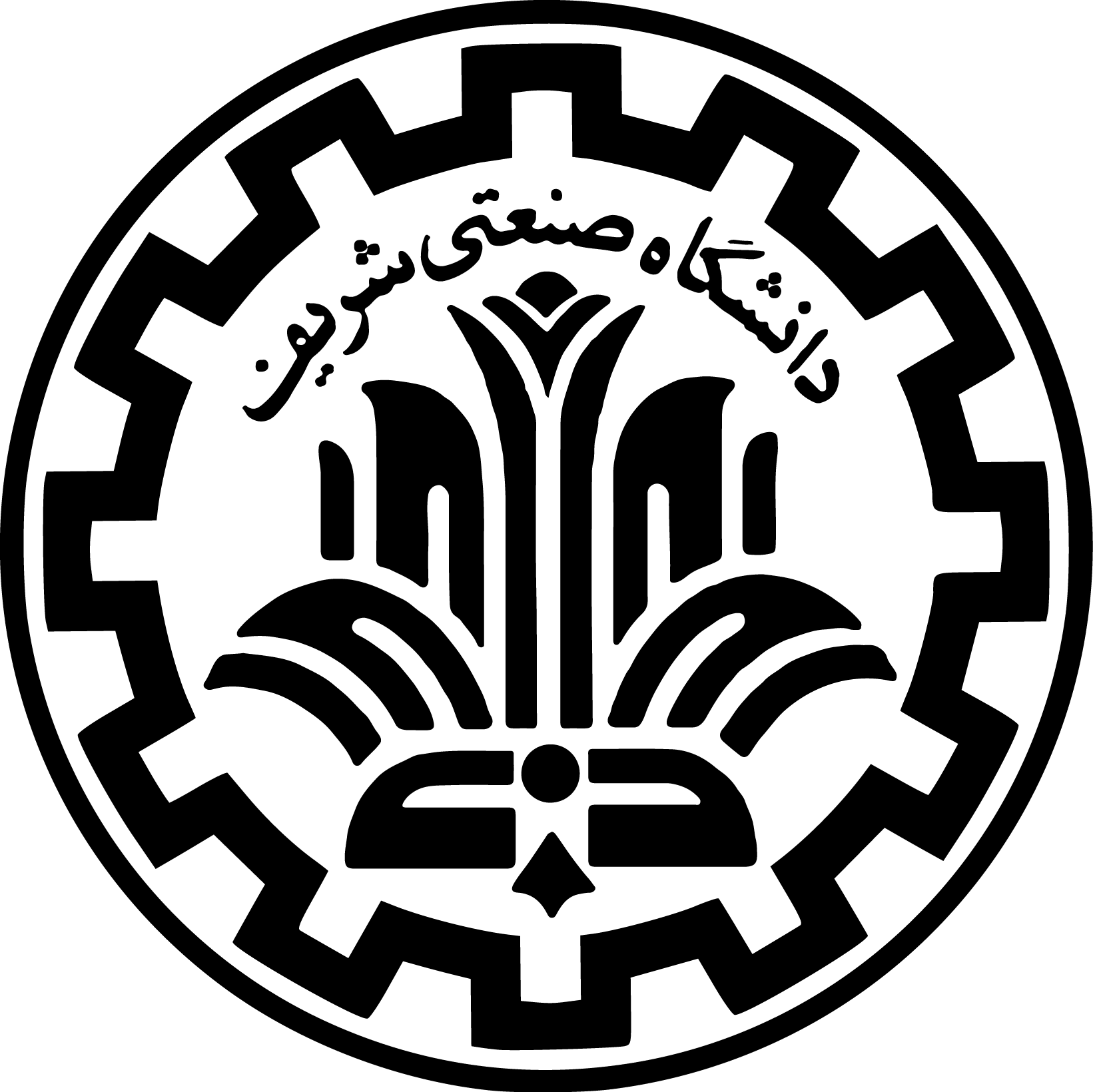
****

آزمایشگاه سخت‌افزار

**گزارش نهایی**

**نیمسال اول ۰۱-۰۲**

موضوع پروژه:

ارتباط رایانه و بورد رزبری‌پای با ماژول‌های **4/4.5/5G** و مقایسه آنها با هم

شماره گروه: **۴**

اعضای گروه:

عرشیا اخوان 97110422

مهدی صادق شبیری 97110144

غزل شناور 97101897

۱ شرح پروژه

این پروژه، یک پروژه تحقیقاتی است که هدف اصلی آن، بررسی و مقایسه عملی میزان تاخیر بسته‌ها در بسترهای ارتباطی 3G و 4G است.

برای اجرای پروژه، از آردوینو و شیلد آردوینو SIM7000C استفاده کردیم. راهکار مورد استفاده ما، راهکار پیشنهادی دوم، یعنی ارسال از طریق برد به سرور و سپس دانلود اطلاعات از سرور به رایانه بود. متاسفانه بدلیل مشکلات این قطعه، پروژه در نهایت با استفاده از آردوینو و ESP8266 انجام شد.

بررسی پروتکل‌ها در دو لایه انجام شد. در لایه چهارم، TCP، UDP و در لایه پنجم، پروتکل‌ HTTPرا بررسی کردیم.

۲ توضیح برخی اصطلاحات

۲.۱. پروتکل TCP

**این پروتکل، یک پروتکل لایه چهارم‌ (لایه انتقال) است.** TCP **اطلاعات را به ترتیب و با چک کردن برای خطا انتقال می‌دهد؛ از همین رو، پروتکل قابل اعتمادی است. این پروتکل،** connection-oriented **است و نیاز به برقراری ارتباط بین سرور و کلاینت توسط یک** handshake **سه مرحله‌ای دارد. این ویژگی پروتکل، امکان تشخیص و تصحیح خطا را فراهم می‌کند اما تاخیر را نسبت به روش** UDP **افزایش می‌دهد. از این پروتکل در** email**، انتقال فایل و بسیاری موارد دیگر استفاده می‌شود.**

۲.۲. پروتکل UDP

**این پروتکل، یک پروتکل لایه چهارم‌ (لایه انتقال) است.** UDP **سرعت را به تصحیح خطا ترجیح می‌دهد. این پروتکل، نیازی به برقراری ارتباط اولیه میان سرور و کلاینت ندارد. این روش هیچ تضمینی درباره ترتیب داده و یا رسیدن آن به مقصد نمی‌دهد. این پروتکل در مواردی استفاده می‌شود که سرعت از تشخیص و تصحیح خطا مهمتر است و یا توسط بقیه اپلیکیشن‌ها انجام می‌شود.**

۲.۳. پروتکل QUIC

**این پروتکل، یک پروتکل لایه چهارم‌ (لایه انتقال) است.** QUIC **با هدف بهبود عملکرد اپلیکیشن‌های** connection-oriented **طراحی شده است و این کار را با برقراری چند ارتباط بر پایه** UDP **انجام می‌دهد. هدف دیگر این پروتکل کاهش تاخیر است. این پروتکل در** 2021 **توسط** IETF **استاندار شد. این پروتکل توسط** Chrome**،‌** Edge**،** Firefox**، و** Safari **پشتیبانی می‌شود.**

۲.۴. پروتکل HTTP

**این پروتکل، یک پروتکل لایه پنجم (لایه اپلیکیشن) است.** HTTP **پایه داده‌ها در** world wide web **است. این روش،‌ از مدل درخواست-پاسخ پیروی می‌کند.**

۲.۵. دستورهای AT

دستور‌های AT مجموعه دستوراتی است برای کنترل کردن ارتباط با مودم (یا دکل). علت این نامگذاری این است که با دستورات AT قرار است attention مودم گرفته شود.

این دستورات ۴ نوع دارند:

**۲.۵.۱. دستورات تست Test commands**

این دستورات برای بررسی پشتیبانی مودم از یک دستور است.

فرمت دستور به این شکل است:

AT<command name>=?

برای مثال

ATD=?

**۲.۵.۲. دستورات خواندن Read command**

این دستورات برای گرفتن تنظیمات گوشی یا مودم است

فرمت دستور به این شکل است:

AT<command name>?

برای مثال

AT+CBC?

**۲.۵.۳. دستورات ست کردن Set commands**

این دستورات برای مقدار دهی تنظیمات گوشی یا مودم است.

فرمت دستور به این شکل است:

AT<command name>=value1, value2, …, valueN

برای مثال

AT+CSCA=”+9876543210”, 120

**۲.۵.۴. دستورات اجرا Execution commands**

این دستورات برای اجرای یک عملیات است.

