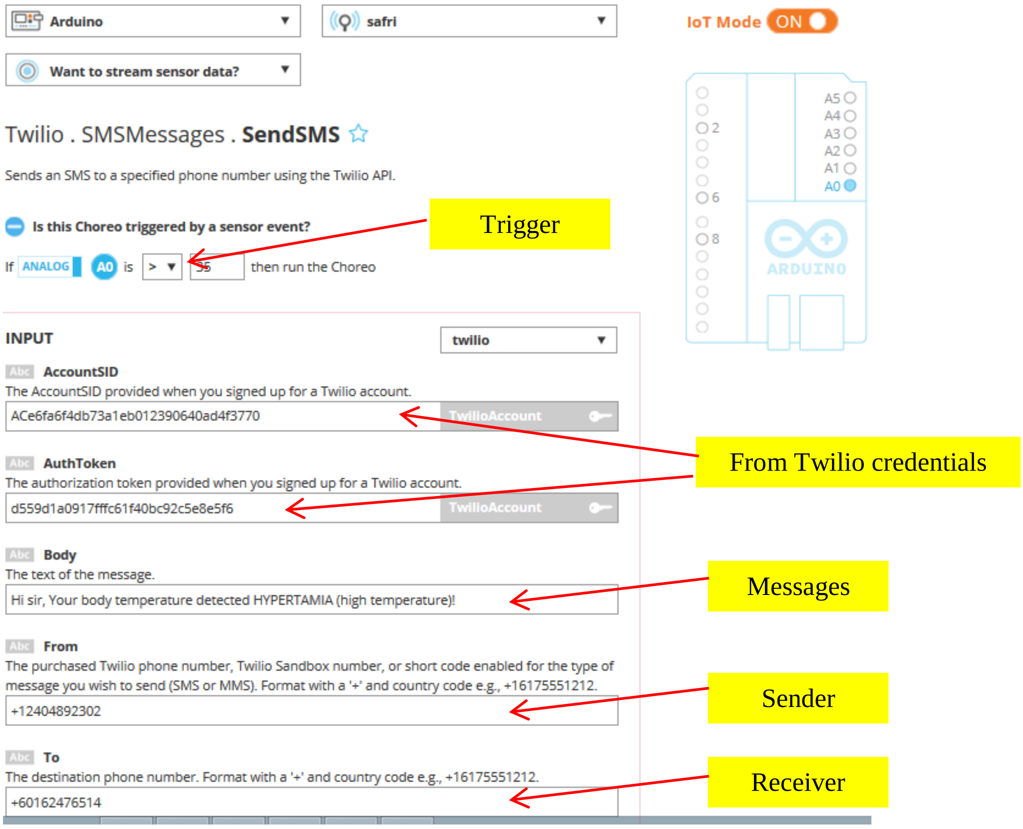
API Credentials

To use the Twilio API you will need your AccountSid and Auth Token.

ACCOUNT SID ACe6fa6f4db73a1eb012390640ad4f3770 AUTH TOKEN d559d1a0917fffc61f40bc92c5e8e5f6

Figure 4.26: *API credentials to fill out Temboo choreos.*

8. After completing all steps in Twilio webpage, the choreo can be used by filling all requirements as below figure 4.27. This send SMS choreo is triggered by sensor event. This triggered value is set as the temperature limit and using the triggered value the event will occur by calling Twilio choreo using Temboo cloud server. With this resulting Arduino code and header file insert into Arduino IDE or compiler. The receiver referring to use mobile phones while sender is based on purchased number.



Arduino ▼ safri ▼

Want to stream sensor data? ▼

IoT Mode ON

Twilio . SMSMessages . **SendSMS** ☆

Sends an SMS to a specified phone number using the Twilio API.

Is this Choreo triggered by a sensor event?

If ANALOG A0 is > ▼ 35 then run the Choreo

Trigger

INPUT twilio ▼

AccountSID
The AccountSID provided when you signed up for a Twilio account.
ACe6fa6f4db73a1eb012390640ad4f3770 TwilioAccount

AuthToken
The authorization token provided when you signed up for a Twilio account.
d559d1a0917fffc61f40bc92c5e8e5f6 TwilioAccount

From Twilio credentials

Body
The text of the message.
Hi sir, Your body temperature detected HYPERTAMIA (high temperature)!

Messages

From
The purchased Twilio phone number, Twilio Sandbox number, or short code enabled for the type of message you wish to send (SMS or MMS). Format with a '+' and country code e.g., +16175551212.
+12404892302

