Instructions To follow

**Step 1- Wire Up the entire system**

Make connections according to the circuit diagram attached below. There are lots of connections to be made so be patient and do all the connections carefully. I will suggest making some home type architecture using cardboard so that you can get the real feel of Home Automation. Be careful about PIR Motion sensor as it is prone to sensitive change. Fix it tightly so that it doesn't give false positive otherwise your door will be opening unexpectedly.

RFID Connection is as follow:

- MOSI ---> PIN 51

- MISO ---> PIN 50

- SCK ---> PIN 52

- SS/SDA ---> PIN 9

- RST ---> PIN 8

Also connect pin 20 of mega to pin 2 of arduino yun and pin 21 of arduino mega to pin 3 of arduino yun for I2C communication.

**Step 2- Create Temboo Account and set up Dropbox API**

The next step in this project is to create and set up an account on the web service Temboo, so you can use the wide range of services provided by Temboo. To do so, the first step is to simply go to the Temboo website at <http://temboo.com/>. On the main page, simply enter your e-mail address to register and click on Sign up. You will then be asked to enter some basic information about your account, such as your account name. Then, you will be prompted to create your first app. Ensure that you save the details of your account, such as the name of your first app and the key that will be given to you.In case you need to get these values again or create a new application, you can always access this data in the My Account section of the Temboo website by clicking on the MANAGE button below APPLICATIONS.

Now, on the software side, you will need a bit more than just the Arduino IDE. We are going to install the required software for the camera directly on the Yún board when we connect to it via SSH, but you will need the Temboo Python SDK to upload pictures on to Dropbox. You can find the SDK at [https://temboo.com/python.](https://temboo.com/python) It's time to unpack it and place it at the root of the SD card as well. Just make sure that it appears with the name temboo in the root of the SD card, so the Python file we just created can access it correctly Then, you also need to have a Dropbox account, so you can upload pictures on to it. You can simply create an account by going to <https://www.dropbox.com/home.> Once your account is created, you need to create an app that will be used by your project. This basically means that you have to authorize the project you are going to build in this chapter to automatically send pictures to your Dropbox account without having to enter your login and password every time. You will also be given all the required information (such as an API key) that we will enter later in the Python script on Yún.

1. To create an app, first go to <https://www.dropbox.com/developers/apps>.

2. Then, click on Create app in the top-right corner of the window. You can now choose the type of app you want to create. In our case, we want to use the Dropbox API directly.

3. You will then be prompted to choose the kind of data your app needs to store. We want to upload pictures, so choose Files and datastores.

4. You can then finish the process of creating your Dropbox app.

5. On the confirmation page that describes the app, you will need to write down the App key and App secret, which we will need for the rest of the project.

6. Also, make sure that the Permission type field is set to App folder. This will ensure that the pictures are uploaded to the folder dedicated to the app and that the Yún won't have access to the rest of your Dropbox folder.

7. What you need to get now is the Token key and Token secret relative to your Dropbox app, so you can enter them later in the software of our project. To get them, the first step is to go to the InitialiseOAuth Choreo on the Temboo website at <https://temboo.com/library/Library/Dropbox/OAuth/InitializeOAuth/>. Here, you will need to enter the App key and App secret. This will generate some additional information such as a callback ID and a temporary token secret. You'll also be asked to visit a link to Dropbox to confirm the authentication.

8. Finally, go to the FinalizeOAuth page to finish the process. You'll be asked to enter your App key, App secret, callback ID, and temporary token secret at <https://temboo.com/library/Library/Dropbox/OAuth/FinalizeOAuth/>. After this step, you'll be given your final Token key and Token secret. Write them down as you'll need them later.

**Step 3:Installing drivers in Arduino Yun**

Install Putty software and open it. Enter the ip address of your arduino yun and select ssh and click open. Enter root as your username and you will then be prompted to enter the password that you chose for your Yún (Default password is arduino). If it works, you should see the following screenshot being displayed on your terminal, which indicates that you are now working directly on the Yún.

You can access all the functions from your Yún Linux machine. We are now going to install the required software for the camera. This requires the Arduino Yún to be connected to the Internet so that it can get the required packages, as described in the following steps:

1. The process starts by updating the package manager, opkg, as follows: opkg update

2. Install the UVC drivers, as follows: opkg install kmod-video-uvc

3. Install the python-openssl package that we will use later in the project, as shown in the following command: opkg install python-openssl

4. Finally, you can install the fswebcam software that we will use to take pictures, as shown in the following command: opkg install fswebcam

**Step 4: Live Video Surveillance**

To make the application work, we first need to install some additional software packages on the Yún, as shown in the following steps:

1. Connect to the Yún again using SSH with your Arduino Yún name and password, and type the following command to get the correct package for live streaming: wget http://www.custommobileapps.com.au/downloads/mjpgstreamer.ipk

2. Note that if the link is not valid anymore and you can't find the files, this package is also available inside the code of this chapter. You can now install it with the following command: opkg install mjpg-streamer.ipk

3. You can now start the live streaming software on your Arduino Yún using the following command:

mjpg\_streamer -i "input\_uvc.so -d /dev/video0 -r 640x480 -f 25" -o "output\_http.so -p 8080 -w /www/webcam" &

Here, the parameter after –h is the resolution and the one after –i is the port on which the stream will be available. We also specified the number of frames per second using the –I command. The other options are less important and you do not have to worry about them. You can then access your stream by going to your Arduino Yún's address in your web browser followed by 8080 to specify the correct port. For example, the link will be like this <http://yourarduinoyunipaddress.local:8080/stream.html.> Replace yourarduinoyunipaddress in the above link with the ip address of your arduino yun.