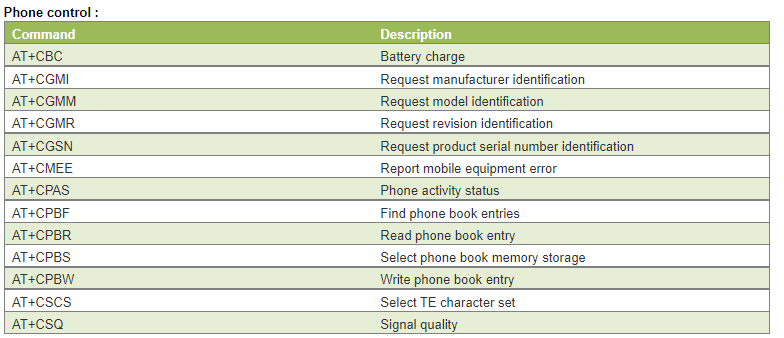
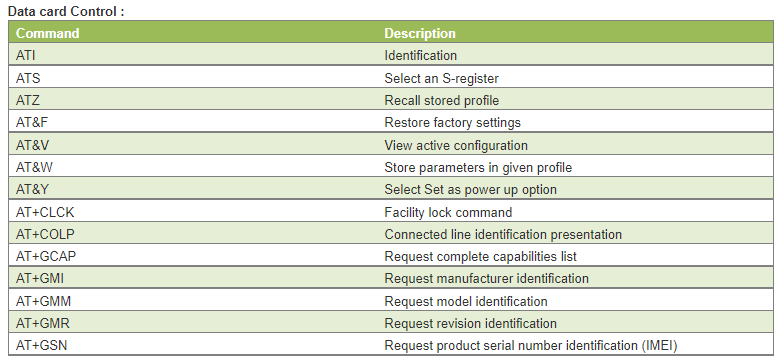
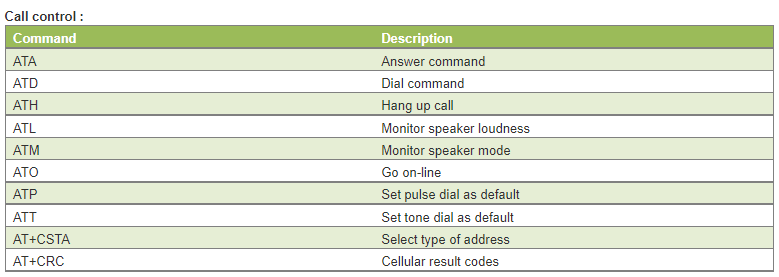
فرمت دستور به این شکل است:

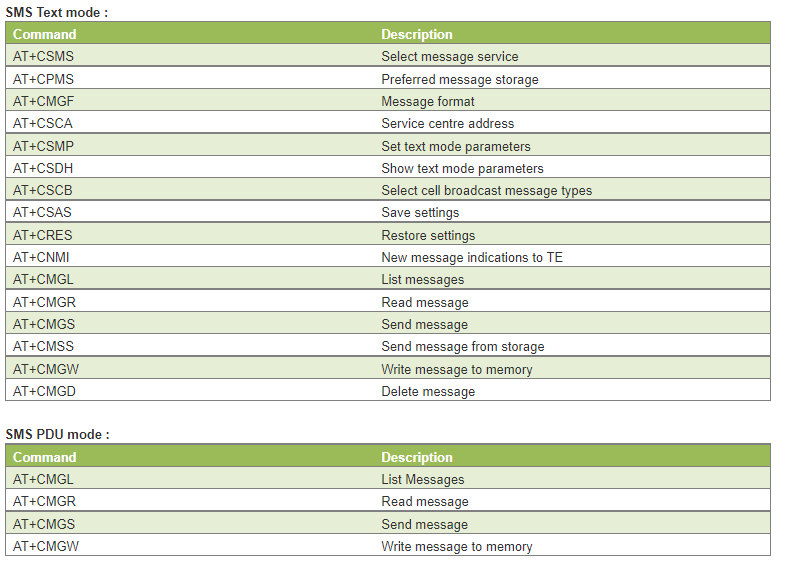
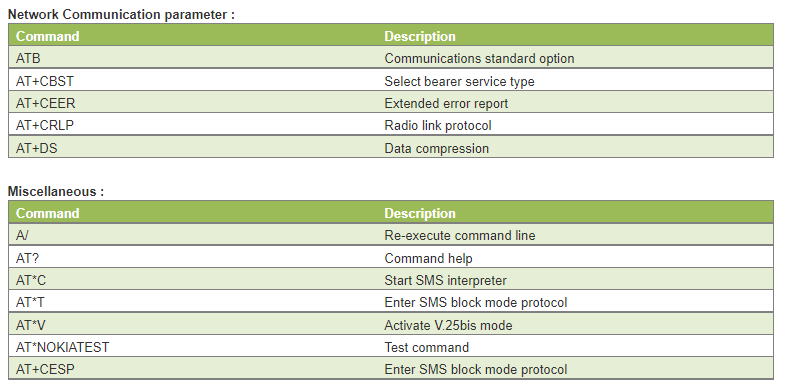
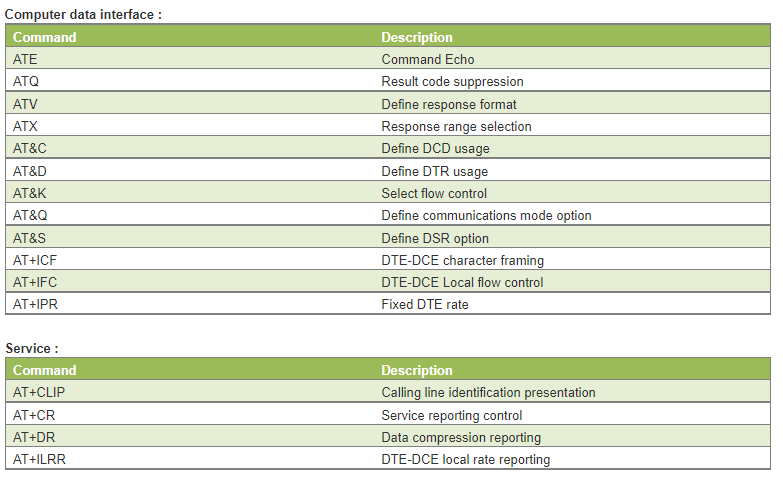
AT<command name>=parameter1, parameter2, …, parameterN

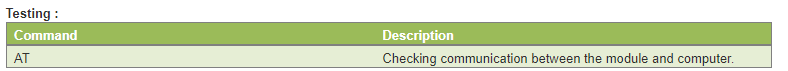
برای مثال

AT+CMSS=1,”+ 9876543210”, 120

در جدول پایین تعدادی از این دستورات آمده است:







۲.۶. نرخ خرابی

**خرابی[[1]](#footnote-1) هنگامی رخ می‌دهد که بسته‌های داده در یک شبکه کامپیوتری به مقصد خود نمیرسند. این اتفاق به دلیل مشکلات موجود در سیستم، و یا ترافیک[[2]](#footnote-2) رخ می‌دهد. نرخ خرابی[[3]](#footnote-3) درصد بسته‌های از دست رفته به کل بسته‌های ارسالی است. نرخ خرابی تاثیر قابل توجهی بر تجربه کاربری دارد به نحوی که ۵ تا ۱۰ درصد نرخ خرابی به طرز قابل توجهی این تجربه را تحت‌الشعاع قرار می‌دهد.**

۲.۷. پهنای باند

**پهنای باند به حداکثر نرخ انتقال داده بر روی یک مسیر اطلاق می‌شود. این ویژگی معمولا بر اساس تعداد** bit **های منتقل شده بر ثانیه اندازه گرفته‌ می‌شود.**

۲.۸. تاخیر[[4]](#footnote-4)

**تاخیر در شبکه، به زمانی اطلاق می‌شود که بسته در مسیر عبور از فرستنده به گیرنده قرار دارد. تاخیر نیز تاثیر قابل توجهی بر تجربه کاربری دارد.**

۳ لیست آزمایش‌ها

**آزمایش‌های ما شامل ۴ مورد می‌شوند:**

* **بررسی پروتکل** TCP
* **بررسی پروتکل** UDP
* **بررسی** QUIC **بر پایه** UDP
* **بررسی** HTTP **بر پایه** TCP

**برای هر یک از چهار آزمایش،‌ سه متغیر نرخ خرابی، تاخیر و پهنای باند با انجام چند تست و سپس میانگین‌گیری میان نتایج محاسبه می‌شوند.**

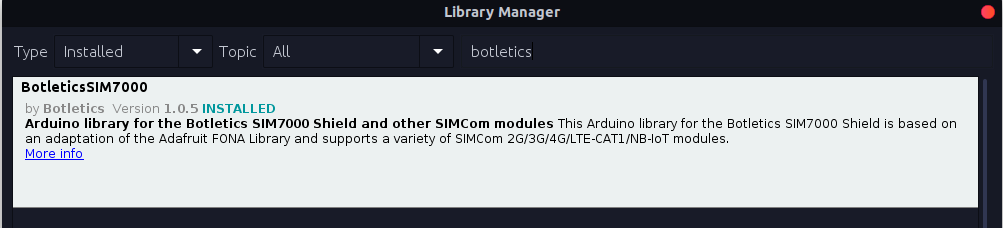
**در این مرحله، تصمیم ما بر این شده است که از** packet generator **ها استفاده نکنیم؛ چرا که هیچ‌کدام از آنها هر چهار آزمایش ما را پوشش نمی‌دهند و استفاده از چند ابزار مختلف می‌تواند باعث ایجاد ناهماهنگی در نتایج آزمایش شود.**

۴ **آزمایش با Sim7000**

۴.۱. کتابخانه‌های مورد استفاده

**۴.۱.۱. کتابخانه‌ی Botletics-SIM7000**

برای تست کردن این پروژه از کتاب‌خانه‌ی  Botletics-SIM7000 استفاده شده است. ([لینک به گیت‌هاب](https://github.com/botletics/Botletics-SIM7000)) این کتاب‌خانه بر اساس کتاب‌خانه‌ی Adafruit FONA طراحی شده است و از ماژول‌های SIM5320، SIM7000، SIM7070، و SIM7500 پشتیبانی می‌کند.