Sender

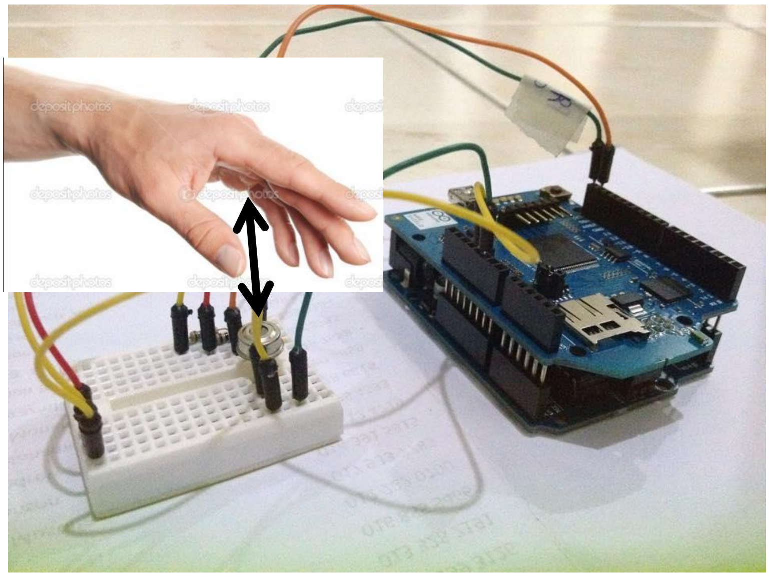
To
The destination phone number. Format with a '+' and country code e.g., +16175551212.
+60162476514

Receiver

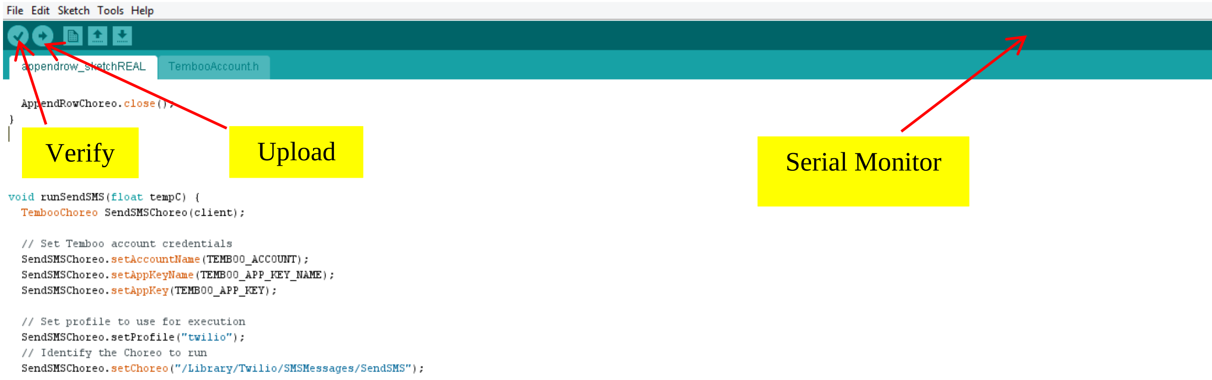
Figure 4.27: *Temboo Twilio SMS Messages choreo.*

Result and Testing guide

1. After the hardware circuit connected, clicking verify button will debug the codes inside Arduino IDE while clicking upload button will uploading the process execution into the microprocessor Arduino R3 and Wi-Fi shield as shown below figure 4.31. Clicking the serial monitor button will showing the result on shield and Wi-Fi shield connection by showing OK or FAIL and temperature reading as shown in figure 4.32.



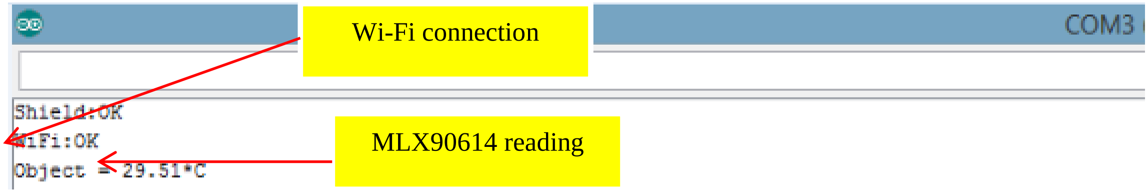
2. **Figure 4.36**: *Hand put on top without contact with sensor*



```
  AppendRowChoreo.close();
}

void runSendSMS(float tempC) {
  TembooChoreo SendSMSChoreo(client);

  // Set Temboo account credentials
  SendSMSChoreo.setAccountName(TEMBOO_ACCOUNT);
  SendSMSChoreo.setAppKeyName(TEMBOO_APP_KEY_NAME);
  SendSMSChoreo.setAppKey(TEMBOO_APP_KEY);

  // Set profile to use for execution
  SendSMSChoreo.setProfile("twilio");
  // Identify the Choreo to run
  SendSMSChoreo.setChoreo("/Library/Twilio/SMSMessages/SendSMS");
```

