**آزمایشگاه سخت افزار**

**دانشکده مهندسی کامپیوتر**

**دانشگاه صنعتی شریف**

**ترم تابستان** **1400-1401**

**استاد : جناب آقای دکتر فصحتی**

**موضوع پروژه:**

**کلید هوشمند کم مصرف (پروژه 2)**

**شماره گروه : 4**

**اعضای گروه:**

**محمدعلی پشنج - 97110036**

**محمدباقر سلطانی - 9810583**

فهرست مطالب

[مقدمه 1](#_Toc112536944)

[شرح کلی پروژه 2](#_Toc112536945)

[کلید هوشمند 2](#_Toc112536946)

[روش و تئوری انجام آزمایش 5](#_Toc112536947)

[راه اندازی سیستم کلید هوشمند 7](#_Toc112536948)

[راه اندازی سیستم اندازه گیری ولتاز 12](#_Toc112536949)

[راه‌اندازی سرور بک‌اند 15](#_Toc112536950)

[نحوه آزمایش 16](#_Toc112536951)

[نتایج و نمودار ها 18](#_Toc112536952)

[اعداد و نمودارهای مربوط به روشن شدن لامپ 1 21](#_Toc112536953)

[اعداد و نمودارهای مربوط به روشن شدن لامپ 2 24](#_Toc112536954)

[اعداد و نمودارهای مربوط به روشن شدن هر دو لامپ 28](#_Toc112536955)

[اعداد و نمودارهای مربوط به روشن شدن لامپ 1 در نبود کد وایفای 31](#_Toc112536956)

[اعداد و نمودارهای مربوط به روشن شدن لامپ 2 در نبود کد وایفای 35](#_Toc112536957)

[اعداد و نمودارهای مربوط به روشن شدن هردو لامپ در نبود کد وایفای 38](#_Toc112536958)

[بررسی و تحلیل نتایج 41](#_Toc112536959)

[تحلیل اول 41](#_Toc112536960)

[تحلیل دوم:‌ بررسی تاثیر ماژول وای فای در مدار 43](#_Toc112536961)

[جمع بندی 45](#_Toc112536962)

فهرست تصاویر

[**1** شمای کلی مدار 2](#_Toc112536979)

[**2** تصویر مدار 3](#_Toc112536980)

[**3** نمای کلی سایت 3](#_Toc112536981)

[**4** بخش کنترل از راه دور 4](#_Toc112536982)

[**5** بخش زمانبندی لامپ ها 4](#_Toc112536983)

[**6** نمایش زمانبندی های ثبت شده 4](#_Toc112536984)

[**7** بخش صوتی 5](#_Toc112536985)

[**8** مدار اندازه گیری ولتاز 5](#_Toc112536986)

[9 قطعه NodeMCU 7](#_Toc112536987)

[انجام تنطیمات مازول 10 8](#_Toc112536988)

[ست کردن مازول11 9](#_Toc112536989)

[تنطیم پورت ورودی12 10](#_Toc112536990)

[13 اتصال قطعه 10](#_Toc112536991)

[14 تنظیمات کتابخانه ها 11](#_Toc112536992)

[15 اردوینو مگا 2560 12](#_Toc112536993)

[**16** کد اردوینو 13](#_Toc112536994)

[**17** خروجی اردوینو 14](#_Toc112536995)

[18سرور فندق 15](#_Toc112536996)

[19اتصال بکند 16](#_Toc112536997)

[**20 کد روشن و خاموش کردن لامپ ها** 17](#_Toc112536998)

[**21** کد تحلیل داده ها 20](#_Toc112536999)

[**22** توان برحسب زمان لامپ 1 21](#_Toc112537000)

[**23** توزیع داده های مربوط به توان بالا لامپ 1 21](#_Toc112537001)

[**24** توزیع داده های مربوط به توان پایین لامپ 1 22](#_Toc112537002)

[**25** منحنی های بالارونده لامپ 1 22](#_Toc112537003)

[**26** منحنی های پایین رونده لامپ 1 23](#_Toc112537004)

[**27** منحنی بالارونده میانگین لامپ 1 23](#_Toc112537005)

[**28**منحنی پایین رونده میانگین لامپ 1 24](#_Toc112537006)

[**29**توان برحسب زمان لامپ 2 24](#_Toc112537007)

[**30** توزیع داده های مربوط به توان بالای لامپ 2 25](#_Toc112537008)

[**31** توزیع داده های مربوط به توان پایین لامپ 2 25](#_Toc112537009)

[**32**منحنی های بالارونده لامپ 2 26](#_Toc112537010)

[**33** منحنی های پایین رونده لامپ 2 26](#_Toc112537011)

[**34** منحنی بالارونده میانگین لامپ 2 27](#_Toc112537012)

[**35** منحنی پایین رونده میانگین لامپ 2 27](#_Toc112537013)

[**36** توان برحسب زمان هردو لامپ 28](#_Toc112537014)

[**37** توزیع داده های مربوط به توان بالای هردو لامپ 28](#_Toc112537015)

[**38** توزیع داده های مربوط به توان پایین هردو لامپ 29](#_Toc112537016)

[**39** منحنی های بالارونده هردو لامپ 29](#_Toc112537017)

[**40** منحنی های پایین رونده هردو لامپ 30](#_Toc112537018)

[**41** منحنی بالارونده میانگین هردو لامپ 30](#_Toc112537019)

[**42** منحنی پایین رونده میانگین هردو لامپ 31](#_Toc112537020)

[43 توان بر حسب زمان لامپ 1 بدون وایفای 31](#_Toc112537021)

[44 توزیع داده های مربوط به توان بالای لامپ 1 بدون وایفای 32](#_Toc112537022)

[45 توزیع داده های مربوط به توان پایین لامپ 1 بدون وایفای 32](#_Toc112537023)

[46 منحنی های بالارونده لامپ 1 بدون وایفای 33](#_Toc112537024)

[47 منحنی های پایین رونده لامپ 1 بدون وایفای 33](#_Toc112537025)

[48 منحنی بالارونده میانگین لامپ 1 بدون وایفای 34](#_Toc112537026)

[49 منحنی پایین رونده میانگین لامپ 1 بدون وایفای 34](#_Toc112537027)

[50 توان بر حسب زمان لامپ 2 بدون وایفای 35](#_Toc112537028)

[51 توزیع داده های مربوط به توان بالای لامپ 2 بدون وایفای 35](#_Toc112537029)

[52توزیع داده های مربوط به توان پایین لامپ 2 بدون وایفای 36](#_Toc112537030)

[53 منحنی های بالارونده لامپ 2 بدون وایفای 36](#_Toc112537031)

[54 منحنی های پایین رونده لامپ 2 بدون وایفای 37](#_Toc112537032)

[55 منحنی بالارونده میانگین لامپ 2 بدون وایفای 37](#_Toc112537033)

[56 منحنی پایین رونده میانگین لامپ 2 بدون وایفای 38](#_Toc112537034)

[57 توان برحسب زمان هردو لامپ بدون وایفای 38](#_Toc112537035)

[58توزیع داده های مربوط به توان بالا هردو لامپ بدون وایفای 39](#_Toc112537036)

[59توزیع داده های مربوط به توان های پایین هردو لامپ بدون وایفای 39](#_Toc112537037)

[60 منحنی های بالارونده هردو لامپ بدون وایفای 40](#_Toc112537038)

[61 منحنی های پایین رونده هردو لامپ بدون وایفای 40](#_Toc112537039)

[62 منحنی بالارونده میانگین هردو لامپ بدون وایفای 41](#_Toc112537040)

[63 منحنی پایین رونده میانگین هردو لامپ بدون وایفای 41](#_Toc112537041)

[باتری 500 میلی امپرساعت64 43](#_Toc112537042)

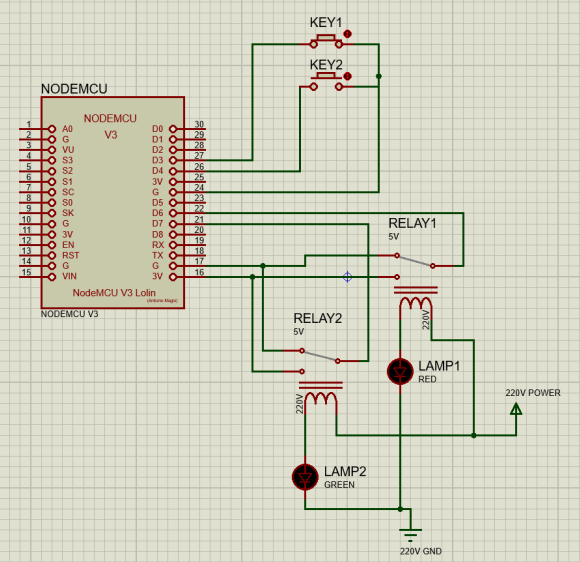
# مقدمه

امروزه با پیشرفت تکنولوژی بسیاری از قطعاتی که در گذشته به شکل مکانیکی کنترل می شدند به صورت الکترونیکی و اینترنتی کنترل می شوند. این پدیده در مورد کلید های الکتریکی نیز صادق می باشد. کلید های هوشمند توانایی کنترل از راه دور را دارا هستند و برای آنها می توان برنامه زمانی تعیین کرد که روشن و یا خاموش شوند. البته میزان مصرف انرژی این کلید های هوشمند همواره مورد بررسی بوده است. در بازار می توان کلید های هوشمند مختلفی پیدا کرد اما میزان مصرف آنها مورد سوال می باشد. در این پروژه ما با استفاده از یک کلید هوشمند دو پل قصد داریم مصرف آن را مورد بررسی قرار دهیم و ببینیم که آیا بر حسب میزان مصرف آن باتری ای در بازار وجود دارد که تا 2 سال طول عمر داشته باشد یا خیر. در ادامه به شرح مراحل انجام آزمایش و نتایج آن می پردازیم.

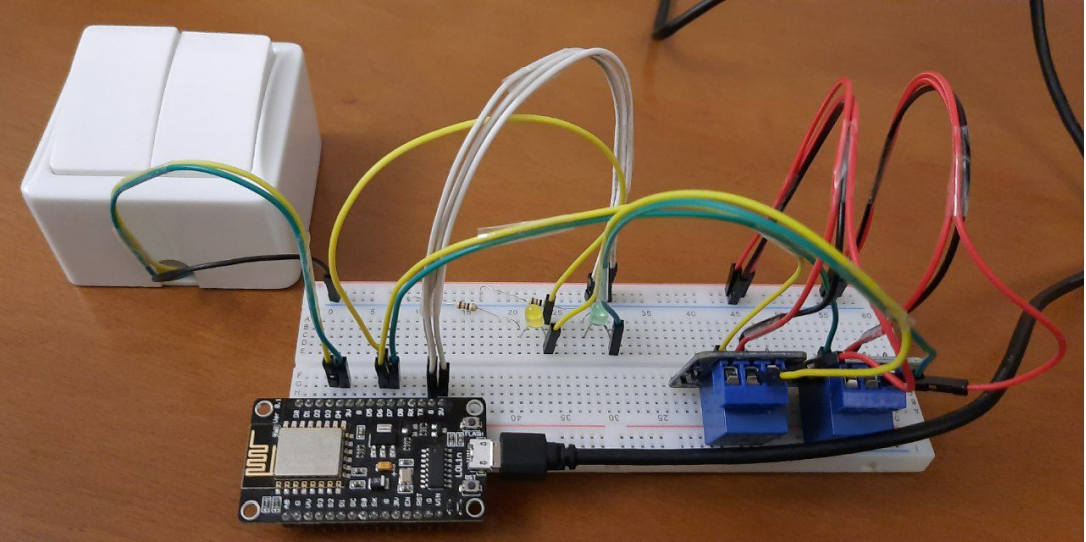
# شرح کلی پروژه

## کلید هوشمند

در این پروژه از کلید هوشمند ساخته شده توسط دانشجویان ترم های گذشته استفاده کردیم. این دستگاه شامل یک کلید دوپل، دو ال ای دی به عنوان لامپ، یک ماژول وایفای NodeMCU و دو رله می باشد. در شکل زیر می توانید شمای کلی و فیزیکی مدار را مشاهده نمایید.

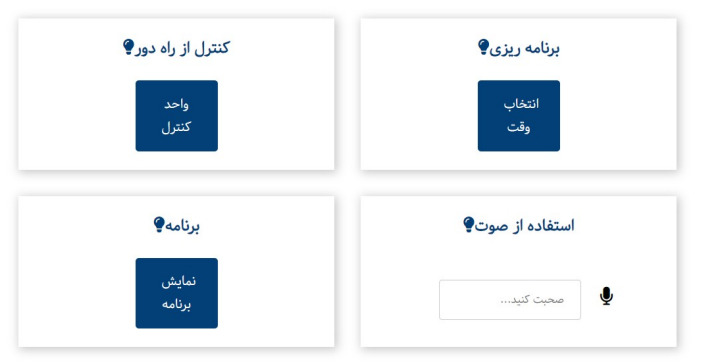


**1** شمای کلی مدار



**2** تصویر مدار

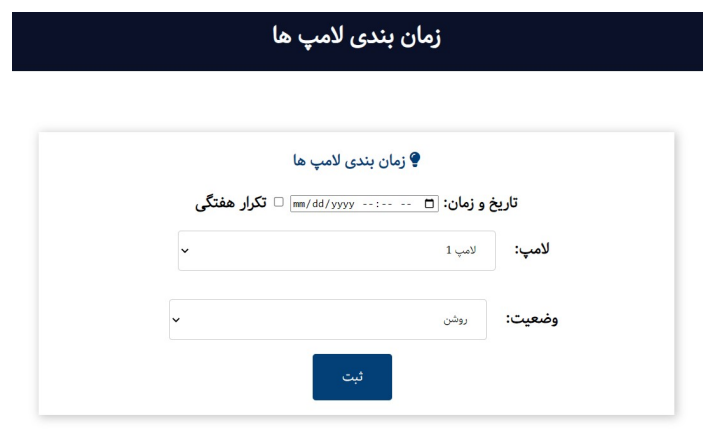
همچنین این مدارس به دلیل داشتن ماژول وایفای توانایی اتصال به اینترنت را دارد. به همین خاطر می توان آن را علاوه بر زدن کلید به صورت فیزیکی، از راه دور نیز کنترل کرد و همچنین این کار را با صوت انجام داد. این کلید دارای محیط نرم افزاری در ادرس http://myproject-magiclamp.fandogh.cloud/control/ می باشد که محیط ان به شکل زیر است.