شکل 1: کتابخانه استفاده شده برای اتصال به اینترنت

۴.۲. تست کارکرد قطعه

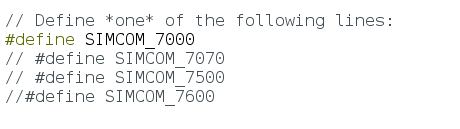
برای تست کردن ماژول از مثال SIM7XXX\_LTE\_Demo از کتاب‌خانه‌ی گفته شده استفاده شده است.

برای بارگذاری این کد باید از مسیر File -> Examples -> BotleticsSIM7000 -> SIM7XXX\_LTE\_Demo آن را انتخاب کرد.

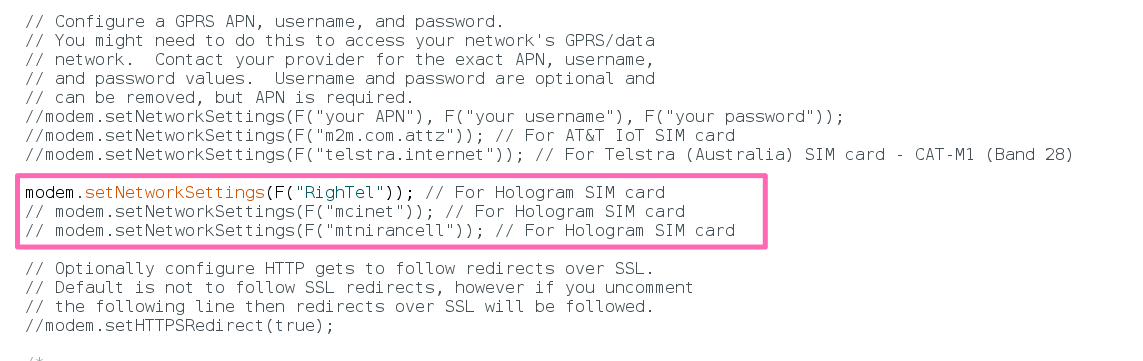
سپس باید define های TX و RX را روی ۸ و ۷ قرار داد.

https://lh5.googleusercontent.com/OD0o5C-LuMhjahRJSDl7xboixhv92DNgin_WNzWE0AAHW9c4nmnLJfpqSAx5da0umxvDbd6rKaXfqCieYsIsKE3vJz5WyXIy_ARXyDWYXLXcBk7l_pStkniJT_ZacNjH9m2VG5m8gzHVKF-j4dtK75GBBhXcWjZ8MudxlqYtpE7BjFYBhKqew2W3BFWumA

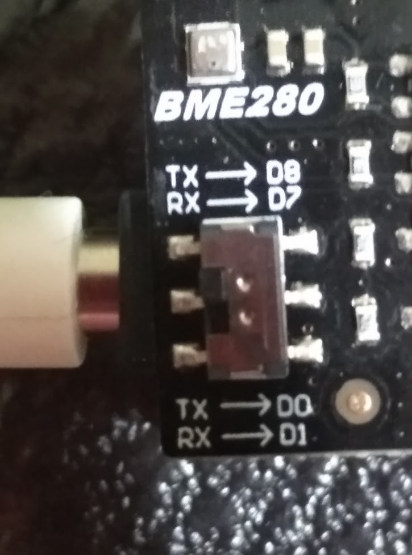
با توجه به اینکه ماژول ما SIM7000 است باید این مدل را انتخاب کرد و ۴ define دیگر را کامنت کرد.



در نهایت باید APN مناسب با سیمکارت را انتخاب کرد

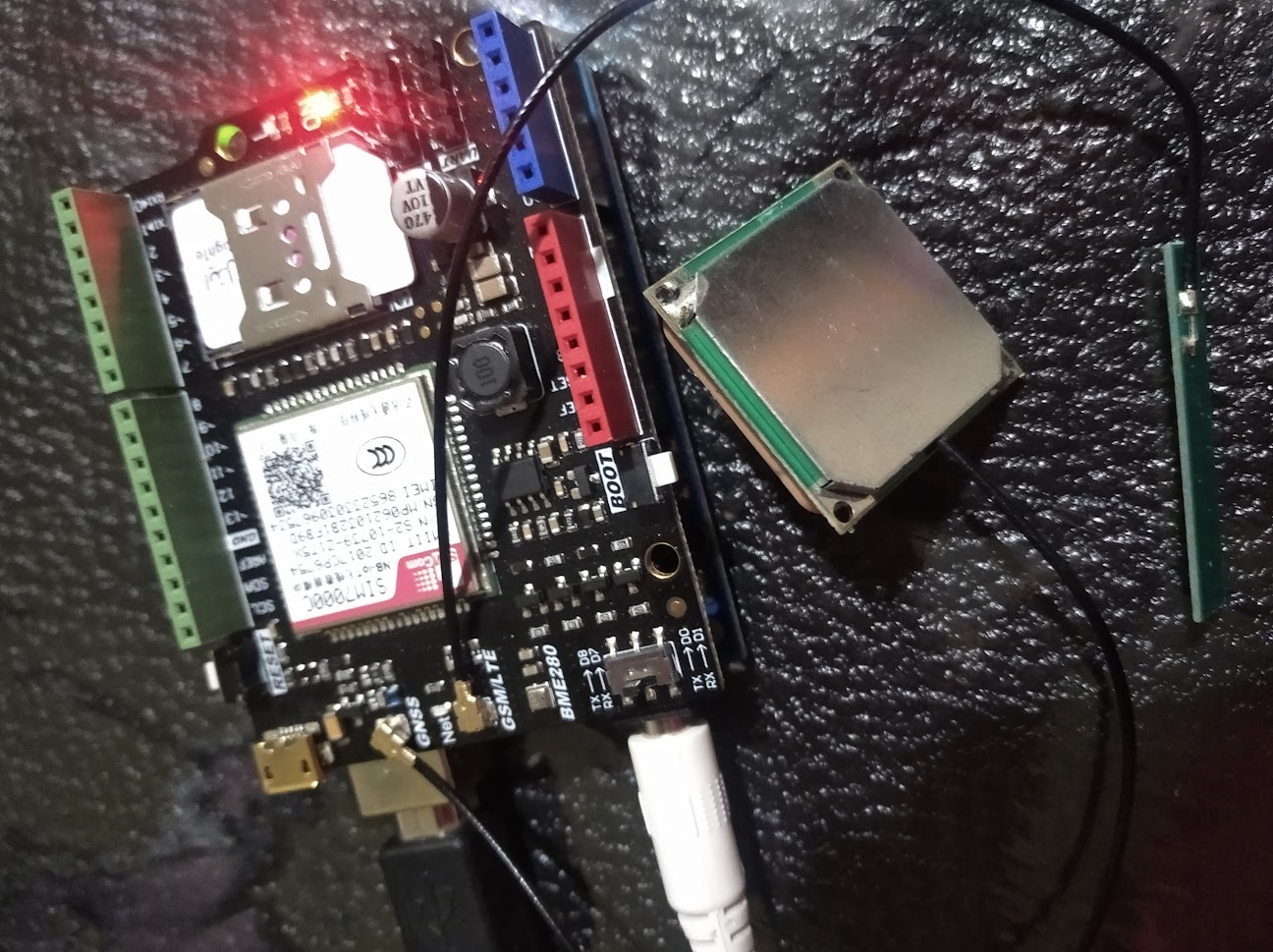
که برای رایتل RighTel، همراه اول mcinet ،و ایرانسل mtnirancell است.

چون TX و RX را پایه ۸ و ۷ قرار دادیم سوئیچ پاور را باید روی ۷ و ۸ تنظیم کنیم. مانند شکل زیر:



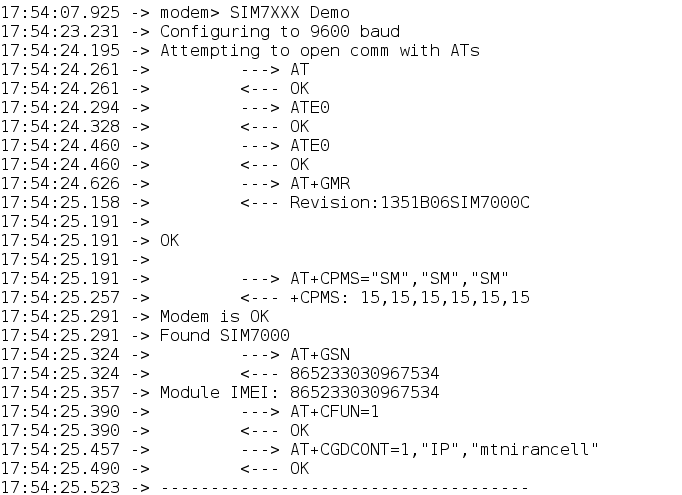
شکل 2: تنظیمات سوییچ پاور

شیلد SIM7000 شامل یک آنتن برای اینترنت و یک آنتن برای GPS، یک منبع تغذیه (سیم سفید) ،و پورت سریال (سیم سیاه) است.

