

عنوان گزارش

ارتباط برد رزبری پای با برد های آردوینو

مقدمه

اگر رزبری پای و خانواده برد های آردوینو را هر کدام به عنوان یک ماشین در نظر بگیریم برای برقراری ارتباط بین آن ها باید از پروتکل هایی برای ارتباط ماشین با ماشین (اصطلاحاً M2M) استفاده کنیم که یکی از مناسب ترین آن ها پروتکل ارتباطی MQTT می باشد که در ادامه به شرح آن و چگونگی برقراری ارتباط می پردازیم. [2]

شرح گزارش

MQTT یک پروتکل ارسال / دریافت است که به دستگاههای لبه شبکه (edge-of-network devices) اجازه می دهد تا با یک سرور (broker) پیغام رد و بدل کنند. هر دستگاه در این شبکه یک نام کاربری دارد و می تواند با هدری خاص (particular topic) یک پیغام را ارسال یا دریافت کند.

MQTT دو طرفه است و اتکاپذیری خود را تضمین می کند به طوری که اگر دستگاه لبه شبکه (subscribed clients)، اتصال خود را از دست بدهد، با استفاده از ویژگی "Last Will and Testament" به سرور (broker) اطلاع داده می شود تا هر دستگاه، بتواند مقدار جدیدی را برگرداند. [3]

در این آزمایش از برد های (Raspberry Pi 3) و (NodeMcu wemos) از خانواده برد های (Arduino) استفاده شده است که هر دوی آن ها دارای Wifi هستند که این ارتباط را در بستر بیسیم فراهم می کنند. همانطور که در گزارش های قبلی اشاره شده است سیستم عامل رزبری پای ابزاری را برای توسعه برنامه نویسی به زبان پایتون فراهم می کند.

این گزارش را به پنج قسمت دسته بندی می کنیم:

۱. در گام اول MQTT Server را بر روی RPi به همراه Library های مورد نیاز نصب می کنیم.
۲. برای ایجاد ارتباط پایا یک ip static برای RPi تعریف می کنیم.
۳. در گام سوم کتابخانه (Library) های مورد نیاز محیط توسعه Arduino IDE را برای استفاده از MQTT نصب می کنیم و کد مورد نظر (به زبان C) را بر روی NodeMcu آپلود می کنیم.
۴. سپس از طرف RPi پیام هایی برای تست ارتباط برای NodeMcu ارسال می کنیم.
۵. در گام آخر فایل مورد نظر با زبان پایتون را بر روی RPi اجرا می کنیم تا ارتباط دو طرفه را میسر کنیم.

۱.

ابتدا وارد ترمینال سیستم عامل رزبین شده و دستورات زیر را به نوبت اجرا می کنیم:

در وهله اول باید یک فایل از لینک زیر دریافت کنیم و آن را در سیستم عامل نصب کنیم (درج sudo در ابتدای دستور اجازه دسترسی کامل را فراهم میکند که معمولاً برای چنین دستوراتی نیاز است) [9]:

```
wget http://repo.mosquitto.org/debian/mosquitto-repo.gpg.key
sudo apt-key add mosquitto-repo.gpg.key
```

سپس با استفاده از دستور cd وارد یک مسیر مشخص می شویم و بعد از آن فایل های مورد نیاز را با دستور wget دانلود می کنیم:

```
cd /etc/apt/sources.list.d/
sudo wget http://repo.mosquitto.org/debian/mosquitto-wheezy.list
sudo wget http://repo.mosquitto.org/debian/mosquitto-jessie.list
sudo wget http://repo.mosquitto.org/debian/mosquitto-stretch.list
sudo apt-get update
```

پس از دریافت فایل ها با استفاده از دستور apt-get ، سرویس MQTT Server را نصب می کنیم:

```
sudo apt-get install mosquitto
```

اگر نصب موفقیت آمیز نبود از دستور زیر استفاده می کنیم:

```
sudo apt --fix-broken install
```

حال MQTT Client را نصب می کنیم:

```
sudo apt-get install mosquitto-clients
```

با استفاده و دستور زیر میتوانیم سرویس نصب شده را تست کنیم:

```
systemctl status mosquitto.service
```