**3** نمای کلی سایت



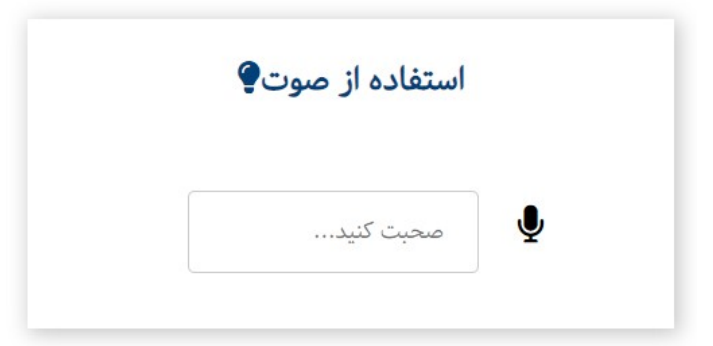
**4** بخش کنترل از راه دور



**5** بخش زمانبندی لامپ ها



**6** نمایش زمانبندی های ثبت شده



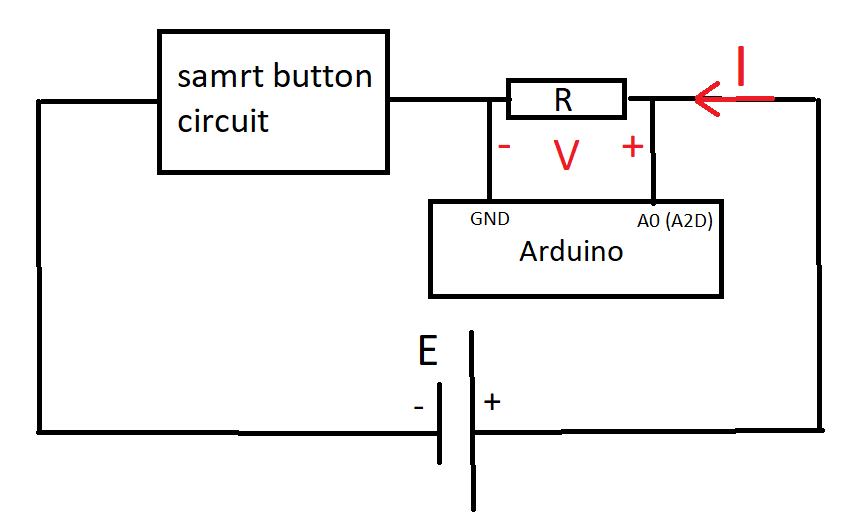
**7** بخش صوتی

جهت کسب اطلاعات بیشتر و تکمیلی در مورد کلید هوشمند می توانید گزارش نهایی پروژه دانشجویان انجام دهنده آن را در لینک مشاهده فرمایید . https://github.com/Sharif-University-ESRLab/project-team-9/blob

## روش و تئوری انجام آزمایش

در این آزمایش به دلیل اینکه هدف بررسی میزان مصرف کلید هوشمند می باشد نیاز بود راهکاری بیندیشیم تا بتوانیم توان مصرفی این مدار را محاسبه کنیم. به دلیل تغییرات مکرر ولتاز و در نتیجه جریان این قطعه در هر بار خاموش و روشن شدن امکان استفاده از ولتمتر و آمپرمتر برای به دست آوردن ولتاز و جریان مدار وجود نداشت. به همین خاطر چاره ای دیگر اندیشیدیم تا با استفاده از آن بتوانیم متغیر های مورد نظر را به شکلی دقیق و سریع به دست بیاوریم تا بتوان از توان مصرفی خازن را به طور لحظه ای محاسبه کنیم.

روش ما با استفاده از یک مقاومت و یک چیپ آردوینو انجام می شود. لازم به ذکر است که در این آزمایش از Arduino Mega 2560 استفاده شد. مدار اندازه گیری ولتاز دو سر کلید هوشمند به شکل زیر پیاده سازی شد.



**8** مدار اندازه گیری ولتاز

در مدار شکل بالا مشخص است که مقاومت را با مدار سری میکنیم و با وصل کردن زمین و پورت آنالوگ آردوینو به دو سر آن می توان ولتاز و در نتیجه جریان آن را به دست آورد. به این ترتیب می توان توان مصرفی مدار کلید هوشمند را محاسبه کرد.

به این ترتیب توان مصرفی کلید هوشمند محاسبه می شود. گفتنی است در این آزمایش و با استفاده از اندازه گیری با مولتی متر می باشند. توجه کنید که عدد E را کوچک انتخاب کردیم که حتی وقتی مقاومت سهم کمی از افت ولتاژ را دارد و در نتیجه ولتاژ مدار از 3.8 بیشتر نشود و در نتیجه نسوزد. ضمناً این ولتاژ به اندازه‌ای نیز زیاد است که حتی در بیشترین افت ولتاژ در مقاومت نیز مدار روشن می ماند.

# راه اندازی سیستم کلید هوشمند

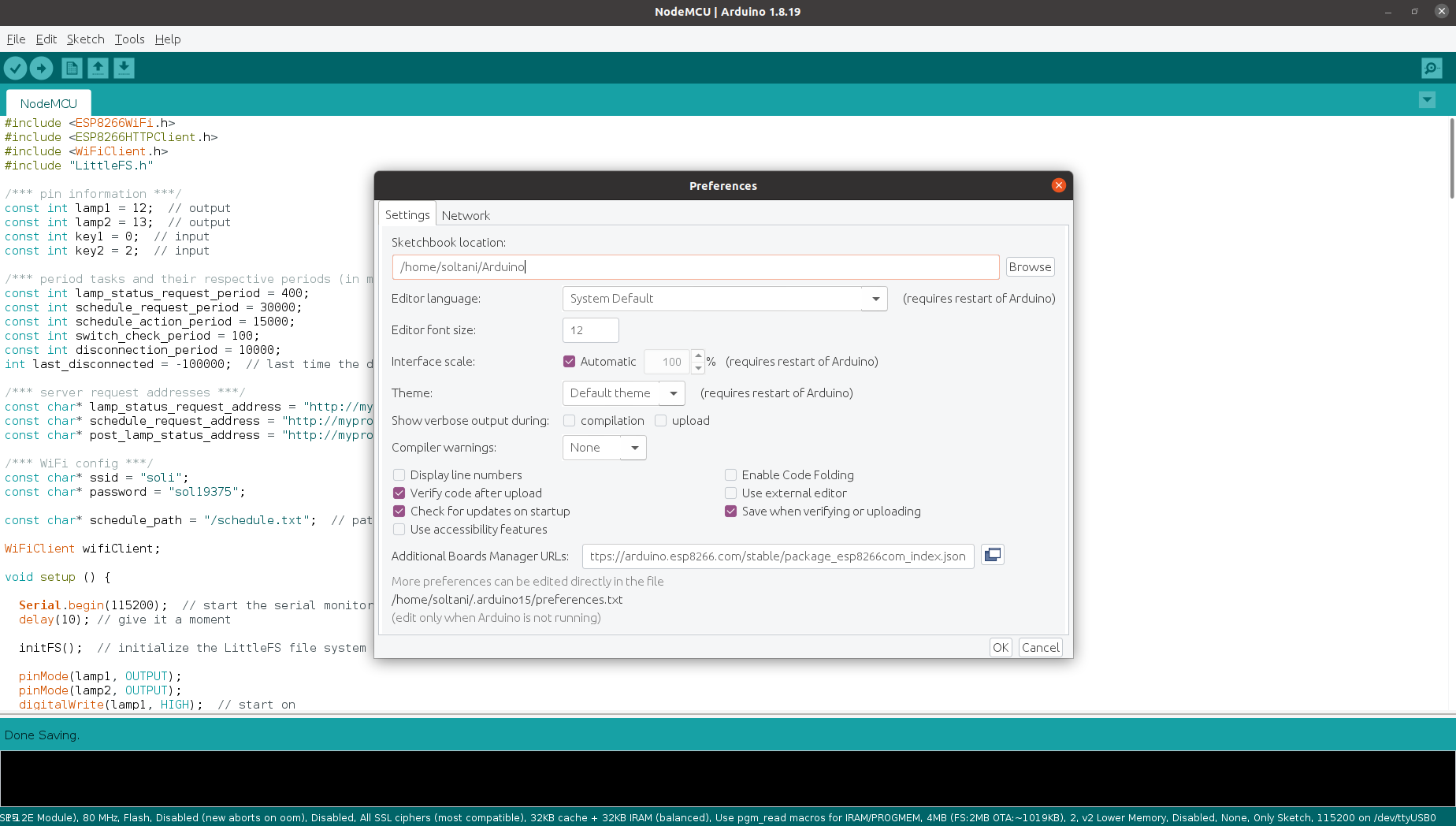
برای راه اندازی سیستم کلید هوشمند از میکروکنترلر NodeMCU V3 با تراشه CH340 استفاده شده است. این ماژول را می‌توان در تصویر زیر مشاهده کرد.



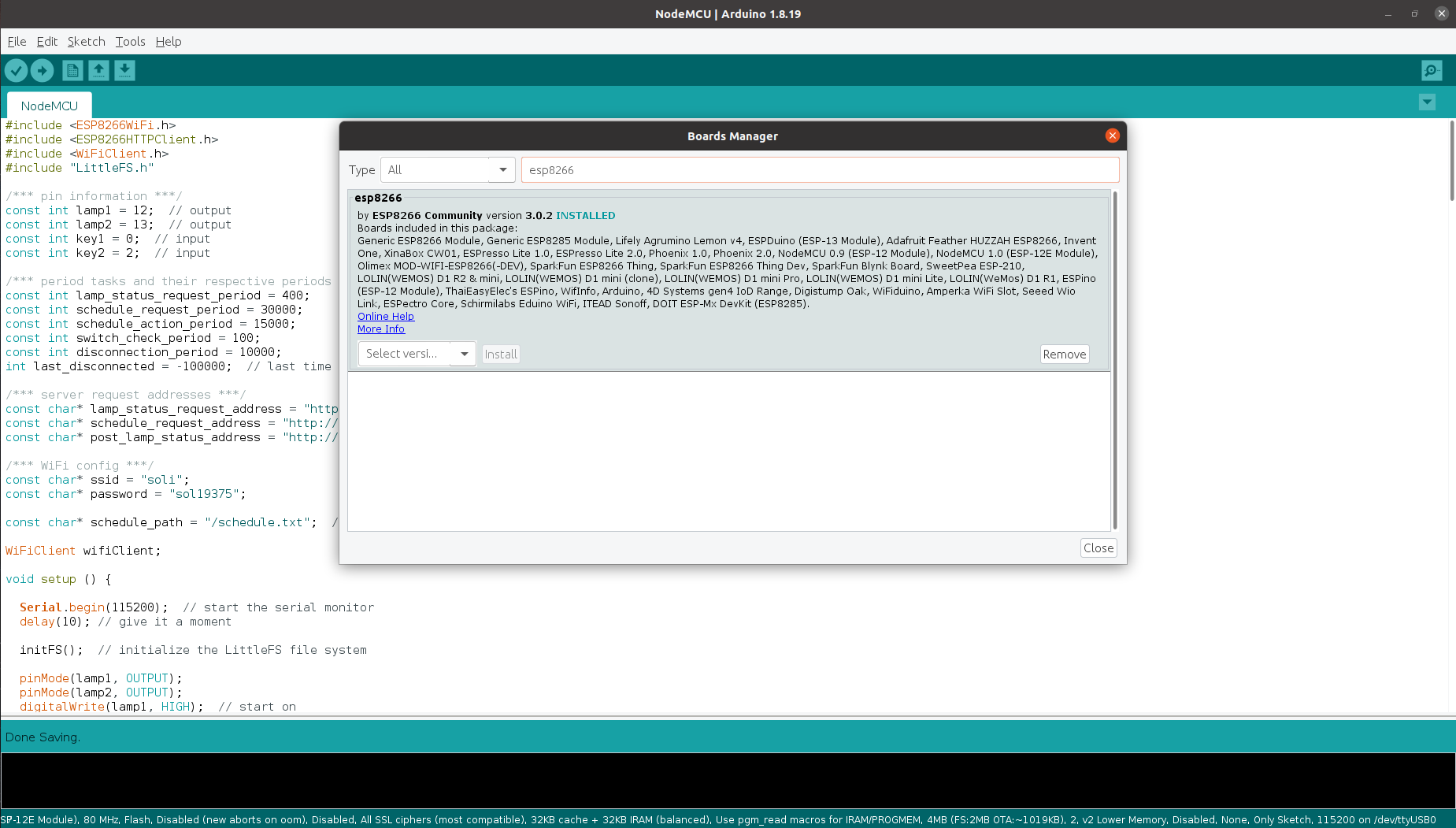
9 قطعه NodeMCU

برای این که مدار به درستی کار کند باید میکروکنترلر را برنامه نویسی کرد. به این منظور باید مراحل زیر را طی کرد.