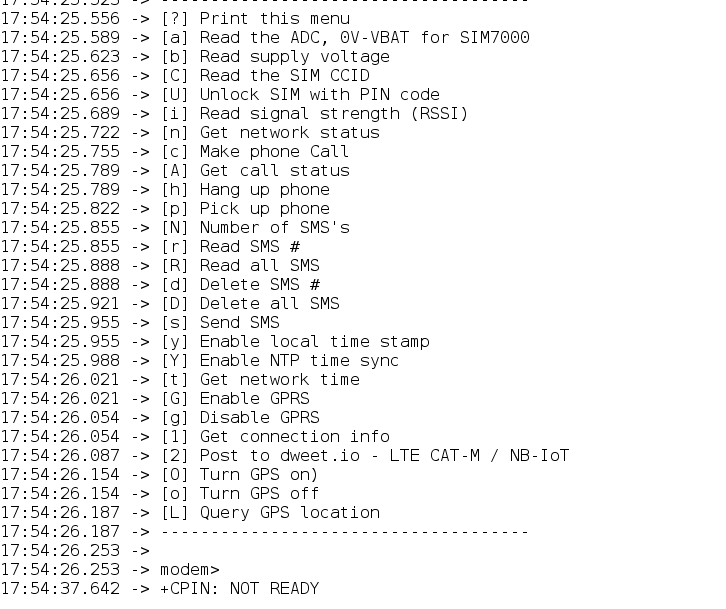


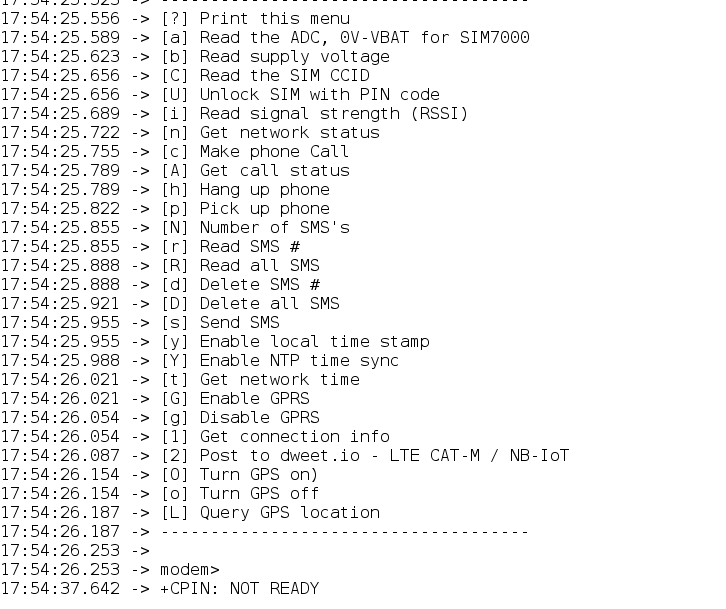
شکل 3: شیلد به همراه آنتن‌ها

بعد از اجرای کد ابتدا به مودم (دکل) وصل می‌شود که لاگ‌های آن در زیر مشاهده می‌شود.

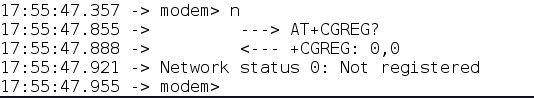


سپس منوی زیر مشاهده خواهد شد.





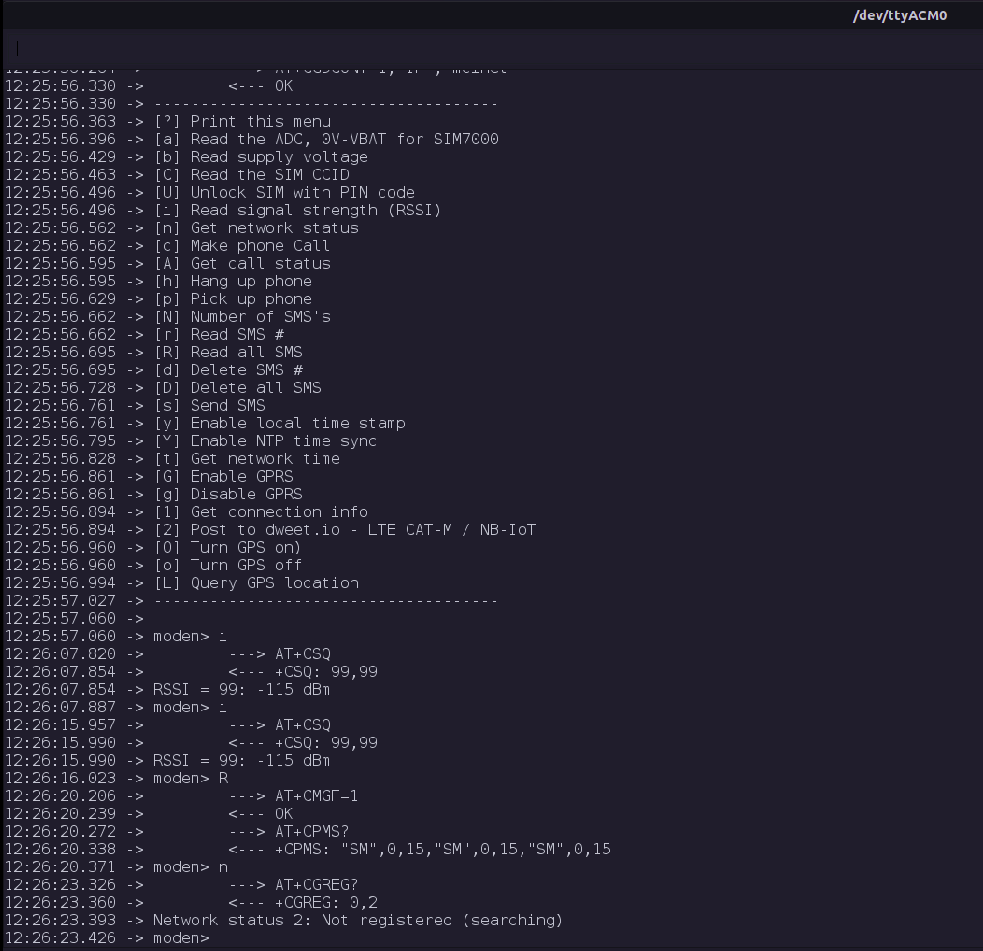
در صورتی که n یعنی Get network status را صدا بزنیم خروجی زیر را می‌دهد.



با وجود تلاش‌های بسیار تیم، و نظر به محدودیت‌های اعمال شده بر روی سامانه مخابرات کشور، پروژه‌ی این گروه با شکست مواجه شد. طبق بررسی‌های انجام شده توسط تیم توسعه پروژه، متاسفانه امکان اجرای طرح تحقیقاتی ارائه شده وجود ندارد.

به طور دقیقتر ماژول Sim7000 در ایران به درستی کار نمی‌کند و حتی خطا‌های درستی نیز نمی‌دهد که بشود آن‌ها را رفع کرد.

موقعی که سیمکارت داخل ماژول قرار می‌گیرد در شبکه Register نمی‌شود و این باعث می‌شود امکان استفاده از آن وجود نداشته باشد.



تصویر بالا از اجرای کد hardware.ino به دست آمده است. همانطور که دیده می‌شود وضعیت شبکه سیم کارت در حالت Not Registered می‌ماند.

برای همین به جای استفاده از ماژول sim7000 از ماژول ESP8266 استفاده کردیم.