و در صورت انجام موفقیت آمیز مراحل قبل با پیغام پایین مواجه خواهیم شد.

```
pi@raspberrypi:~ $ systemctl status mosquitto.service
● mosquitto.service - Mosquitto MQTT v3.1/v3.1.1 Broker
   Loaded: loaded (/lib/systemd/system/mosquitto.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2019-10-02 18:22:34 +0330; 19h ago
     Docs: man:mosquitto.conf(5)
           man:mosquitto(8)
   Main PID: 427 (mosquitto)
      Tasks: 1 (limit: 2200)
     Memory: 1.9M
    CGroup: /system.slice/mosquitto.service
            └─427 /usr/sbin/mosquitto -c /etc/mosquitto/mosquitto.conf

Oct 02 18:22:34 raspberrypi systemd[1]: Starting Mosquitto MQTT v3.1/v3.1.1 Broker...
Oct 02 18:22:34 raspberrypi systemd[1]: Started Mosquitto MQTT v3.1/v3.1.1 Broker.
```

در این مرحله باید کتابخانه های مورد نیاز (paho-mqtt) [6] را دریافت و بروزرسانی کنیم:

```
sudo apt-get update
sudo apt-get install python python-pip
sudo pip install paho-mqtt
```

۲.

برای تعریف ip static نیاز است در یکی از فایل های config سیستم عامل تغییراتی بوجود آوریم. با استفاده از دستور cd به مسیر فایل مورد نظر می رویم:

```
Cd /etc
```

حال با استفاده از ویرایشگر nano که به صورت پیشفرض بر روی رزبین نصب است فایل "dhcpd.conf" را باز می کنیم:

```
Nano dhcpd.conf
```

و سپس در انتهای فایل باز شده متن زیر را کپی می کنیم:

```

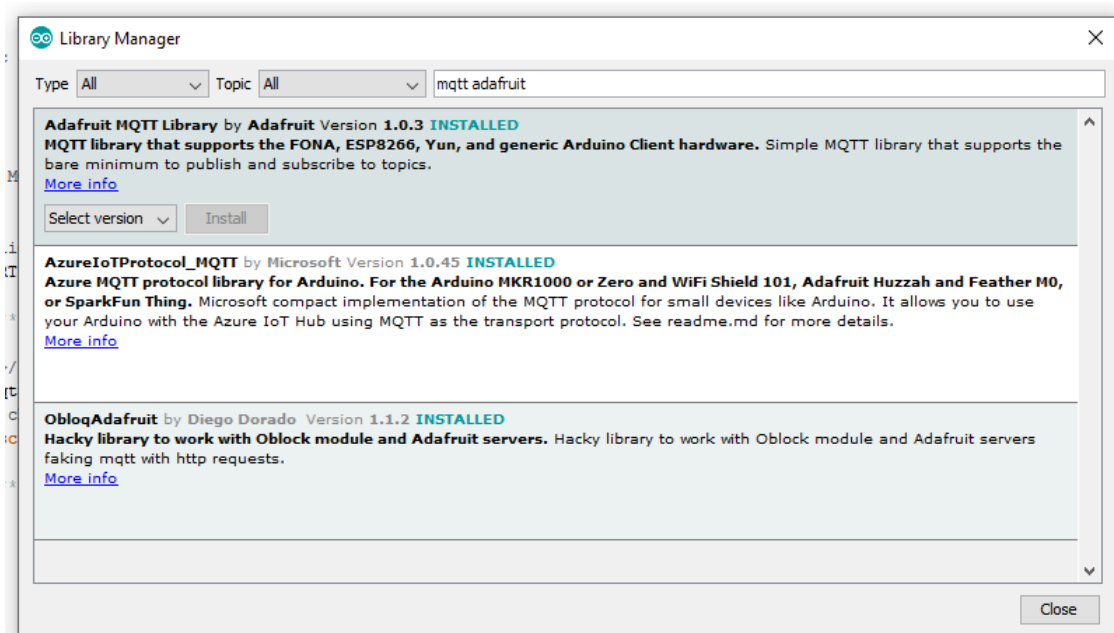
interface eth0
static ip_address=192.168.1.105 // آیپی که استفاده کردیم
interface wlan0
static ip_address=192.168.1.105
static routers=192.168.1.1 // روتر Default gateway
static domain_name_servers=192.168.1.1

```

پس از انجام مراحل بالا فایل را ذخیره کرده و رزبری پای را ری بوت می کنیم.