1. اتصال ماژول به کامپیوتر با استفاده از کابل USB به microUSB: به این منظور باید از کابل مخصوصی که به همراه ماژول است استفاده کرد.
2. نصب درایور پورت COM/Serial:‌ برای این که بتوان به ESP8266 (ماژول وایفای روی میکروکنترلر) کد آپلود کرد و از طریق Console Serial خروجی سریال مدار را مشاهده کرد نیاز است که درایور چیپ سریال، در اینجا CH340G، را از [این لینک](https://github.com/nodemcu/nodemcu-devkit/tree/master/Drivers) دانلود و طبق توضیحات داده شده نصب کرد (توجه کنید در نسخه های جدیدتر کرنل لینوکس این درایور به طور پیشفرض نصب است).
3. نصب Arduino IDE با نسخه ۱.۶.۴ یا بالاتر: برای نوشتن کد روی میکروکنترلر می‌توان از زبان Lua یا نرم افزار Arduino IDE استفاده کرد که کار کردن با دومی راحت تر است و این امکان را می‌دهد که با استفاده از کد آردوینو (زبانی نزدیک به C) ماژول را برنامه نویسی کرد. در اینجا از Arduino IDE 1.18.9 استفاده شده است که از [این لینک](https://www.arduino.cc/en/software) می‌توان آن را دانلود نمود.
4. نصب ESP8266 Board Package: به این منظور نرم افزار Arduino IDE را باید اجرا کرده و در منوی Prefrences لینک <https://arduino.esp8266.com/stable/package_esp8266com_index.json> را در

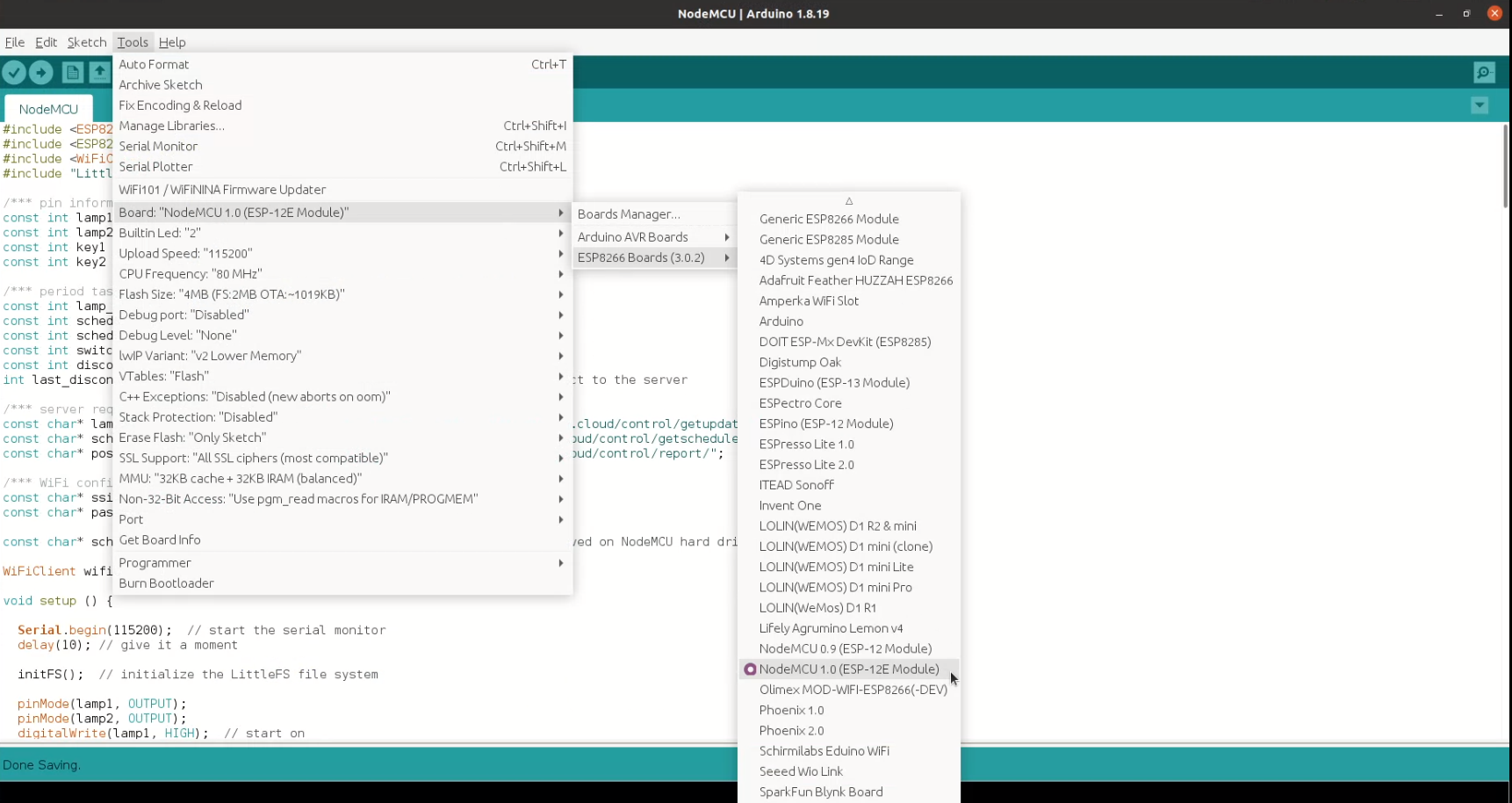
بخش Aditional Board Manager URLs اضافه کرد. سپس باید در منوی Boards Manager پکیج esp8266 by esp8266 را نصب کرد و سپس برنامه Arduino IDE را ریستارت کرد.  
  


انجام تنطیمات مازول 10

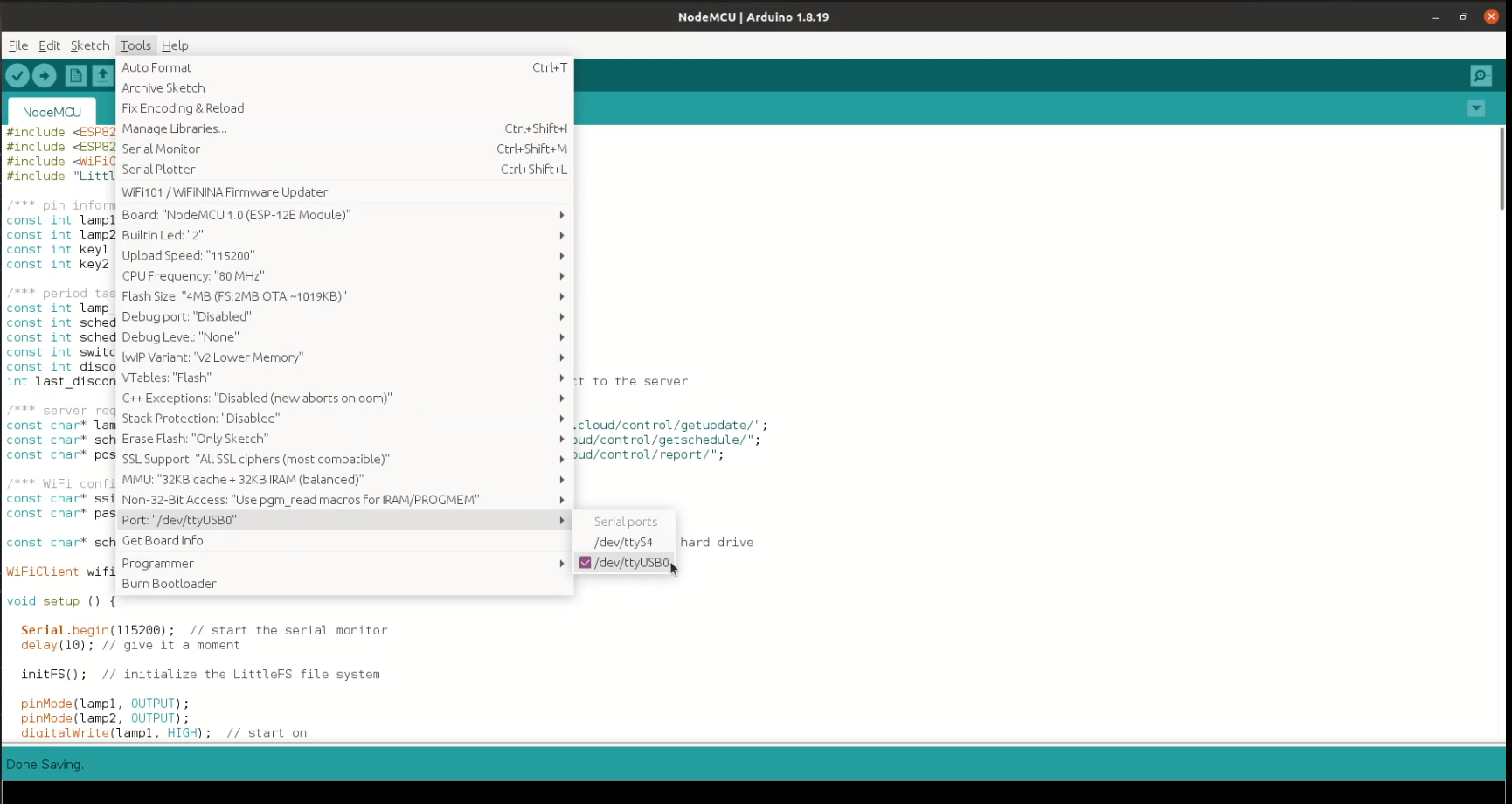


Boards Manager11

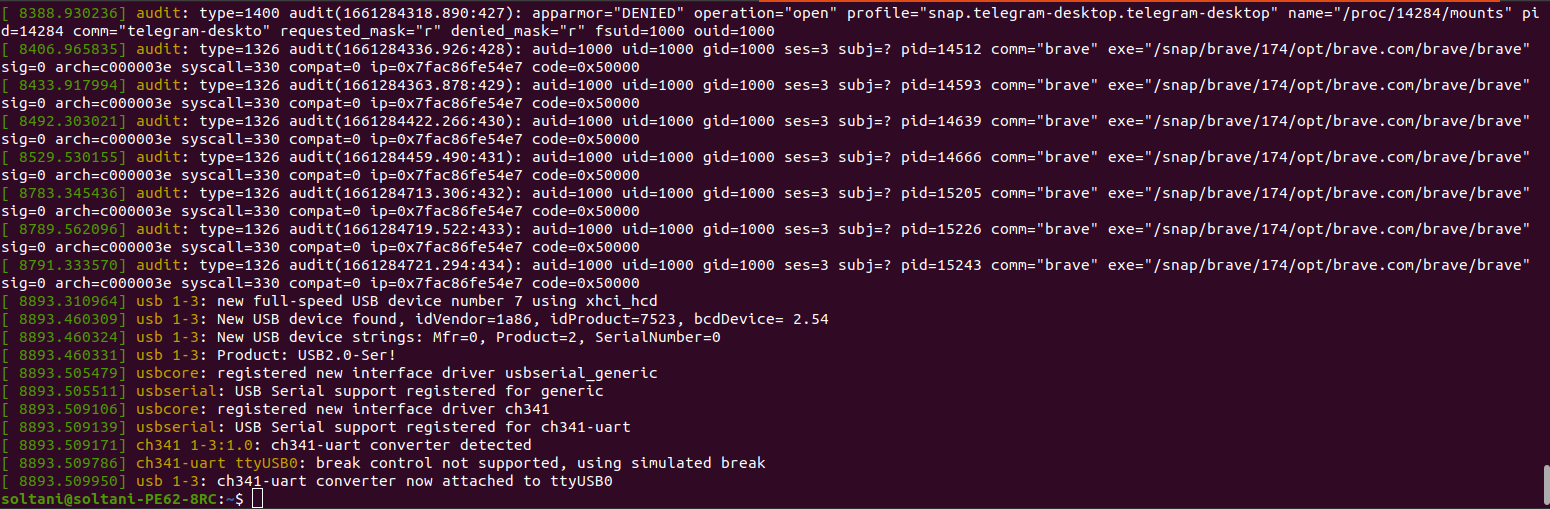
1. تنظیم کردن Board Manager: ابتدا باید از قسمت Tools>Boards برد NodeMCU 1.0 را انتخاب کرد (به این دلیل که مدل دقیق ماژول وایفای ESP8266-12E است). سپس از قسمت Tools>Port پورت USB را که ماژول به آن وصل است را انتخاب کرد. (در لینوکس با استفاده از دستور dmesg میتوان یک لاگ از وضعیت پورت ها مشاهده کرد).