# **۵** **آزمایش ESP8266**

**۵.۱. سرور**

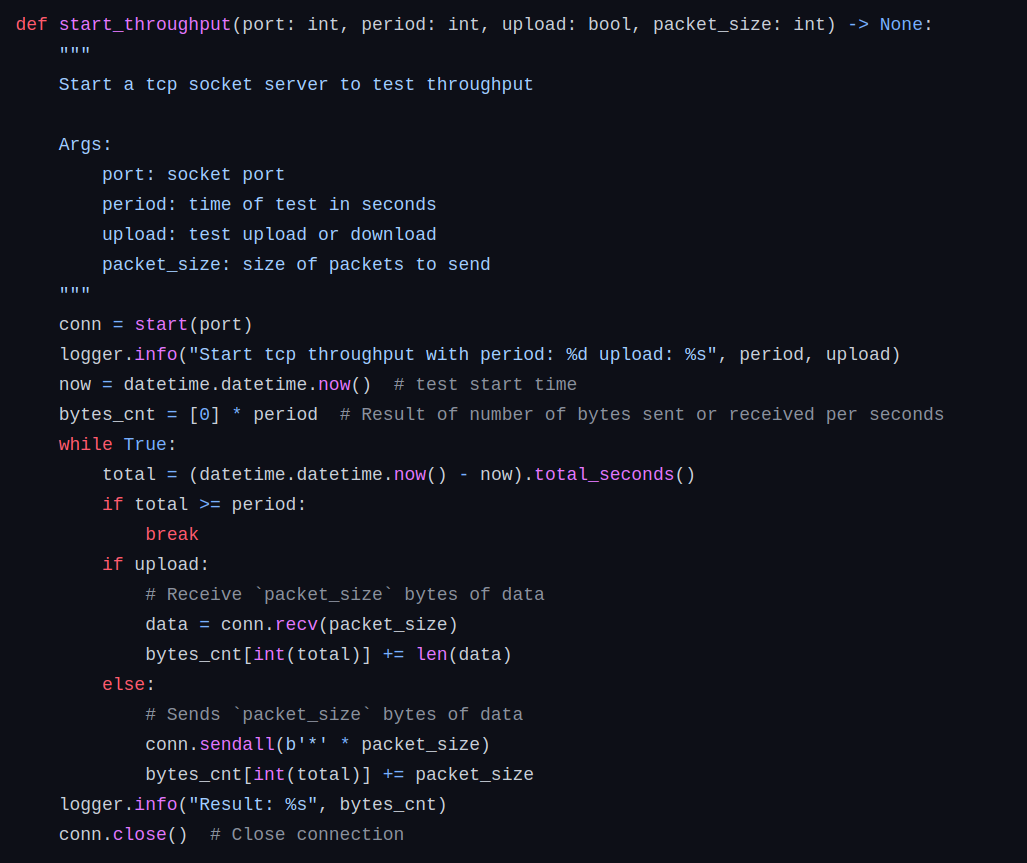
قسمت سرور کاملا زده شده و الان از TCP, UDP, Quic, و HTTP پشتیبانی می‌کند و برای هر کدام Upload, Download, و Latency محاسبه می‌شود.

## **TCP۵.۱.۱.**

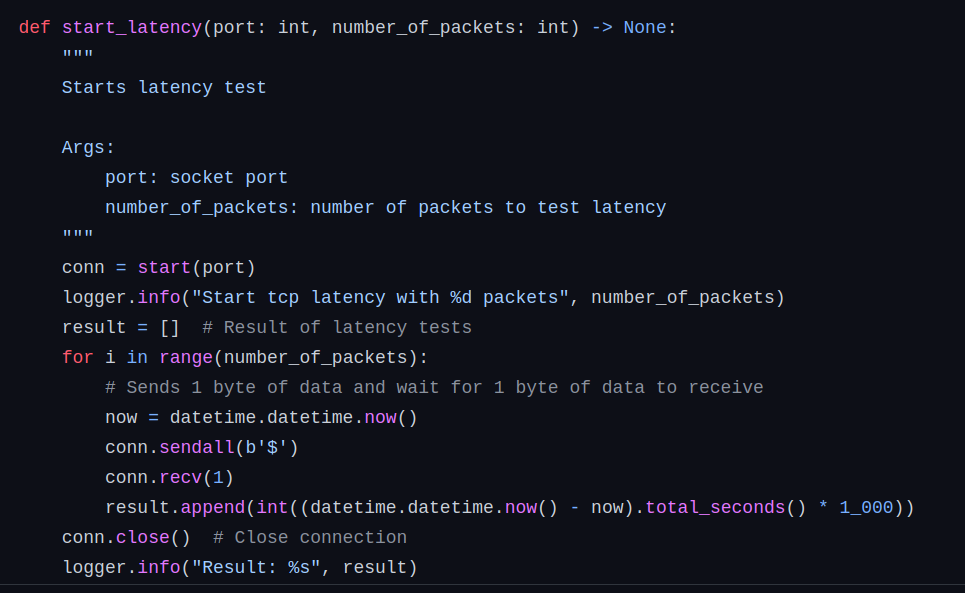
کد‌های مورد نظر در فایل tcp.py هستند.



تابع start یک سوکت از نوع TCP می‌سازد و در ورودی port مربوط به سوکت را می‌گیرد.



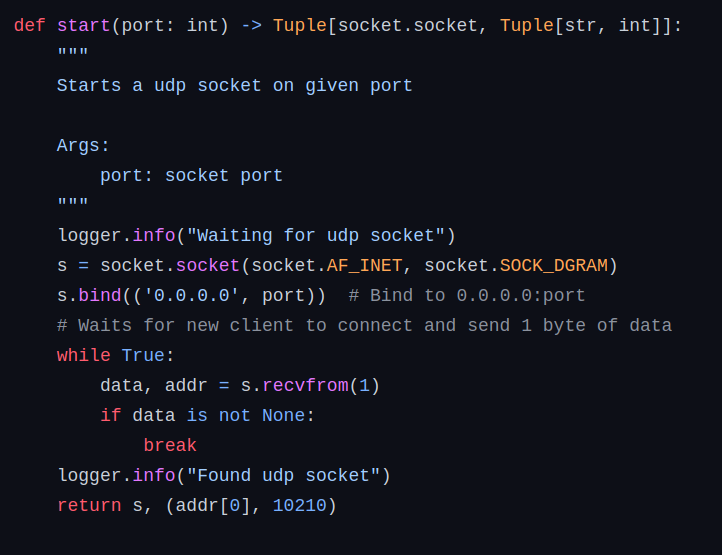
تابع start\_throughput برای راه اندازی سرور upload, download است و در ورودی با period مشخص می‌شود که در چند ثانیه تست را انجام دهد و با بولین upload مشخص می‌شود هدف سنجش آپلود است یا دانلود.

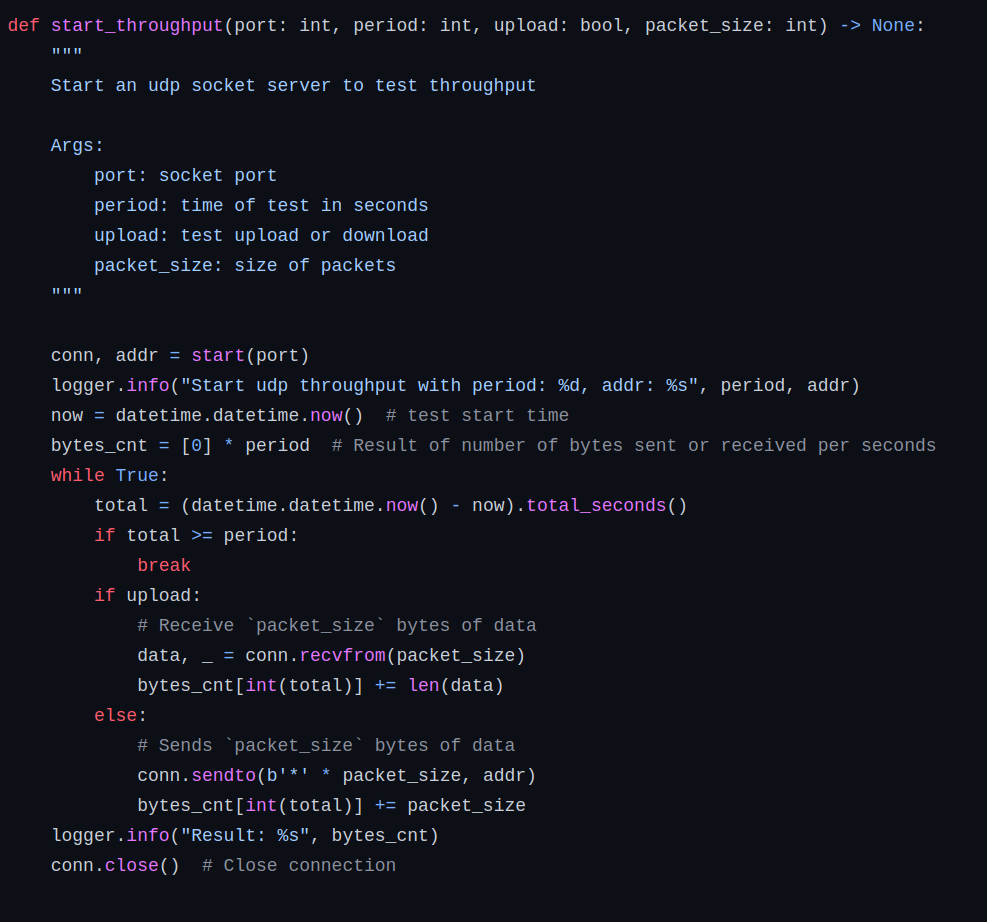


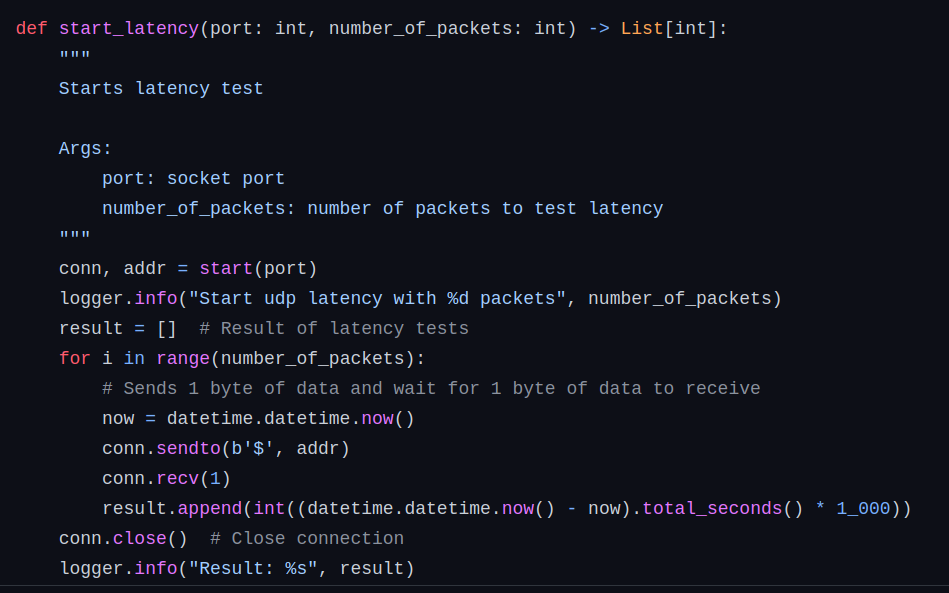
تابع start\_latency برای شروع سرور محاسبه تاخیر است. در ورودی number\_of\_packets برابر تعداد پکتی است که می‌فرستد تا latency را حساب کند. این بخش به این صورت عمل می‌کند که سرور یک پکت به کلاینت می‌فرستد و کلاینت لحظه‌ای که پکت را دریافت کرد یک پکت به سرور می‌فرستد.