۳.

در محیط ویندوز Arduino IDE را اجرا کرده و مسیر زیر را طی می کنیم:
 Sketch ==> Include library ==> Manage libraries
 پس از باز شدن پنجره مربوطه در نوار جستجو عبارت "mqtt adafruit" و "sleepydog" را جستجو می کنیم و تمام نتایج را دانلود و نصب می کنیم:



حال فایل [mqtt_arduino_code.ino](#) [1] را بر روی برد NodeMcu آپلود می کنیم.

۴.

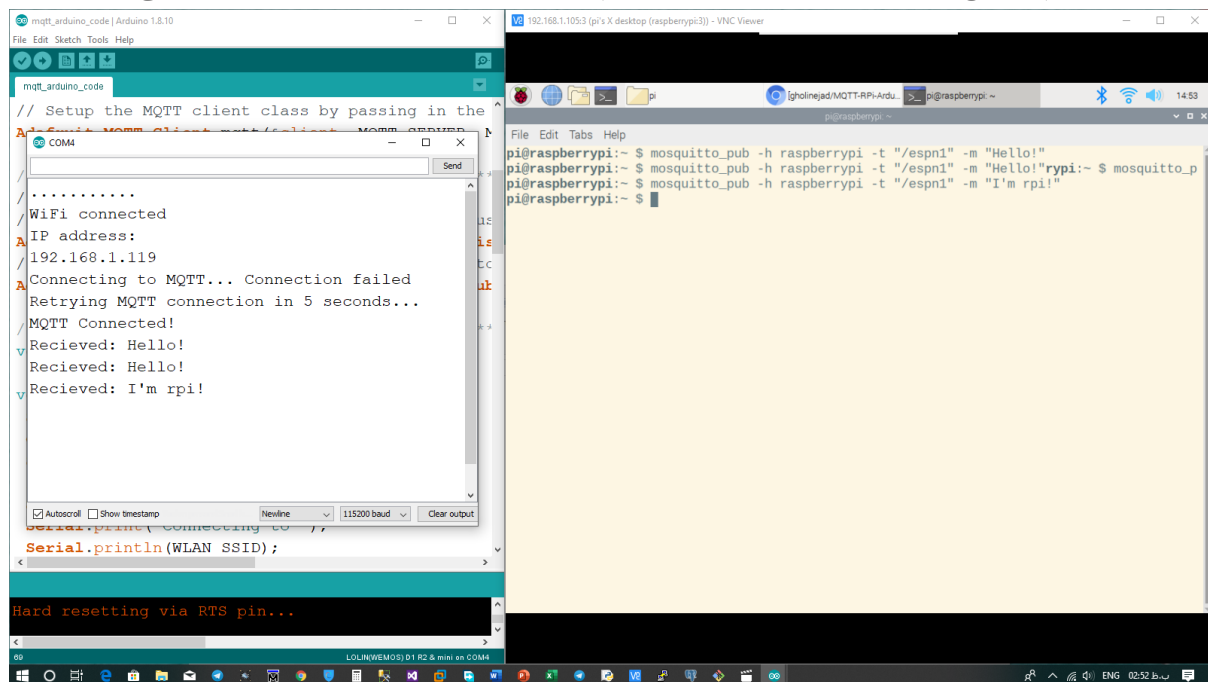
در این مرحله ارتباط یک طرفه یعنی ارسال از طرف rpi و دریافت از طرف NodeMcu را تست می کنیم تا از صحت رابطه مطلع شویم.

در مراحل اولیه سرویس MQTT را بر روی ترمینال رزبین نصب کرده بودیم و حال با استفاده از دستور زیر می توانیم پیغام ها را ارسال کنیم. [4][5]

(نام کاربری NodeMcu را در کد، "/esp1" ذخیره کرده بودیم)