ست کردن مازول11



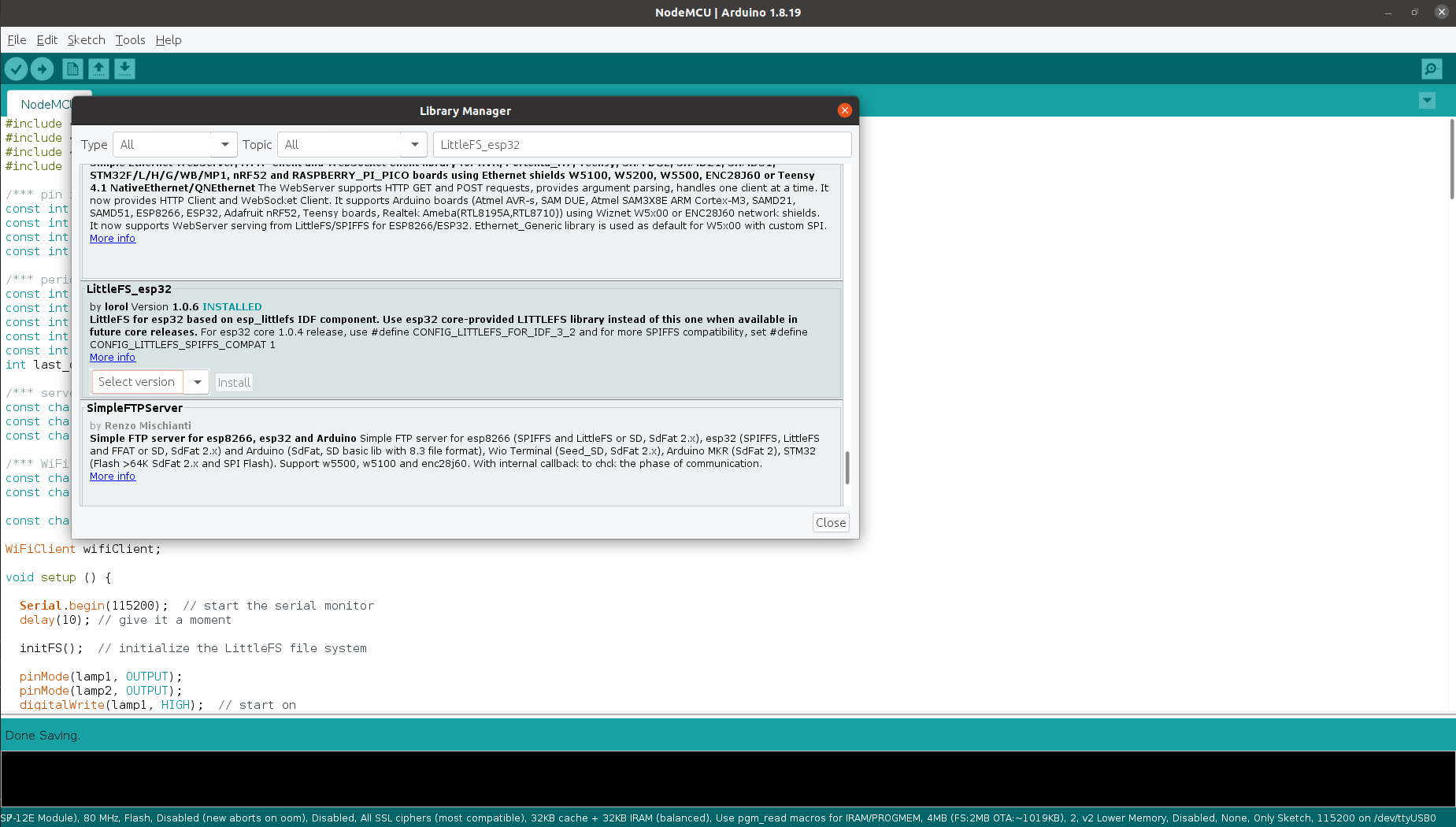
تنطیم پورت ورودی12



13 اتصال قطعه

بعد از راه اندازی اولیه می توان کد مورد نظر را روی ماژول ‌آپلود کرد. در اینجا ما از کد گروه قبلی (با تغییر اندک)‌ برای برنامه نویسی ماژول استفاده کردیم که در [این ریپو](https://github.com/Sharif-University-ESRLab/project-team-9/blob) قابل مشاهده است و کد تغییر داده شده ما در [اینجا](https://github.com/Sharif-University-ESRLab/project-97110036-98105813/blob/main/Code/microcontroller/NodeMCU/NodeMCU.ino) قرار دارد. در ضمن این کد از ۴ کتابخانه ESP8266WiFi، WiFiClient، ESP8266HTTPClient و LittleFS استفاده می‌کند که سه مورد اول به صورت پیشفرض در محیط Arduino IDE در دسترس اند ولی کتابخانه LittleFS، که یک کتابخانه ساده برای خواندن و نوشتن داده روی فایل سیستم میکروکنترلر است، باید به کتابخانه های Arduino اضافه شود که در منوی Tools>Manage Libraries با جستجوی LittleFS\_esp32 یافت می‌شود. ضمنا این نکته نیز حائز اهمیت است که

این کتابخانه در پوشه Libraries محل کار Arduino ذخیره می‌شود و جهت این که به مشکلی در include کردن این کتابخانه ها نخورد لازم است که محل ذخیره کد خارج این پوشه باشد.



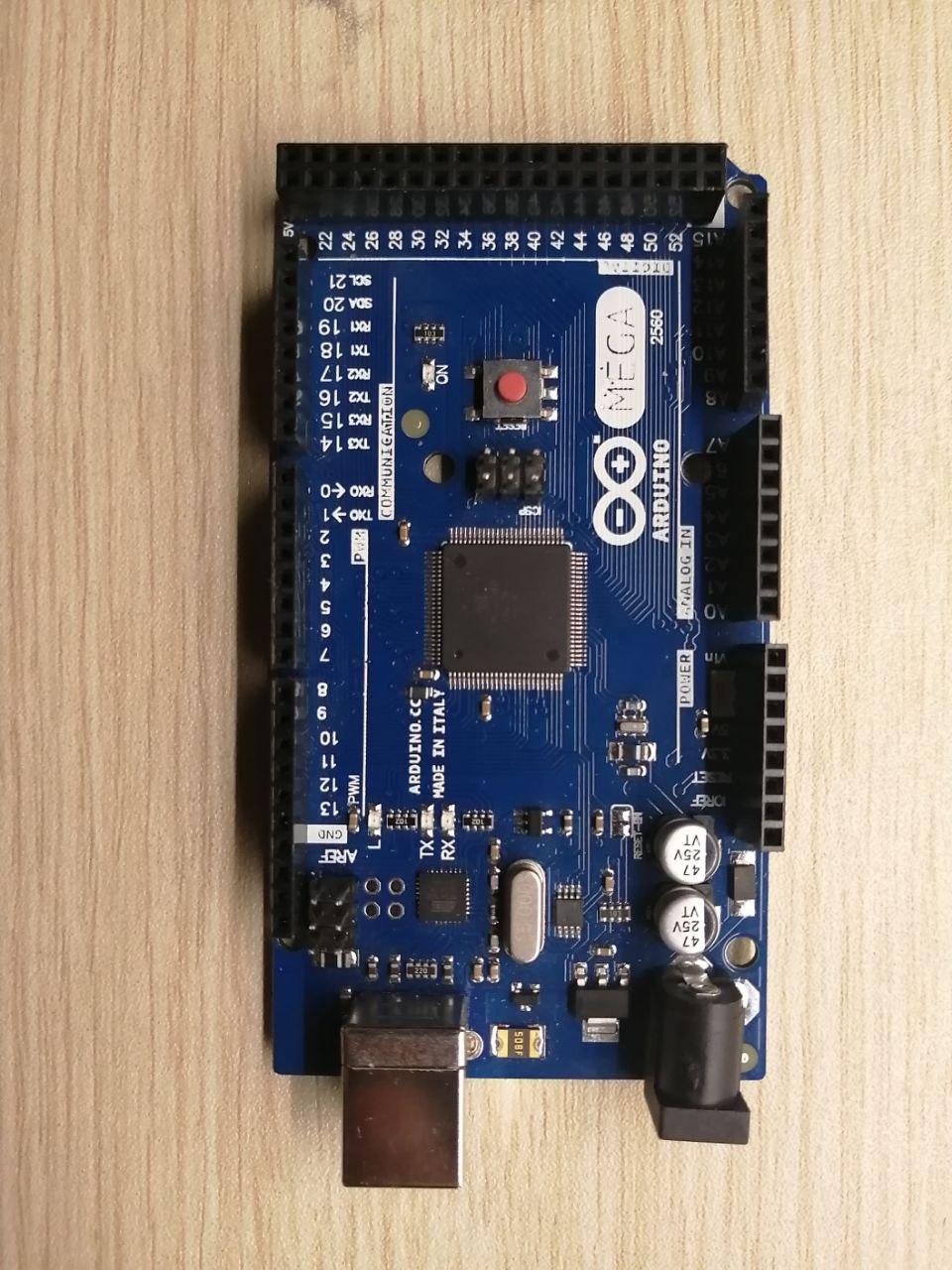
14 تنظیمات کتابخانه ها

در انتهای این مرحله کار ما برای برنامه نویسی میکروکنترلر تمام است.

## 

# راه اندازی سیستم اندازه گیری ولتاز

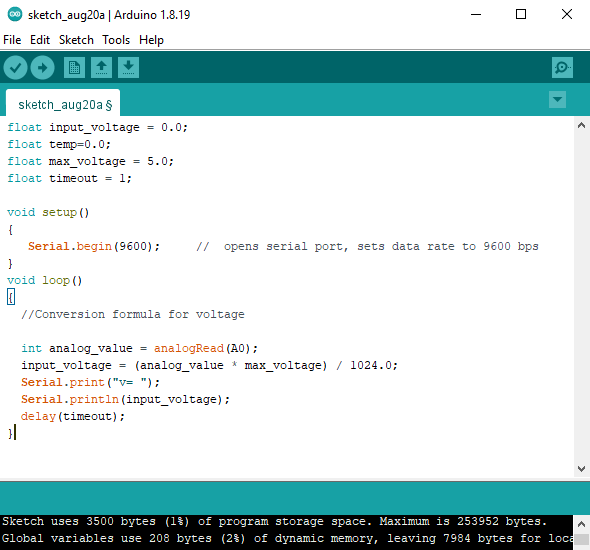
جهت راه اندازی سیستم اندازه گیری ولتاژ از Arduino Mega 2560 استفاده شد که تصویر آن را در زیر مشاهده می کنید.



15 اردوینو مگا 2560

کار همانطور که گفته شد به این صورت است که مقاومت را با مدار سری میکنیم و با وصل کردن زمین و پورت آنالوگ آردوینو به دو سر آن می توان ولتاز و در نتیجه جریان آن را به دست آورد.

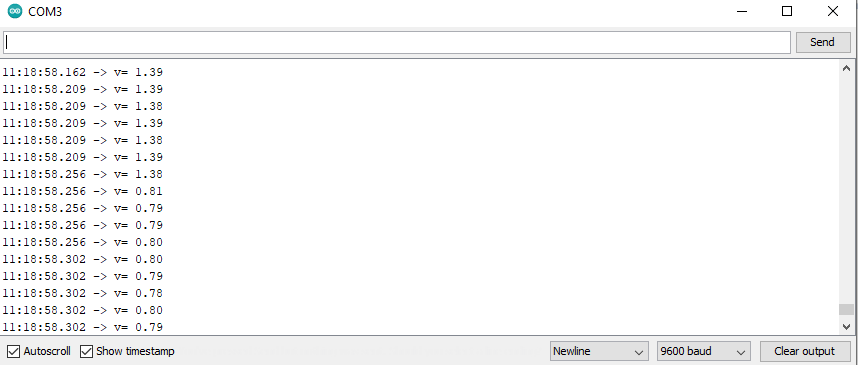
برای به دست آوردن ولتاز از طریق پورت آنالوگ لازم بود تا آردوینو را پروگرم کنیم تا بتواند مقصودمان را پیاده سازی کند. برای این منظور قطعه کد زیر را روی آن پیاده کردیم.



**16** کد اردوینو

در این کد ابتدا پورت سریال 9600 را باز می کنیم و سپس در یک حلقه بی نهایت هر بار ورودی آنالوگ را از پورت می خوانیم و با استفاده از تبدیل خطی آن را به ولتاژ 0 تا 5 ولت مپ می کنیم و آن را چاپ می‌کنیم. این عمل با تاخیر 1 میلی ثانیه ای تکرار می شود تا بتوانیم با بالاترین دقت ولتاز ها را ثبت کنیم.

خروجی های آن به شکل زیر چاپ می‌شوند.