## **UDP۵.۱.۲.**

با توجه به اینکه کد در زبان پایتون زده شده و سوکت UDP, TCP شبیه هم هستند کد‌های مربوط به قسمت UDP نیز شباهت زیادی با TCP دارند.



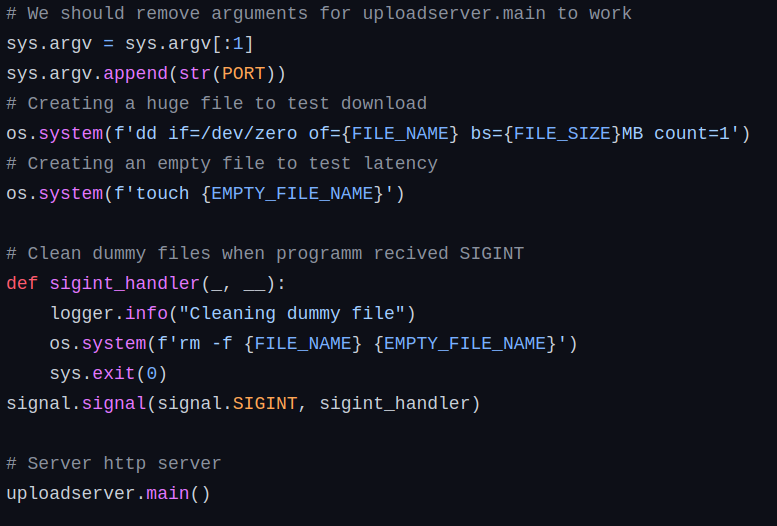


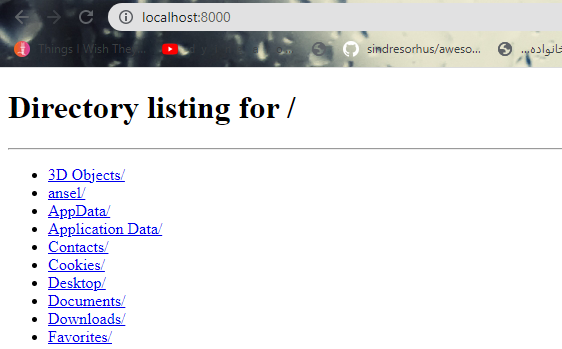


به طور مشابه ۳ بخش TCP هر بخش کار گفته شده را انجام می‌دهد.

## **HTTP۵.۱.۳.**

در این بخش با استفاده از لایبری uploadserver یک سرور http راه اندازی می‌کنیم. همچنین یک فایل حجیم برای تست Download و یک فایل خالی برای تست Latency می‌سازیم.





تصویر بالا از http server است.

## **۵.۲. توضیحات کد سخت افزار**

برای این آزمایش نیاز به کتاب‌خانه‌های ESP8266HTTPClient, ESP8266WiFi و WiFiUdp داریم که هر ۳ کتاب‌خانه built-in هستند.

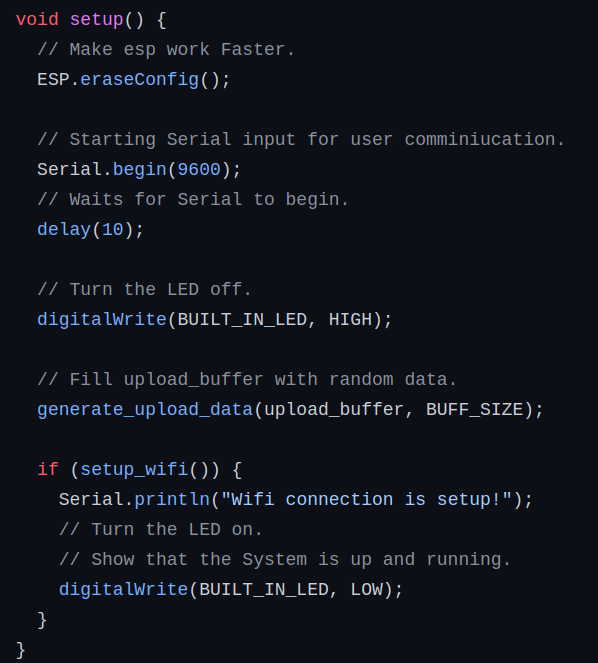
### **۵.۲.۱. تابع setup\_wifi**

این تابع برای اتصال به access point است. کد این تابع در صفحه بعد قابل دسترسی است.



### **۵.۲.۲. تابع setup**

آردوینو هنگام اجرا تابع setup را اجرا می‌کند. سریال برای ورودی و خروجی را روی 9600 تنظیم می‌کنیم و LED را در صورتی که با موفقیت به Access point وصل شد روشن می‌کنیم.

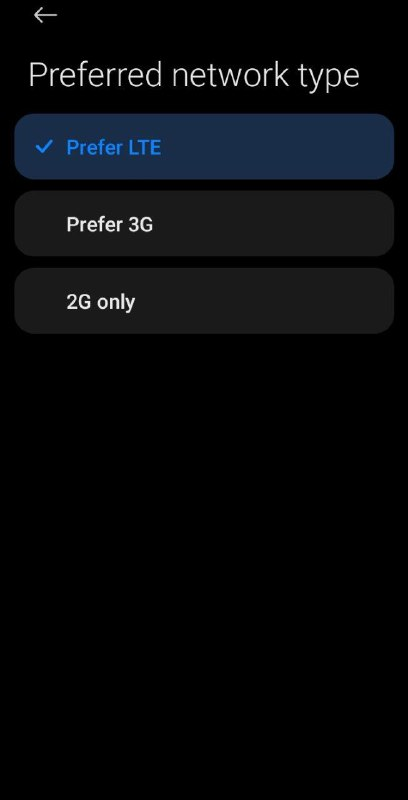
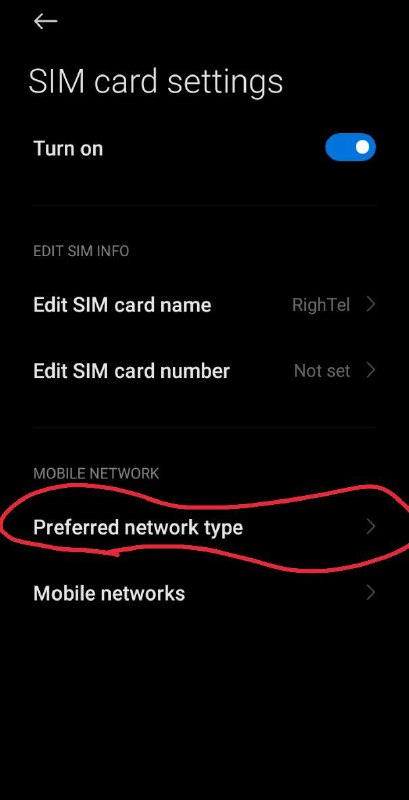


## **۵.۲.۳. توضیحات آزمایش**

با استفاده از ماژول ESP8266 به hotspot وصل می‌شویم و سپس تست‌های لازم را انجام می‌دهیم.

### **۵.۲.۴. هات اسپات**

برای مشخص کردن نوع اینترنت در گوشی اندروید باید به قسمت SIM cardsd & mobile network رفت سپس از قسمت Preferred network type نوع اینترنت را مشخص می‌کنیم.