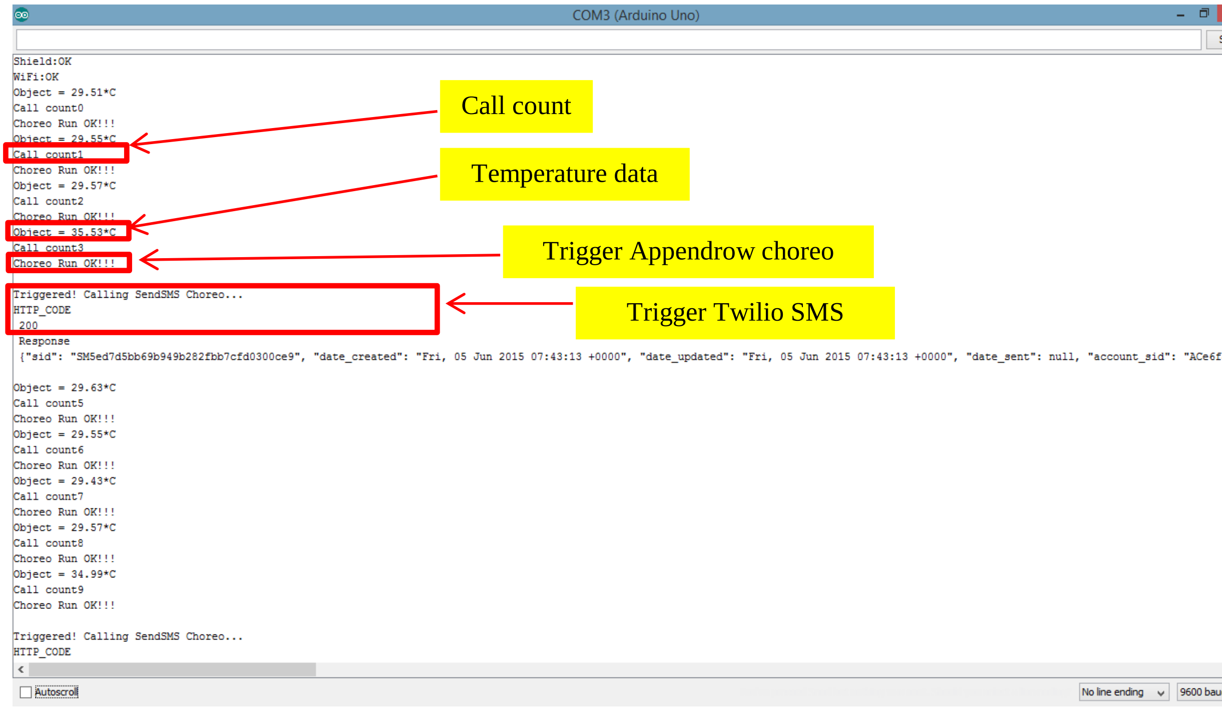
3.

4. **Figure 4.31**: *Verify, upload and serial monitor buttons inside Arduino IDE.*



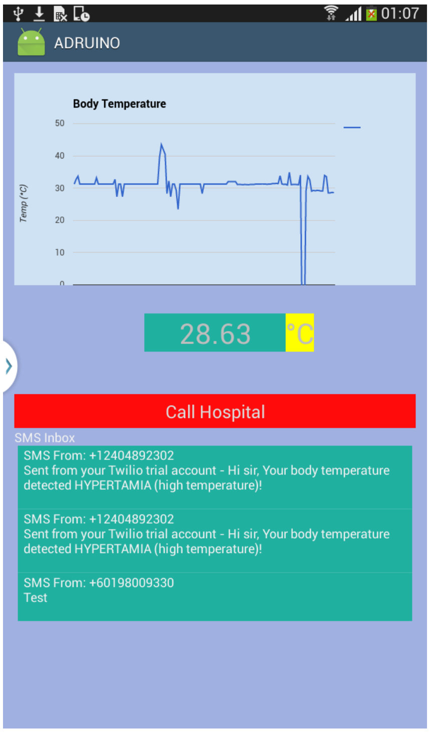
```
Shield:OK
WiFi:OK
Object = 29.51*C
```

5. Clicking the serial monitor button will pop out as below figure



```
Shield:OK
WiFi:OK
Object = 29.51*C
Call count0
Choreo Run OK!!!
Object = 29.55*C
Call count1
Choreo Run OK!!!
Object = 29.57*C
Call count2
Choreo Run OK!!!
Object = 35.53*C
Call count3
Choreo Run OK!!!

Triggered! Calling SendSMS Choreo...
HTTP_CODE
 200
 Response
 {"sid": "SM5ed7d5bb69b949b282fbb7cfd0300ce9", "date_created": "Fri, 05 Jun 2015 07:43:13 +0000", "date_updated": "Fri, 05 Jun 2015 07:43:13 +0000", "date_sent": null, "account_sid": "ACe6f

Object = 29.63*C
Call count5
Choreo Run OK!!!
Object = 29.55*C
Call count6
Choreo Run OK!!!
Object = 29.43*C
Call count7
Choreo Run OK!!!
Object = 29.57*C
Call count8
Choreo Run OK!!!
Object = 34.99*C
Call count9
Choreo Run OK!!!

Triggered! Calling SendSMS Choreo...
HTTP_CODE
```

6. Now everything inside Aandroid application will be updated as shown in:



SMS From: +12404892302
Sent from your Twilio trial account - Hi sir, Your body temperature detected HYPERTAMIA (high temperature)!

SMS From: +12404892302
Sent from your Twilio trial account - Hi sir, Your body temperature detected HYPERTAMIA (high temperature)!

SMS From: +60198009330
Test