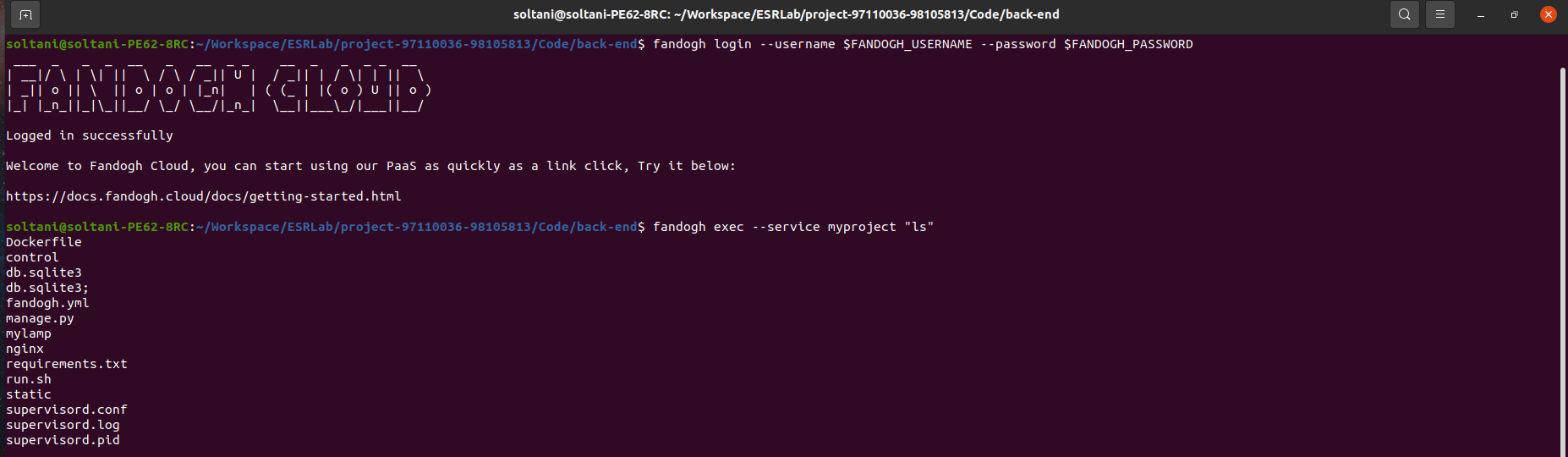


**17** خروجی اردوینو

# راه‌اندازی سرور بک‌اند

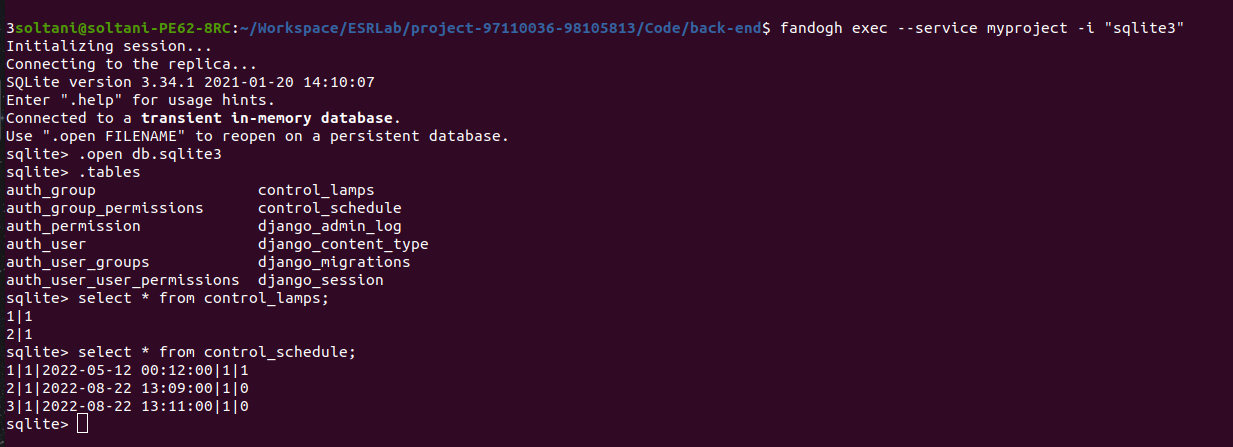
برای راه اندازی سرور بک‌اند نیاز است که یک سرور مجازی از سایت های سرویس دهی فضای ابری خریداری کرد. ما ازسایت fandogh.cloud یک سرویس VPS با ۱ گیگابایت رم، ۰.۵ هسته cpu و ۱۰ گیگابایت فضای ذخیره سازی خریداری کردیم و سرور back-end را روی آن پیاده‌سازی کردیم. برای کار کردن با این سرور مجازی به صورت remote نیاز است که برنامه fandogh cli را طبق [این مراحل](https://docs.fandogh.cloud/docs/preface/getting-started/) نصب کرد و به حساب کاربری fandogh ساخته شده در سایت از طریق fandogh cli متصل شد. نهایتا نیاز است تا کد بک‌اند مورد نظر را روی سرور deploy کرد. به این منظور می‌توان از داکر استفاده کرد یا در مورد تمپلیت های خاص سرور از جمله Django, Laravel, Flask و غیره با توجه به توضیحات داده شده در [این لینک](https://docs.fandogh.cloud/docs/source-deployments/) سرور را deploy کرد. ما از کد بک‌اند گروه قبلی (بدون تغییر) که در [این ریپو](https://github.com/Sharif-University-ESRLab/project-team-9/blob) و همچنین [این لینک](https://github.com/Sharif-University-ESRLab/project-97110036-98105813/tree/main/Code/back-end) قرار داده شده است استفاده کردیم و با توجه به توضیحات داده شده در [این لینک](https://docs.fandogh.cloud/docs/source-deployments/source-django/) آن را روی سرور مجازی deploy کردیم.

تنها نکته ای که در مورد راه اندازی این سرور وجود دارد این است که هر بار که برای اولین بار یک پروژه روی سرویس مجازی deploy می‌شود، دیتابیس آن ریست می‌شود و در نتیجه نیاز است که داده‌های لازم را در جداول control\_lamps و control\_schedules وارد کرد. سرویس fandogh امکان برقراری ارتباط ssh به صورت مستقیم به سرور را نمی‌دهد اما می توان یک سری دستورها را با دستور fandogh exec روی سرور اجرا و در نتیجه تغییرات لازم را روی دیتابیس اعمال کرد.



18سرور فندق

توجه کنید که به صورت پیشفرض برنامه کار با دیتابیس sqlite3 ممکن است نصب نباشد و در نتیجه باید روی سرور با استفاده از دستور apt install sqlite3 آن را نصب کرد.



19اتصال بکند

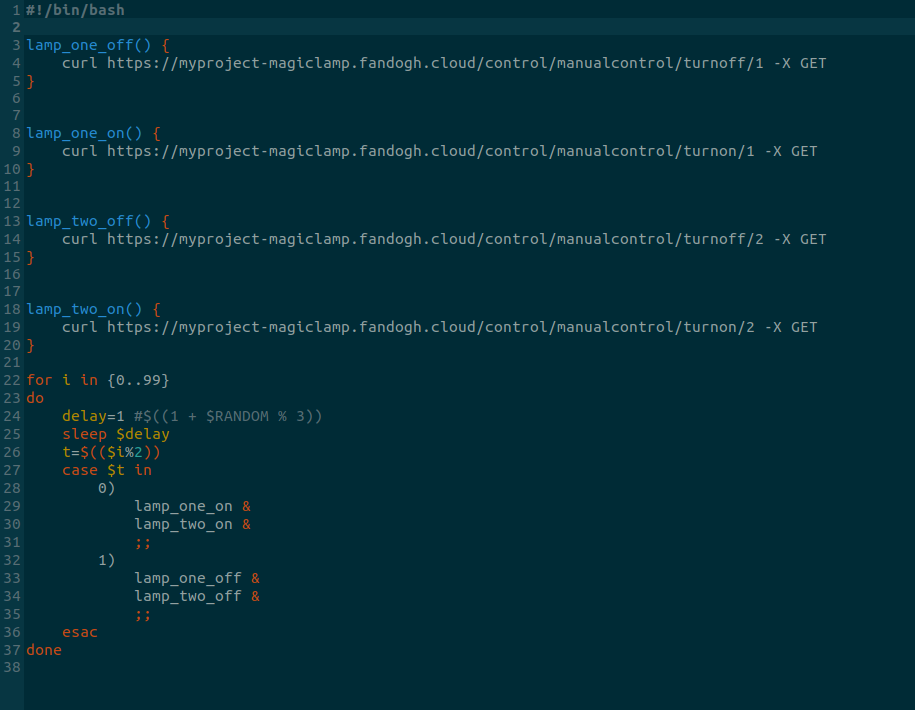
می‌توان عملیات دیتابیسی متعدد را با استفاده از interactive shell روی دیتابیس اعمال کرد. در اینجا هر دو لامپ و زمانبندی ها از قبل به دیتابیس اضافه شده بودند.

# نحوه آزمایش

در این ازمایش ما سه سناریو را بررسی کردیم:

1. لامپ 1 روشن و خاموش شود
2. لامپ 2 روشن و خاموش شود
3. هر دو لامپ با هم روشن و خاموش شوند.
4. کد وایفای در برنامه ای که در NodeMCU پروگرم میشود وجود نداشته باشد و لامپ 1 روشن و خاموش شود
5. کد وایفای در برنامه ای که در NodeMCU پروگرم میشود وجود نداشته باشد و لامپ 1 روشن و خاموش شود
6. کد وایفای در برنامه ای که در NodeMCU پروگرم میشود وجود نداشته باشد و هردو لامپ با هم روشن و خاموش شوند

در هریک از موارد بالا 10 بار داده گیری را انجام دادیم تا متغیر های محاسبه شده نهایی مان از دقت بالایی برخوردار باشند. همچنین برای یکسان بودن شرایط آزمایش و دخیل نبودن عوامل انسانی در آن اسکریپت نوشتیم تا ریکوئست های قطع و وصل شدن چراغ ها را به طور منظم ارسال کند و به مقدار یکسانی بیت هر دو درخواست تاخیر اعمال کند. در زیر و همچنین [این لینک](https://github.com/Sharif-University-ESRLab/project-97110036-98105813/blob/main/Code/util/command.sh) می‌توانید مشاهده نمایید.

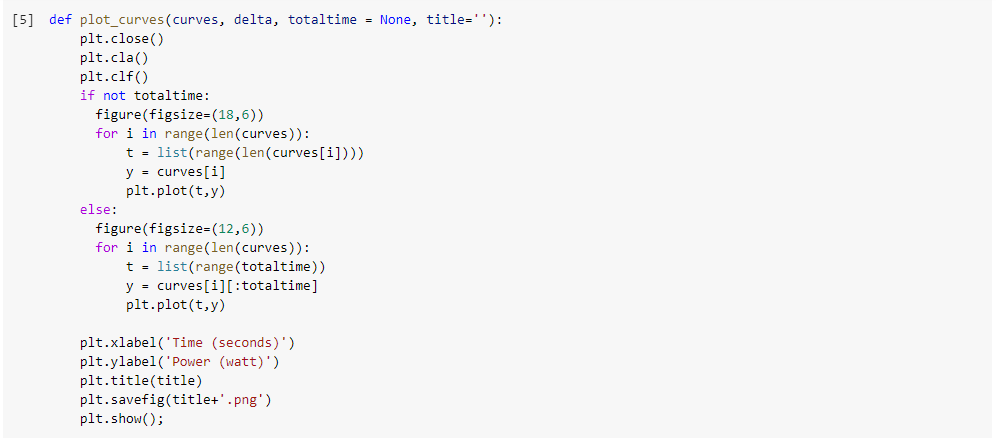
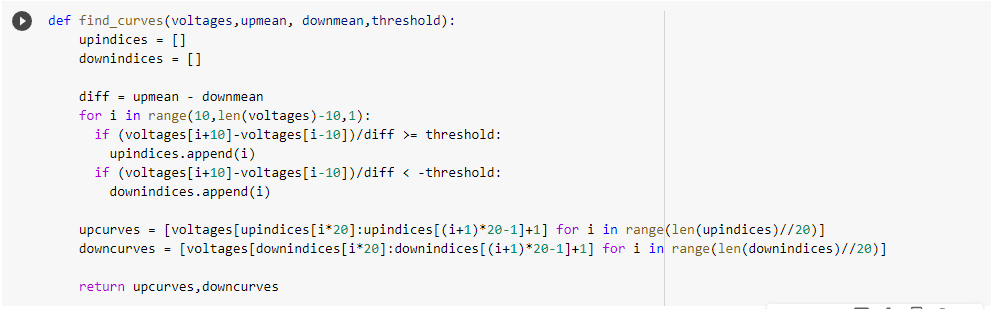
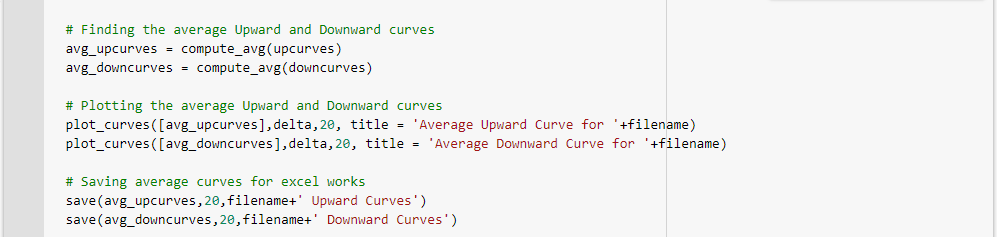
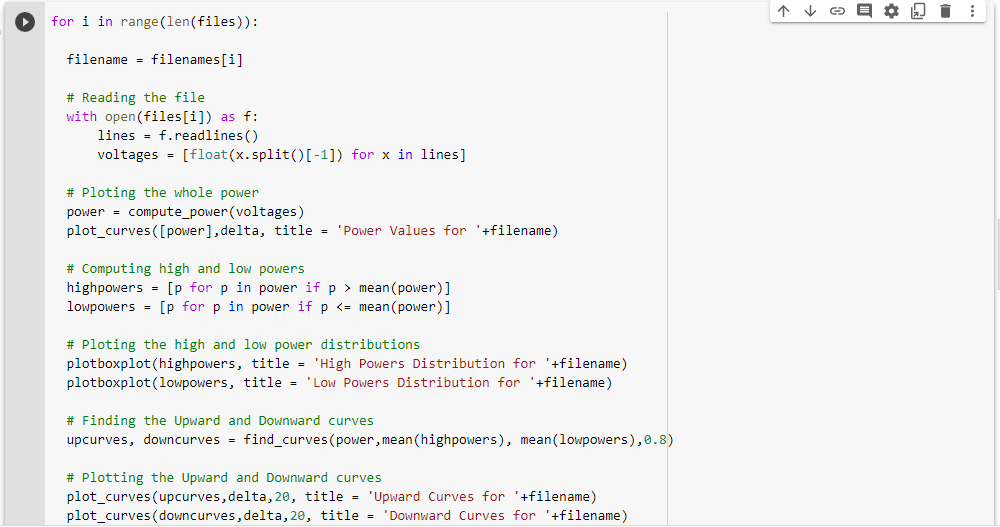
****

**20 کد روشن و خاموش کردن لامپ ها**

با توجه به موارد گفته شده در انتها منحنی توان بر حسب زمان را بررسی می کنیم و توان متوسط در حالت خاموش و روشن را محاسبه می کنیم تا در تخمین های ما استفاده شوند.

# نتایج و نمودار ها

برای تحلیل داده های ولتاژی قطعه کدی به شرح زیر برای پروژه نوشته شد.

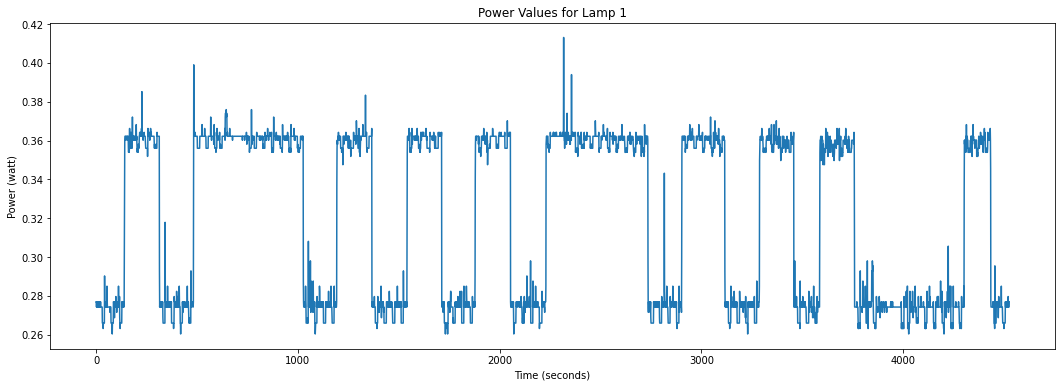
   

**21** کد تحلیل داده ها

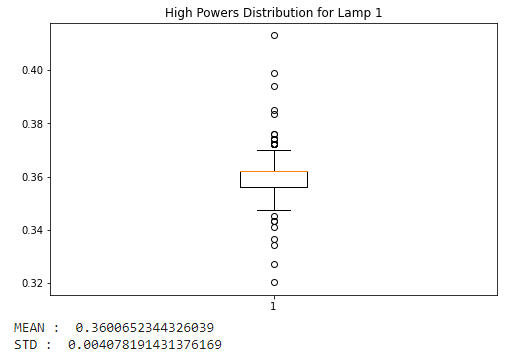
در این کد پس از خواندن داده های ولتاز ابتدا توان محاسبه شده و سپس بر اساس ان قسمت های بالارونده و پایین رونده جدا می شوند. همچنین میانگین مقدار توان در بیشترین و کمترین حالت خود که همان روشن و خاموش بودن لامپ ها می باشد نیز محاسبه میگردد.