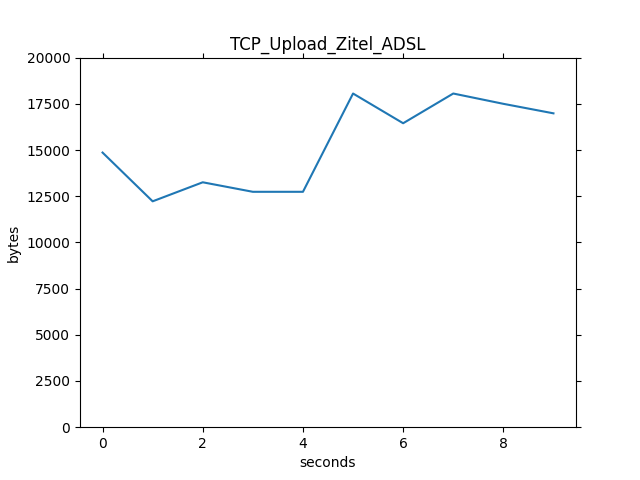
# **۶** **نتایج و نمودارها**

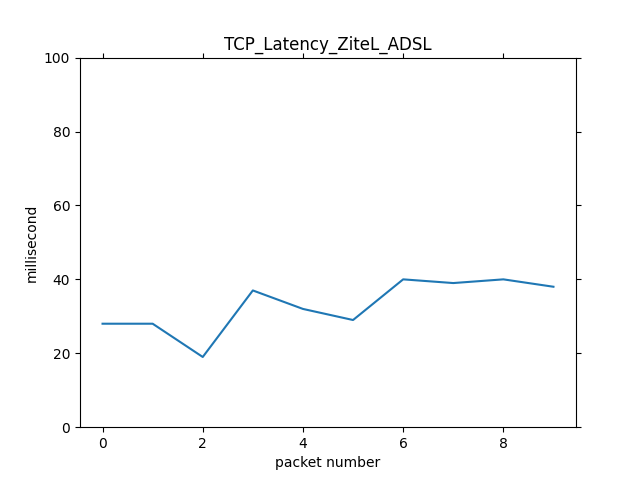
حدود ۳۵ آزمایش مختلف، بر روی سه ارایه دهنده اینترنت رایتل، همراه اول و زیتل انجام شد. این آزمایش‌ها شامل دانلود و آپلود با استفاده از سه پروتکل TCP،‌ UDP و HTTP بودند. همچنین برای همراه اول، نسل‌های چهارم، سوم و دوم و برای رایتل،‌ نسل‌های چهارم و سوم آزمایش شدند. نتایج این آزمایش‌ها را در نمودار‌های ادامه گزارش مشاهده خواهید کرد.

## **۶.۱. ارائه دهنده زیتل**

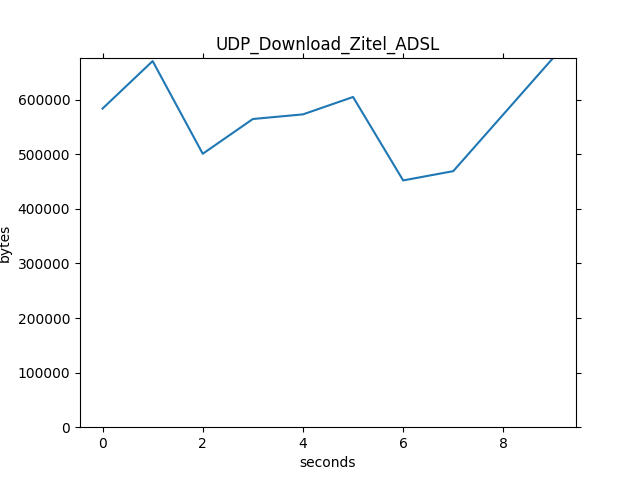
۶.۱.۱. نمودارهای هر آزمایش

نمودار بالا،‌ bandwidth را برای دانلود با استفاده از روش TCP برای این ارایه‌دهنده نمایش می‌دهد. میانگین سرعت دانلود حدود 60 کیلوبایت بر ثانیه است.

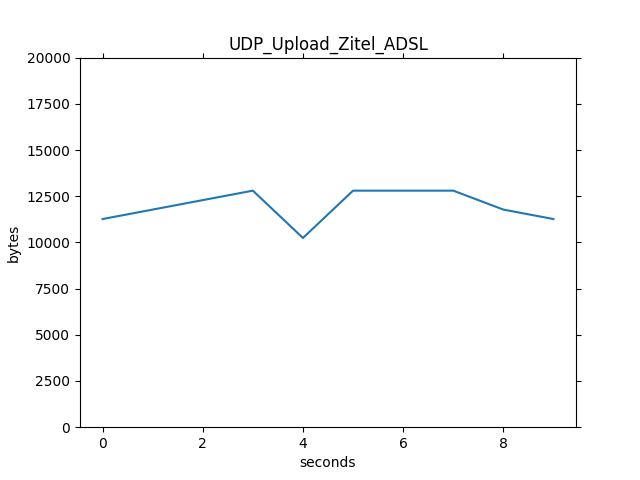
نمودار بالا سرعت آپلود را برای این ارایه‌دهنده نمایش می‌دهد. میانگین سرعت آپلود برای روش TCP، حدود 15 کیلوبایت بر ثانیه است.



میزان تاخیر در حین دانلود و آپلود برای روش TCP در نمودار بالا نمایش داده شده است. میانگین تاخیر در حدود 35 میلی‌ثانیه بوده است.

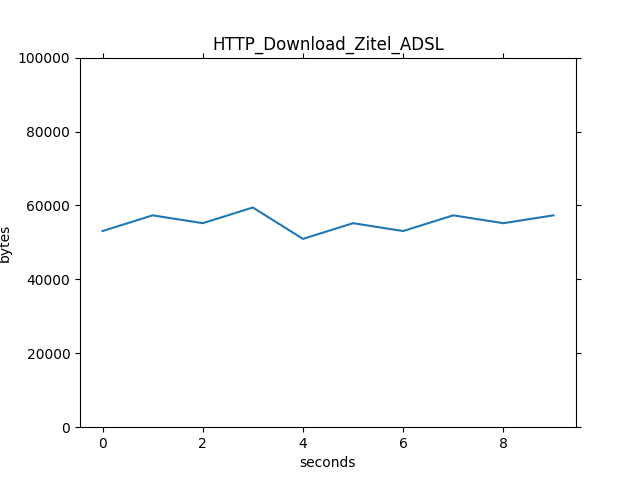


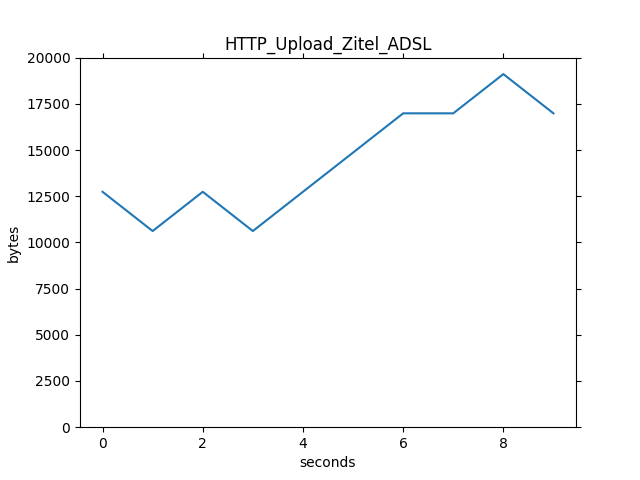
نمودار، نشان‌دهنده سرعت دانلود UDP برای این ارایه‌دهنده است. سرعت دانلود این ارایه‌دهنده برای پروتکل UDP حدود 550 کیلوبایت است.

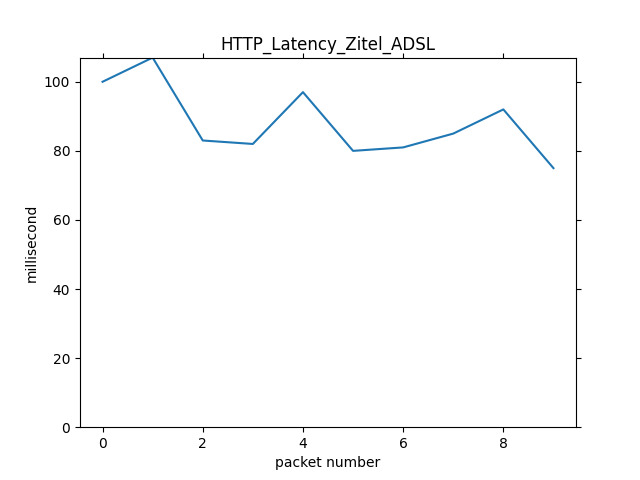


نمودار نشان‌دهنده سرعت آپلود با استفاده از روش UDP است. Bandwidth این شرایط میانگین 11 کیلوبایت را داراست.

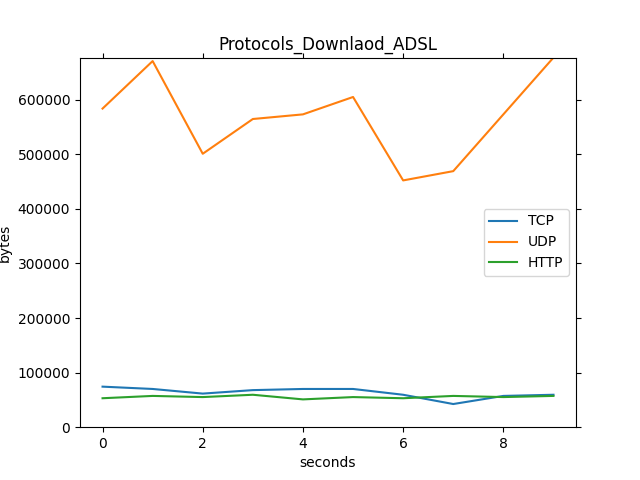
تاخیر روش UDP برای این ارایه‌دهنده،‌ میانگین 40 میلی‌ثانیه را داراست.