**Step 5: Setting Up Twilio Account for text messages**

Sign up on <https://www.twilio.com/> . Note down the account sid and auth token. Buy a free number on twilio and also verify the number to whom you gonna send the text message under verified caller ids section. Once your number is verified, you are ready to send message to that number. Now log in to [temboo.com.](https://www.temboo.com/) Go to this link <https://temboo.com/library/Library/Twilio/SMSMessages/SendSMS/> and provide the credentials you got from twilio. Before you click run now, note down the additional information. Click run now and provide the additional information required and save the profile with some name. This profile name will be used in the arduino yun code. Go to line 208 in the arduino yun code and replace the your\_profile\_name with your profile name. Make sure that when you click run now button in temboo, it shouldn't give any error over there otherwise there is something wrong with your credentials which you need to troubleshoot yourself.

**Step 6: Setting up blynk account and configuring blynk app**

Download the blynk app from playstore using this link [https://play.google.com/store/apps/details?id=cc.blynk.](https://play.google.com/store/apps/details?id=cc.blynk)

After you download the Blynk App, you’ll need to create a New Blynk account. After you’ve successfully logged into your account, start by creating a new project. Select the hardware model as Arduino yun. Auth Token is a unique identifier which is needed to connect your hardware to your smartphone. Every new project you create will have its own Auth Token. You’ll get Auth Token automatically on your email after project creation. You can also copy it manually. Click on devices section. Now press the “Create” button. Add the widgets according to following instructions:

Make four tab as home, hvac, monitoring and control

Home

Under home tab select four widgets: LCD, Value Display, Level H and notification.

Configure the input of LCD as V1.

Name the value display as fire sensor and configure the input of fire sensor as V11 and map it from range 0 to 1023 and make reading rate as 1 sec.

Name the level h as fire level and configure the input as V12 and map it from range 0 to 1023 and make reading rate as push.

Configure the notification with priority as normal.

HVAC

Under HVAC tab select LCD, Slider, Value display(2) , history graph.

Configure the input of lcd as V3.

Rename the slider as fan and configure the output as D9 and map it from 0 to 255.

Rename the two value displays with room1 temp and room2 temp. Configure one input as V2 and other input as V6 and map them both from 0 to 50 with reading rate as 1 sec.

Configure the inputs of history graph as V9 and V8. Put the name room1 temp corresponding to V9 and room2 temp corresponding to V8.

Monitoring

Under monitoring tab select labeled value (2), graph(2), history graph.

Rename the 2 labeled value widgets with room1 humidity and room2 humidity and set the inputs as V5 and V7 and map them from 0 to 100 with reading rate of 1 sec.

Rename the 2 graphs as room 1 humidity and room 2 humidity with input pins as V10 and V9 with range 0 to 100 and reading rate as 1 sec.

Configure the inputs of history graph as V10 and V7. Put the name room1 humid corresponding to V10 and room2 humid corresponding to V7.

Control

Under control tab select lcd, value display(2), labeled value, slider, time input , GPS Track.

Configure the input of lcd as V4.

Configure the inputs of value display as V19 and V20 with range from 0 to 1023 and reading rate of 1 sec.

Configure the input of labeled value as V21 and range varying from 0 to 1023 with reading rate of 1 sec.

Rename the slider settings as Main room light and output from 0 to 255 with write intervals 100ms.

Configure the time input V31 and configure allow start/stop input as yes, allow day of week selection as yes, allow timezone selection as yes.

Configure the gps track input as v29 and trigger when as enter. This will allow the system to trigger the v29 pin on (which may be connected to geysar for example) when user has arrived inside the map. You can change the radius of the map as per your convenience.

Click on run button.

Congrats! Now you can control and monitor the home environment from your android smartphone.

**Voice controlled operations of devices**

Download the AMR\_VOICE app from this [link](https://play.google.com/store/apps/details?id=robotspace.simplelabs.amr_voice&hl=en) and configure the bluetooth with HC05. Deploy the code mentioned below named as voice control for arduino uno to arduino uno. Click on connect robot in the app. Make sure you have your smartphone bluetooth paired with hc05. Press the voice icon and speak the strings mentioned in the code to turn on/off your devices.

Do let me know if you faced any problems while making this project. Some of the steps are not shown in details considering the fact that users might be familiar with basics of arduino, c and basic linux programming. Feel free to ask any questions in the comments section if you got stuck anywhere or in any step.