نتایج حاصل شده از بررسی داده ها توسط کد به شرح ذیل می باشد.

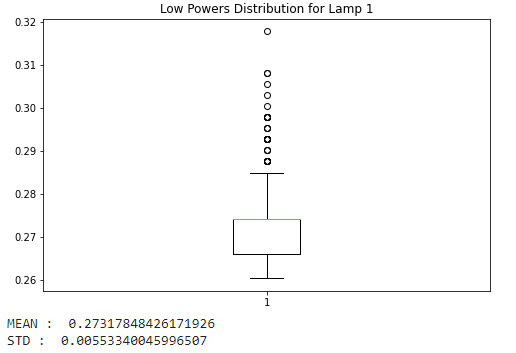
## اعداد و نمودارهای مربوط به روشن شدن لامپ 1



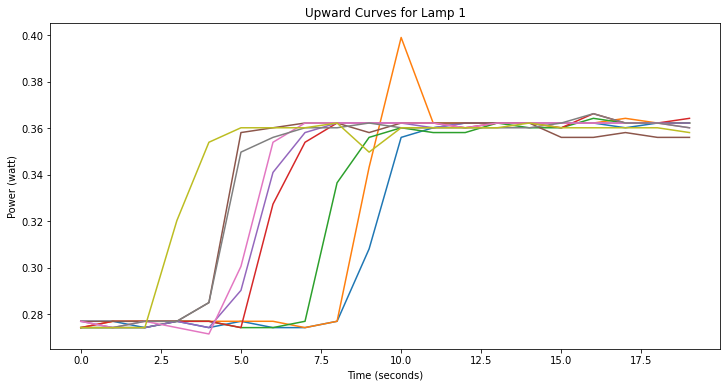
**22** توان برحسب زمان لامپ 1



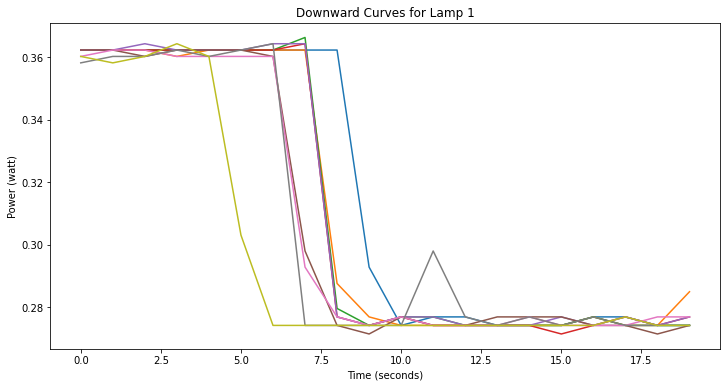
**23** توزیع داده های مربوط به توان بالا لامپ 1



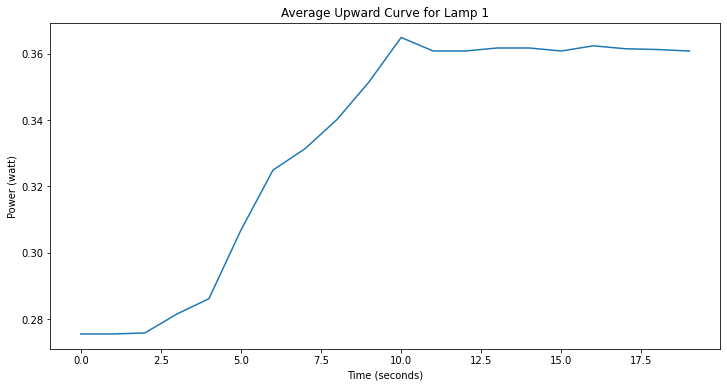
**24** توزیع داده های مربوط به توان پایین لامپ 1



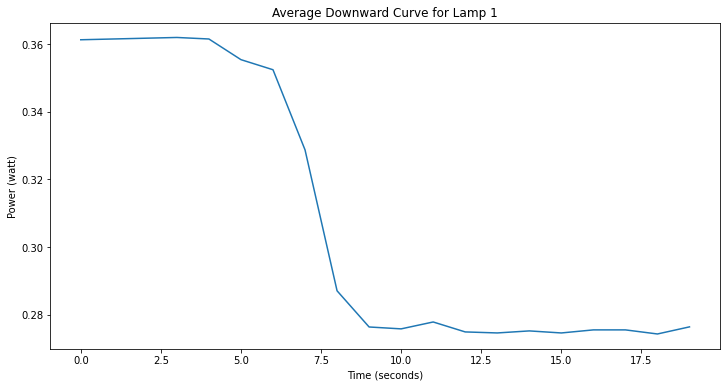
**25** منحنی های بالارونده لامپ 1



**26** منحنی های پایین رونده لامپ 1

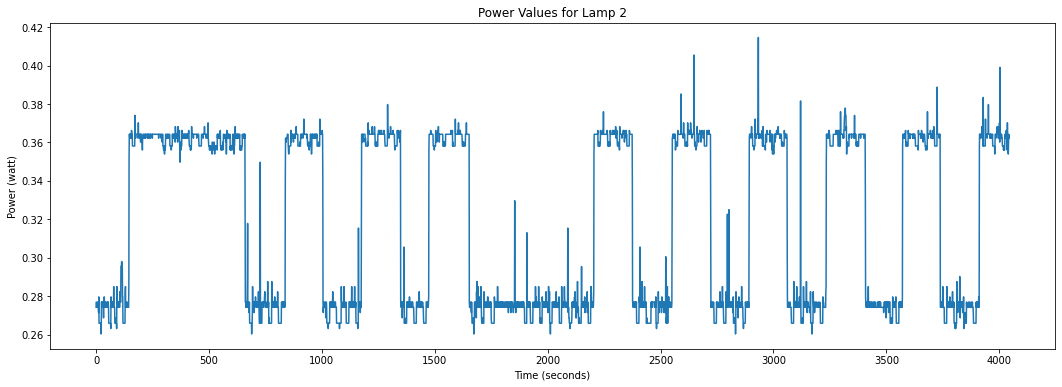


**27** منحنی بالارونده میانگین لامپ 1

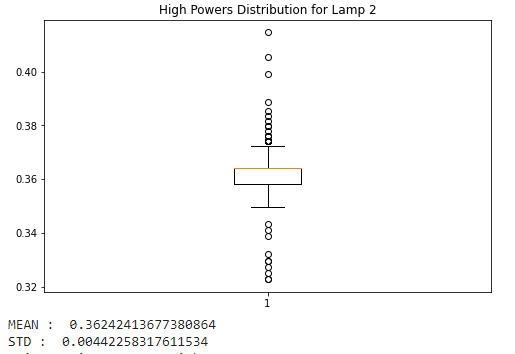


**28**منحنی پایین رونده میانگین لامپ 1

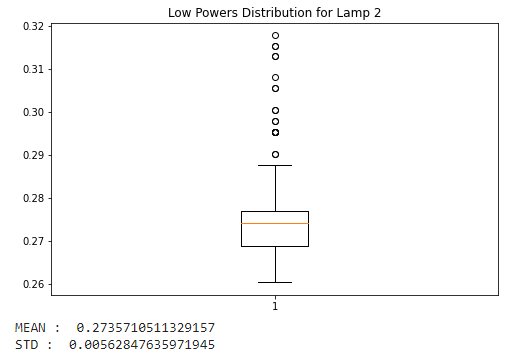
## اعداد و نمودارهای مربوط به روشن شدن لامپ 2



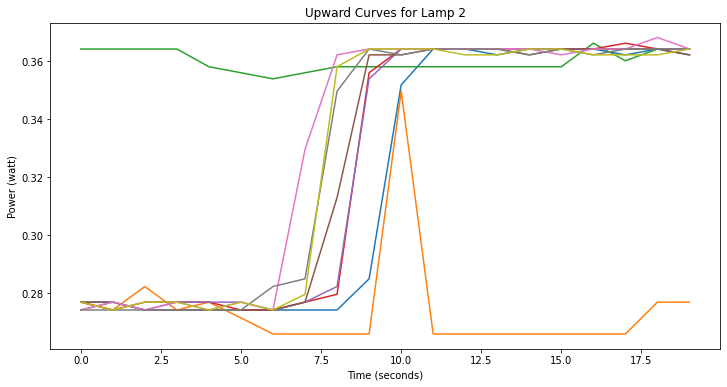
**29**توان برحسب زمان لامپ 2



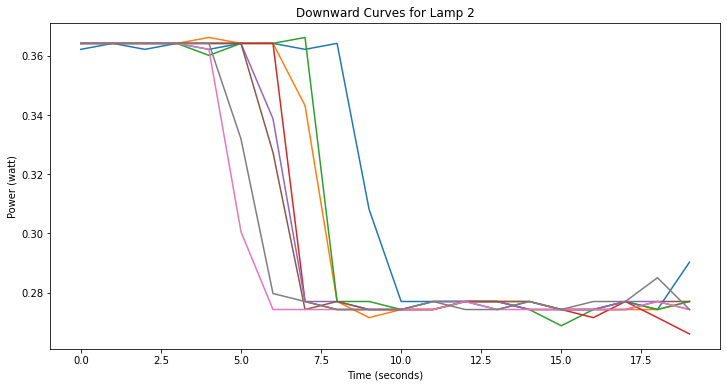
**30** توزیع داده های مربوط به توان بالای لامپ 2



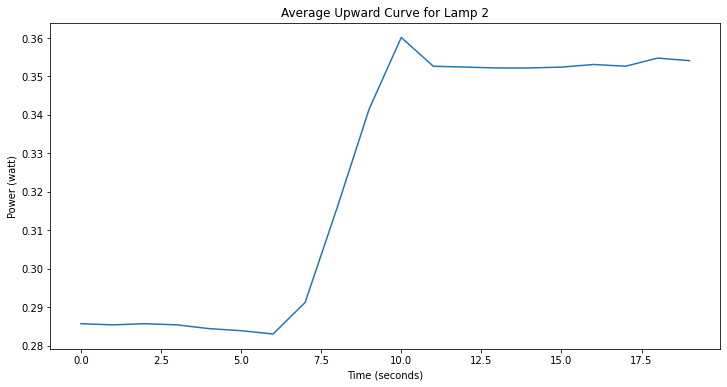
**31** توزیع داده های مربوط به توان پایین لامپ 2



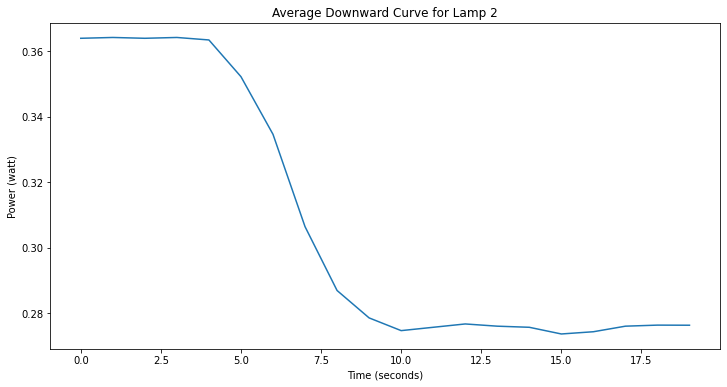
**32**منحنی های بالارونده لامپ 2



**33** منحنی های پایین رونده لامپ 2

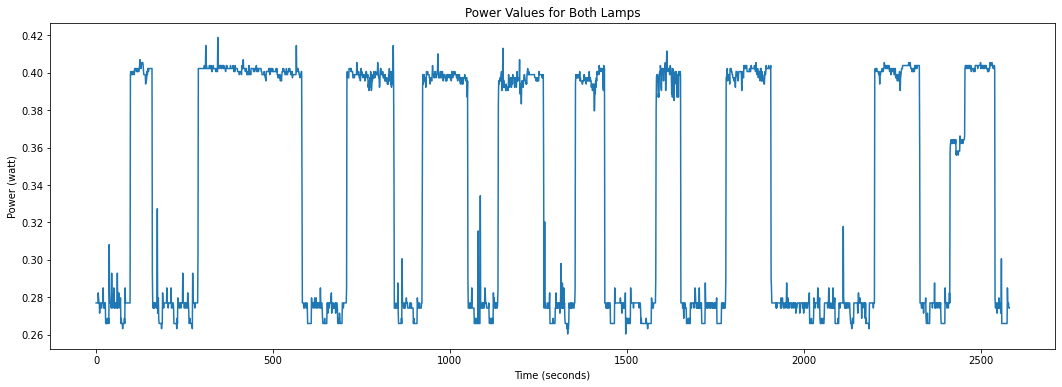


**34** منحنی بالارونده میانگین لامپ 2

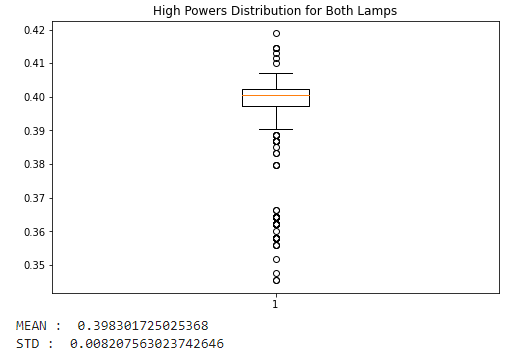


**35** منحنی پایین رونده میانگین لامپ 2

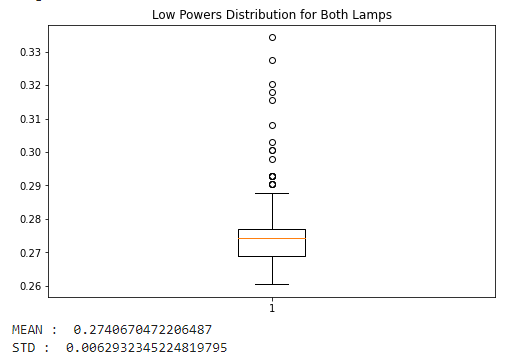
## اعداد و نمودارهای مربوط به روشن شدن هر دو لامپ