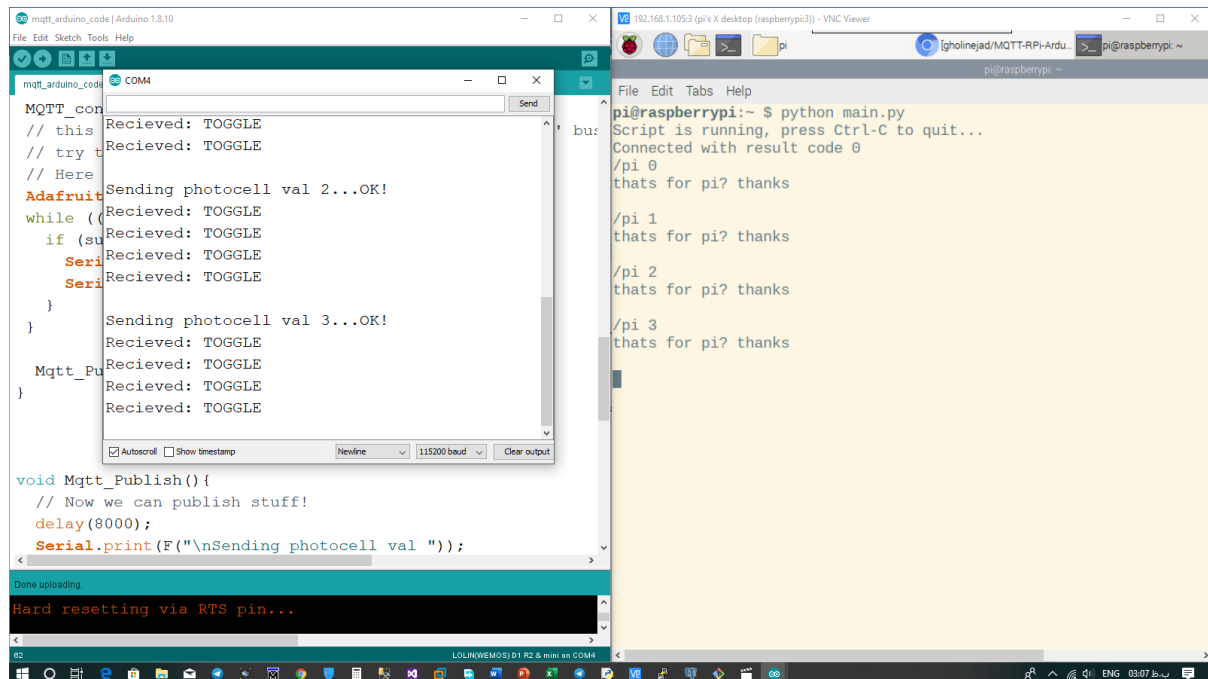
```
mosquitto_pub -h raspberrypi -t "/esp1" -m "Hello!"
```

در صورت انجام صحیح مراحل برد NodeMcu پیغام "Received: Hello!" را نمایش می دهد.



۵.

در این مرحله باید یک اسکریپت پایتون را برای دریافت پیغام از سوی NodeMcu اجرا کنیم. فایل [main.py](#) [1] را اجرا می کنیم و نتیجه را مشاهده می کنیم:



با توجه به () مشاهده می شود هر دو ثانیه یک پیغام "TOGGLE" از رزبری پای ارسال می شود و هر هشت ثانیه یک پیغام "شمارنده یک افزا" از NodeMcu ارسال می شود.

نتیجه گیری

پس از میسر شدن ارتباط دو طرفه باید طبق سناریوی ارتباطی، قوانینی برای رد و بدل کردن پیغام ها در کد ها تعریف شوند. از طرفی ایجاد این زیرساخت کار را برای ارتباط چندین esp به یک رزبری پای را آسان کرده است. تمامی مستندات در لینک گیت هاب [1] بارگذاری شده اند.

مراجع

1. <https://github.com/gholinejad/MQTT-RPi-Arduino>
2. <http://mqtt.org/>
3. <https://www.hackster.io/ruchir1674/raspberry-pi-talking-to-esp8266-using-mqtt-ed9037>
4. https://mosquitto.org/man/mosquitto_sub-1.html
5. https://mosquitto.org/man/mosquitto_pub-1.html
6. <https://pypi.org/project/paho-mqtt/1.1/>
7. <https://openhomeautomation.net/connect-esp8266-raspberry-pi>
8. [https://github.com/adafruit/Adafruit MQTT Library](https://github.com/adafruit/Adafruit_MQTT_Library)
9. <https://mosquitto.org/download/>