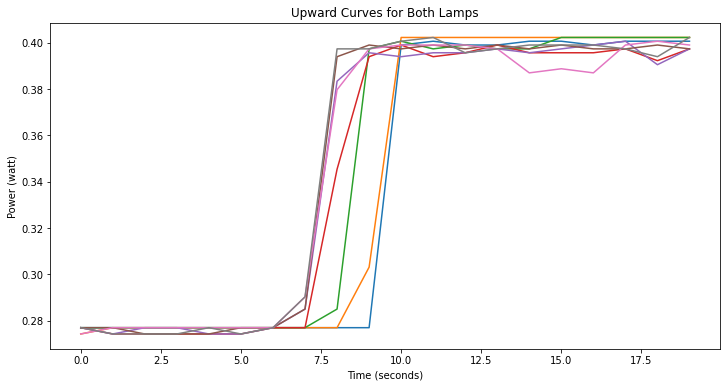
**36** توان برحسب زمان هردو لامپ



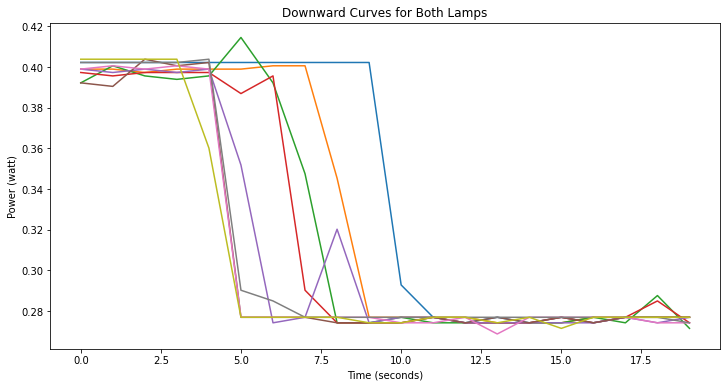
**37** توزیع داده های مربوط به توان بالای هردو لامپ



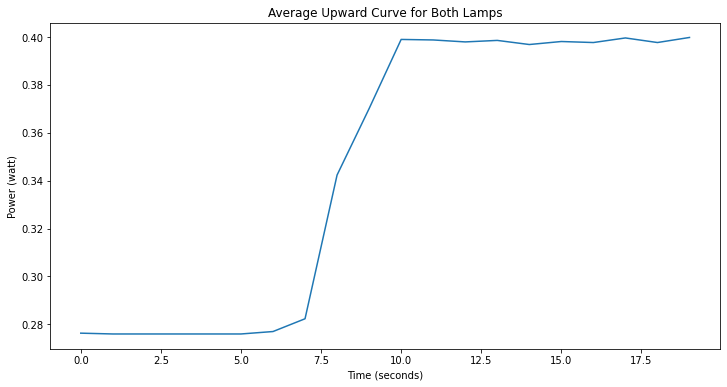
**38** توزیع داده های مربوط به توان پایین هردو لامپ



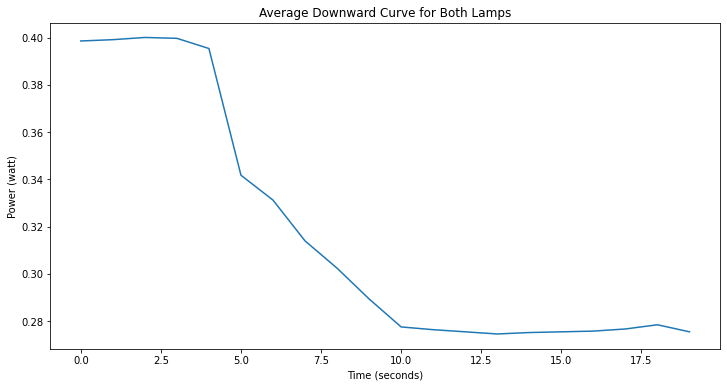
**39** منحنی های بالارونده هردو لامپ



**40** منحنی های پایین رونده هردو لامپ

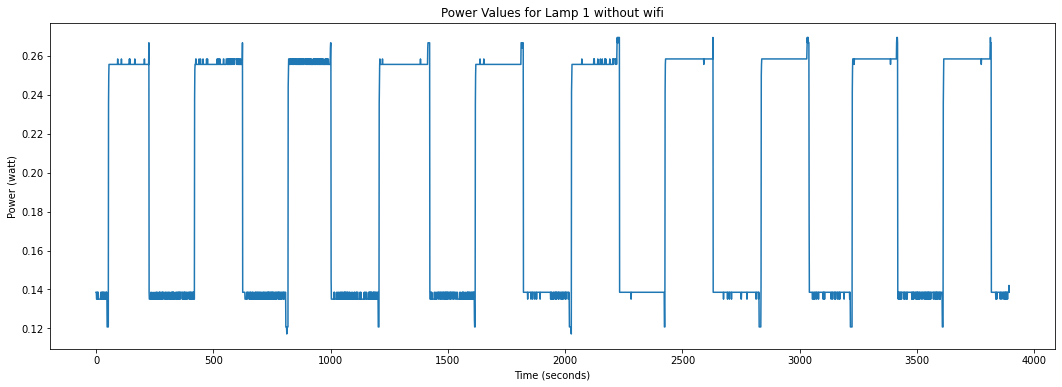


**41** منحنی بالارونده میانگین هردو لامپ

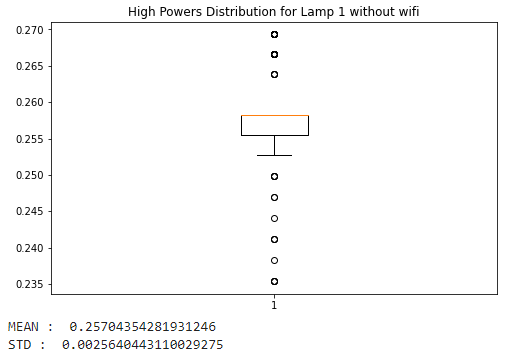


**42** منحنی پایین رونده میانگین هردو لامپ

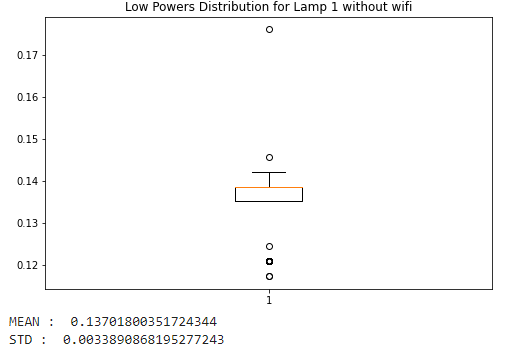
## اعداد و نمودارهای مربوط به روشن شدن لامپ 1 در نبود کد وایفای



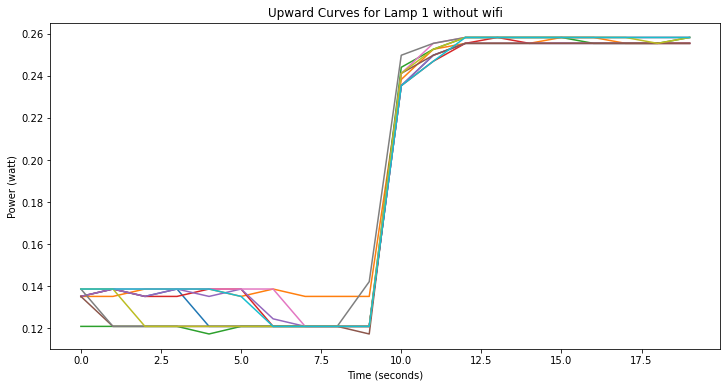
43 توان بر حسب زمان لامپ 1 بدون وایفای



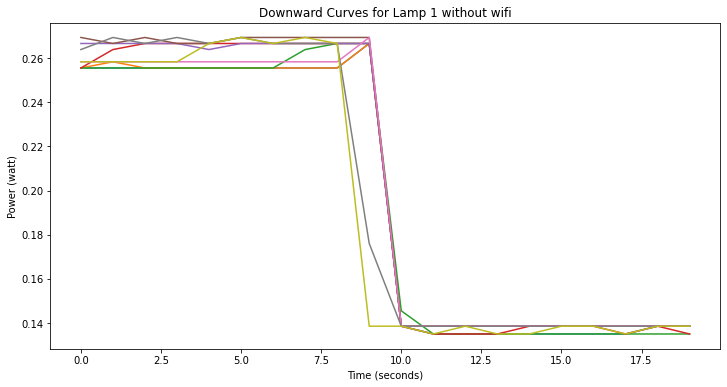
44 توزیع داده های مربوط به توان بالای لامپ 1 بدون وایفای



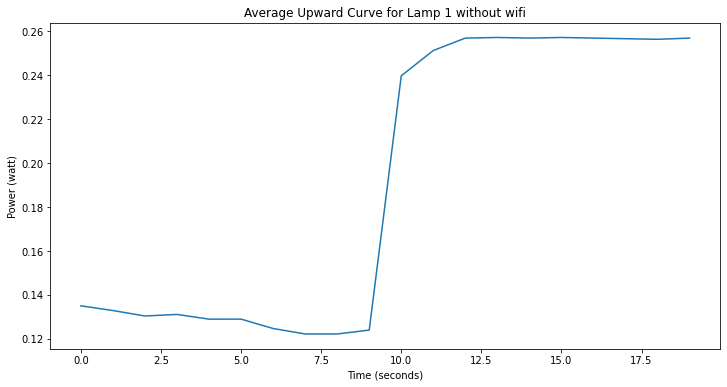
45 توزیع داده های مربوط به توان پایین لامپ 1 بدون وایفای



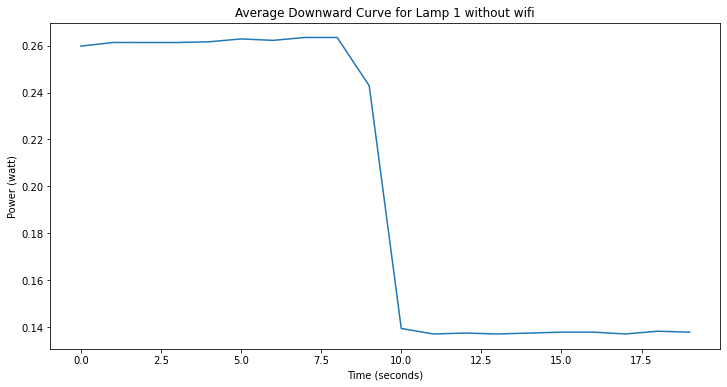
46 منحنی های بالارونده لامپ 1 بدون وایفای



47 منحنی های پایین رونده لامپ 1 بدون وایفای



48 منحنی بالارونده میانگین لامپ 1 بدون وایفای

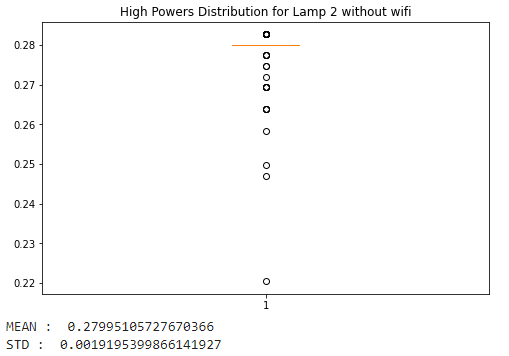


49 منحنی پایین رونده میانگین لامپ 1 بدون وایفای

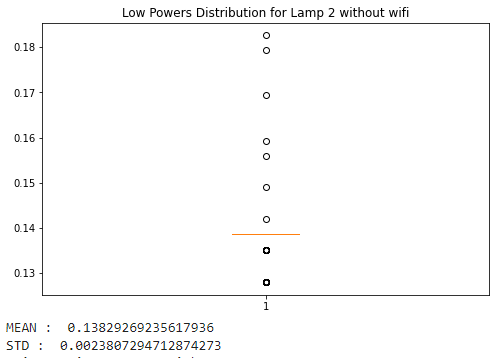
## اعداد و نمودارهای مربوط به روشن شدن لامپ 2 در نبود کد وایفای



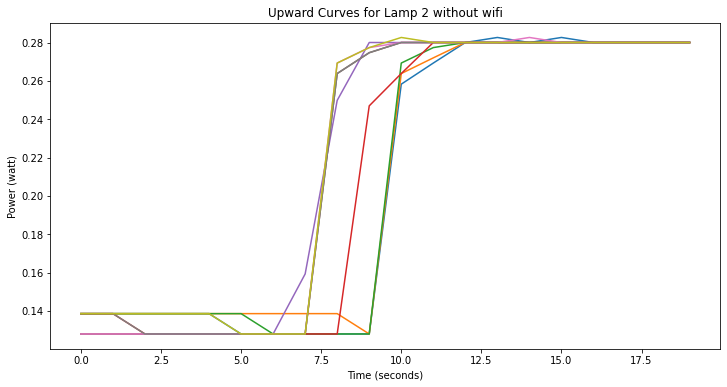
50 توان بر حسب زمان لامپ 2 بدون وایفای



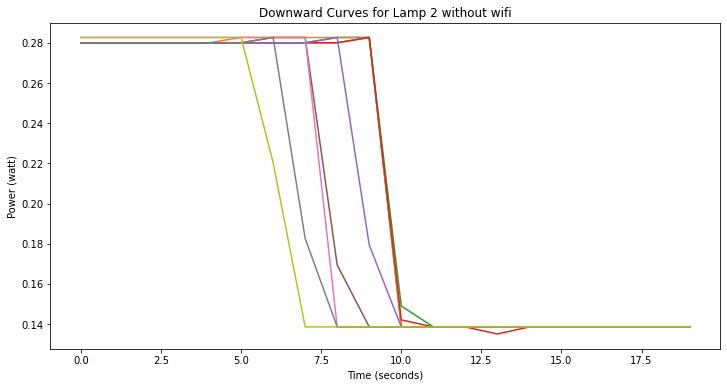
51 توزیع داده های مربوط به توان بالای لامپ 2 بدون وایفای



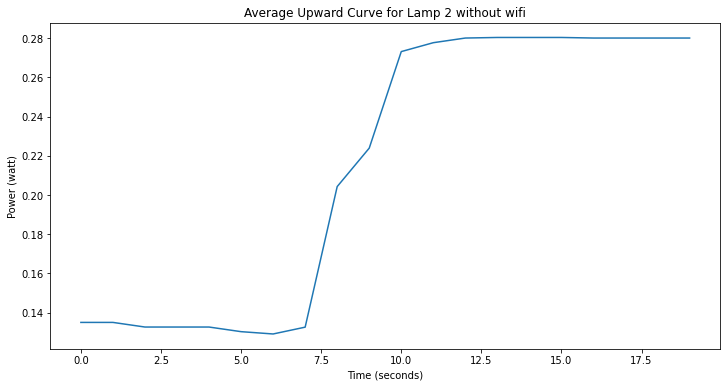
52توزیع داده های مربوط به توان پایین لامپ 2 بدون وایفای



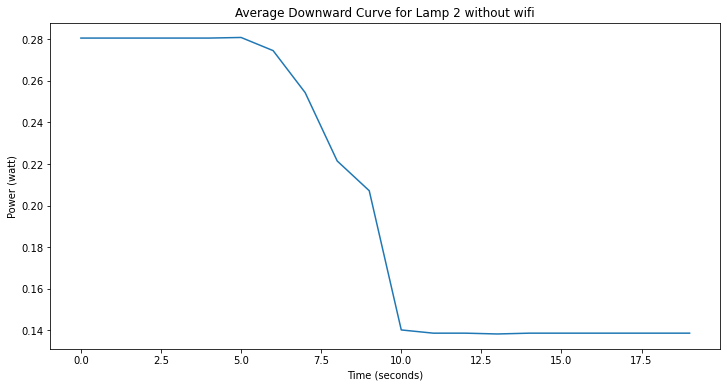
53 منحنی های بالارونده لامپ 2 بدون وایفای



54 منحنی های پایین رونده لامپ 2 بدون وایفای

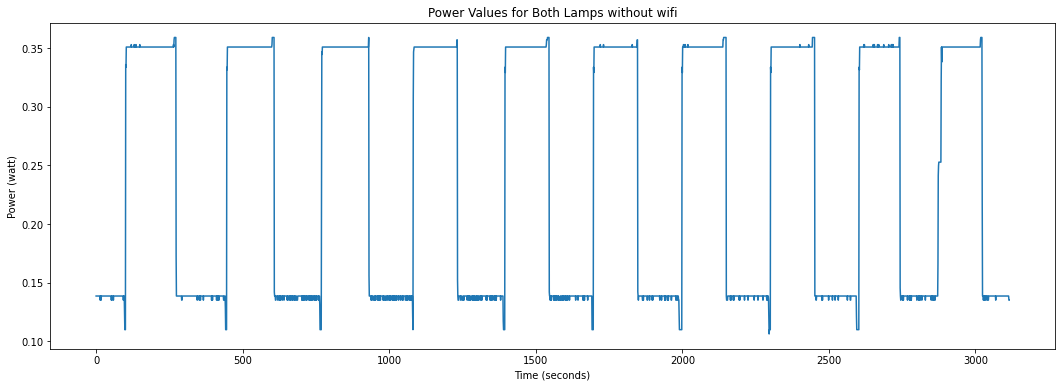


55 منحنی بالارونده میانگین لامپ 2 بدون وایفای

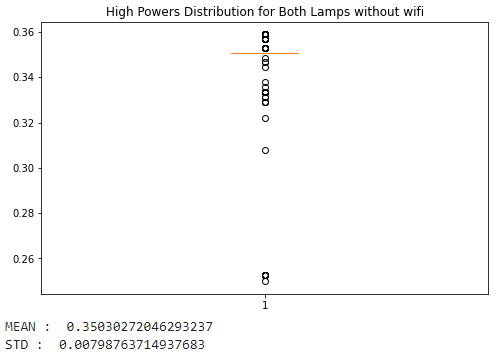


56 منحنی پایین رونده میانگین لامپ 2 بدون وایفای

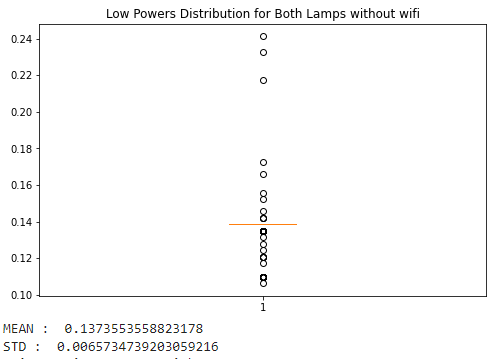
## اعداد و نمودارهای مربوط به روشن شدن هردو لامپ در نبود کد وایفای



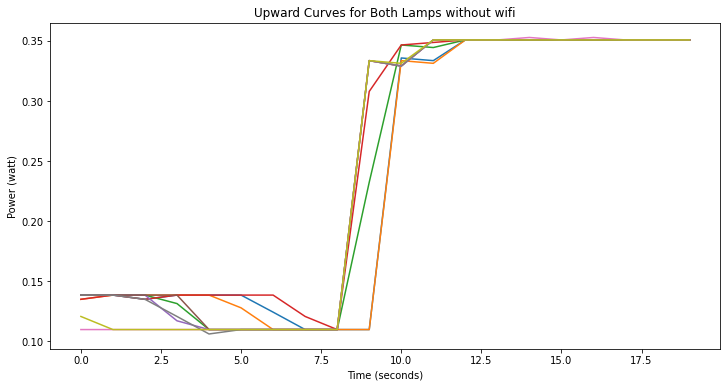
57 توان برحسب زمان هردو لامپ بدون وایفای



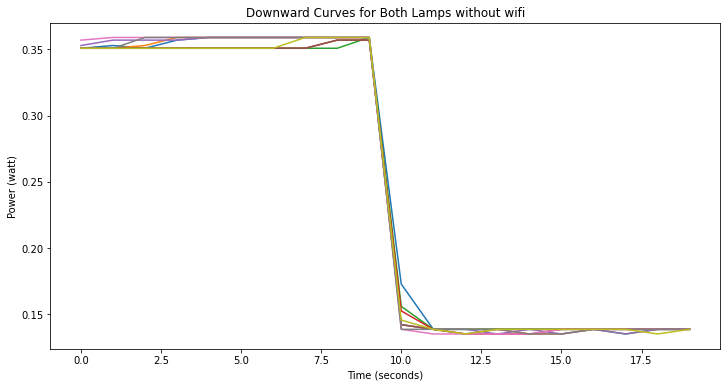
58توزیع داده های مربوط به توان بالا هردو لامپ بدون وایفای



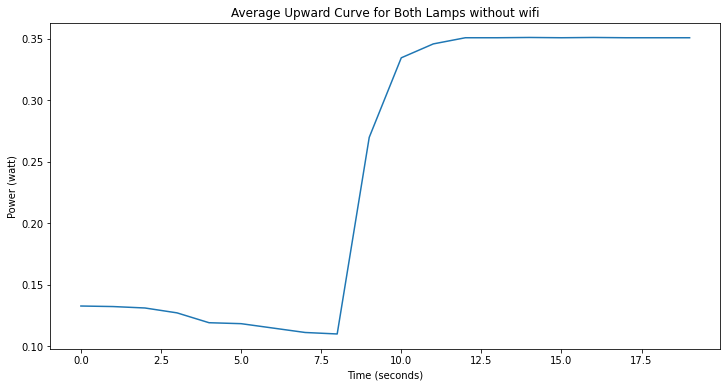
59توزیع داده های مربوط به توان های پایین هردو لامپ بدون وایفای



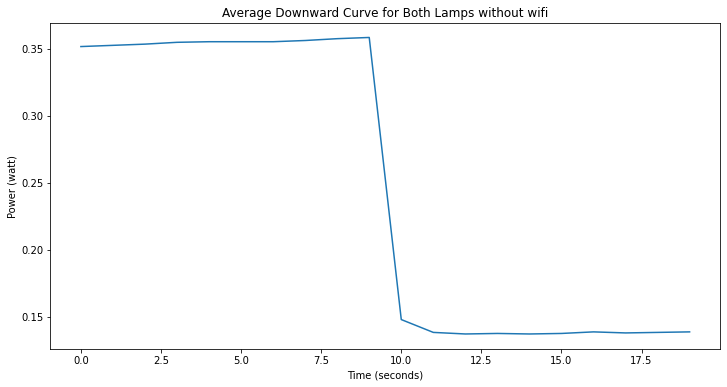
60 منحنی های بالارونده هردو لامپ بدون وایفای



61 منحنی های پایین رونده هردو لامپ بدون وایفای



62 منحنی بالارونده میانگین هردو لامپ بدون وایفای



63 منحنی پایین رونده میانگین هردو لامپ بدون وایفای

# بررسی و تحلیل نتایج

## تحلیل اول

طبق نتایج بدست آمده از قسمت قبل توان مصرفی مدار در سناریوی سوم حداقل 0.274 وات (وقتی هر دو لامپ روشن اند) و حداکثر 0.398 وات (وقتی هر دو خاموش اند) است. در صورتی که بخواهیم مقدار انرژی منبع تغذیه این مدار را اگر بخواهیم طول عمر آن ۲ سال باشد حساب کنیم باید تقریبی از این که هر لامپ به صورت میانگین در روز چند ساعت روشن است داشته باشیم. طبق [این مقاله](https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.576.5491&rep=rep1&type=pdf) در کشور آمریکا در هر خانه به طور میانگین هر لامپ ۱.۵ ساعت روشن می‌ماند. پس طبق محاسبات زیر می‌توان تخمینی از انرژی باتری لازم ارائه کرد.

همچنین می‌توان کمینه و بیشینه مصرف انرژی را ارائه کرد.

برای تهیه یک باتری با این انرژی باید دقت کنیم که باید ولتاژ باتری که مدار را تغذیه می‌کند 3.3v باشد که البته اکثر باتری‌هایی که در این محدوده ولتاژ اند نیروی محرکه برابر با 3.7v دارند. مشخصه دیگر باتری که انرژی باتری وابسته به آن است ظرفیت باتری می‌باشد که از رابطه زیر به دست می‌آید.

اگر از همان تخمین اول برای مقدار انرژی مورد نیاز استفاده کنیم باید ظرفیت باتری باید مقدار زیر باشد.

## تحلیل دوم:‌ بررسی تاثیر ماژول وای فای در مدار

از مقایسه نمودارهای بدون وای فای و با وای فای برای توان تاثیر ماژول وای فای و درخواست های http به سرور در حالتی که لامپ های led خاموش اند به دست آورد. این مقدار از رابطه زیر به دست می آید.

پس بخش اعظم توان مصرفی مدار مربوط به ماژول وای فای است.

تحلیل سوم:‌اندازه گیری طول عمر مدار با باتری معمولی

یک باتری مناسب برای مدار می‌تواند [این باتری](https://www.amazon.com/ADAFRUIT-INDUSTRIES-1578-Lithium-Polymer/dp/B00L0W61VO/ref=sr_1_2?keywords=3.7v+lithium+ion+battery&qid=1661621812&sr=8-2) با مشخصات زیر باشد:

Model Name: ADAFRUIT INDUSTRIES 1578

Technology: Lithium Ion Polymer Battery

Voltage: Output ranges from 4.2V when completely charged to 3.7V

Charge Capacity: 500mAh



باتری 500 میلی امپرساعت64

به طور مثال در صورتی که توان مصرفی مدار حالت میانگین باشد، مدت زمانی که باتری مدار را می‌تواند روشن نگه دارد از روابط زیر به دست می آید.

پس یک باتری معمولی می‌تواند مدار را برای ۶.۵۷ ساعت روشن نگه دارد و برای استفاده های متوالی باید باتری مورد نظر را شارژ نمود.

# جمع بندی

هدف از انجام این پروژه بهبود نتایج به دست آمده از پروژه های قبلی و همچنین تحقیق و بررسی در مورد مصرف توان مدار و یافتن یک باتری مناسب برای مدار بود. با توجه به تخمینی که در بخش تحلیل نتایج انجام گرفت می‌توان نتیجه گرفت که مصرف توان این مدار بیش از حد زیاد بوده که دلیل این امر نیز تا حد زیادی وجود ماژول وایفای و LED ها در مدار است. از آنجا که این اجزای مدار ضروری اند با یک باتری قرصی ۳.۷ ولتی معمولی نمی‌توان مدار را تغذیه کرد و در نتیجه نیاز به یک باتری با ظرفیت بالا هستیم که چنین باتری در بازار وجود ندارد. البته یک راه نیز این است که از UPS استفاده کنیم که هم گران بوده و هم برای یک مدار ساده مانند کلید هوشمند غیر منطقی است.