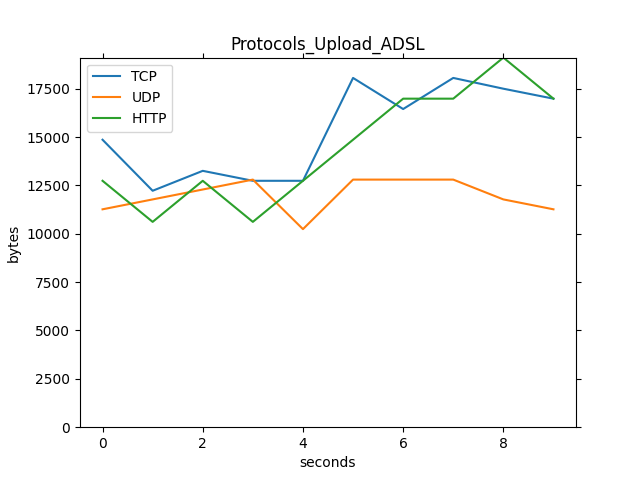
این نمودار، سرعت دانلود برای پروتکل HTTP را نشان می‌دهد. این پروتکل، بیشترین ثبات را داراست و میانگین 58 کیلوبایت را دارد.

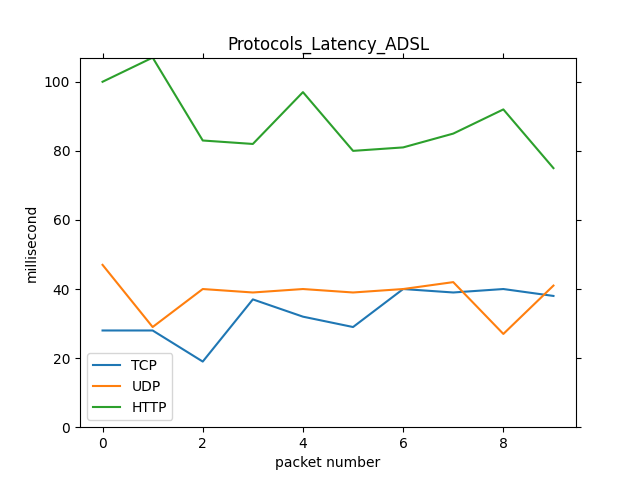
این نمودار، سرعت آپلود را برای پروتکل HTTP نشان می‌دهد. میانگین این نمودار، 15 کیلوبایت است.

این نمودار، تاخیر روش HTTP را برای این ارایه‌دهنده نمایش می‌دهد. تاخیر این روش، از UDP و TCP بیشتر است و میانگینی حدود ۹۰ میلی‌ثانیه دارد.

### **۶.۱.۲. مقایسه پروتکل‌های مختلف**

سرعت دانلود روش‌های مختلف در این نمودار با هم مقایسه شده‌اند. همانطور که مشاهده می‌شود، سرعت دانلود HTTP و TCP بسیار مشابه هم است که با توجه به اینکه روش HTTP بر پایه TCP عمل می‌کند این ارزیابی قابل پیش‌بینی است. سرعت دانلود UDP از دو روش دیگر بسیار بیشتر است که به همین دلیل است که در استریم ویدیو از این روش استفاده می‌شود.

سرعت آپلود این سه پروتکل در نمودار بالا با هم مقایسه شده‌است. مشابه نمودار قبل، نمودار TCP و HTTP مشابه یکدیگر است. میانگین آپلود UDP از سه روش دیگر پایینتر است.

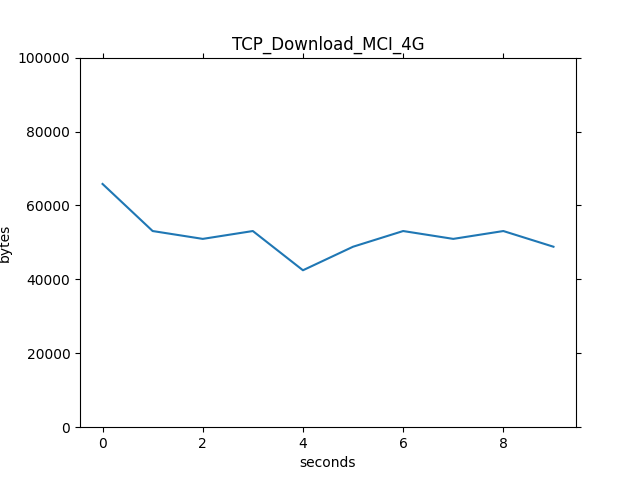


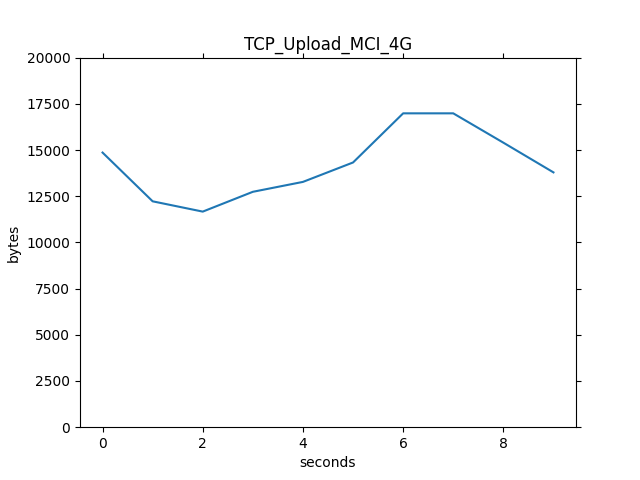
تاخیر سه روش در نمودار بالا مقایسه شده‌است. بیشترین تاخیر در پروتکل HTTP دیده می‌شود. تفاوت این پروتکل با TCP بدلیل overhead آن است.

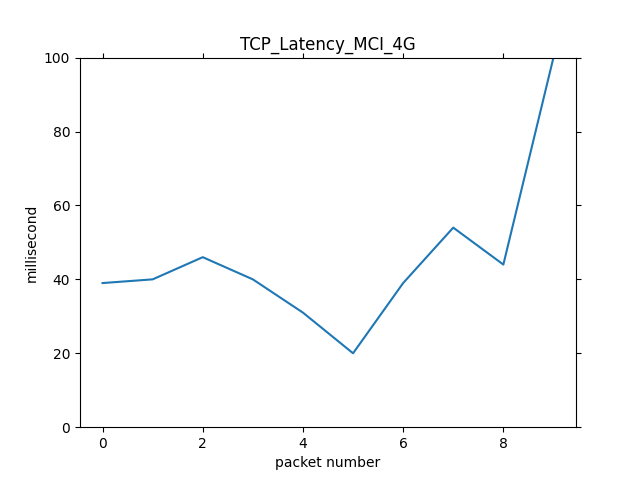
## **۶.۲. ارائه دهنده همراه اول**

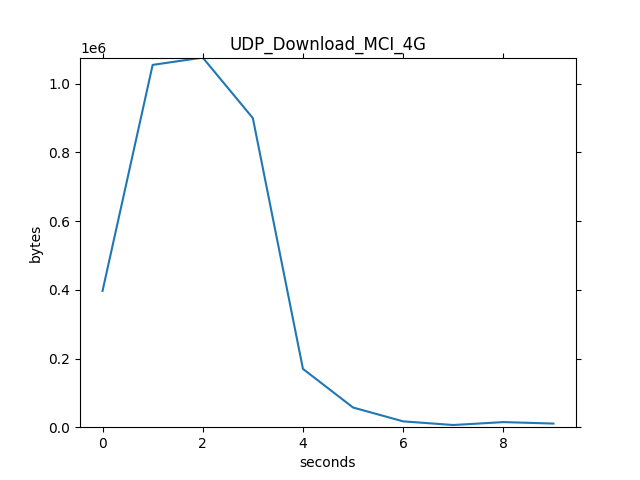
### **۶.۲.۱. اینترنت نسل ۴**

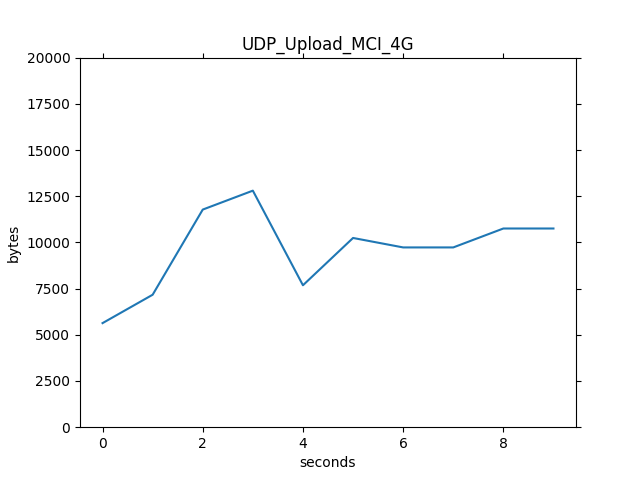
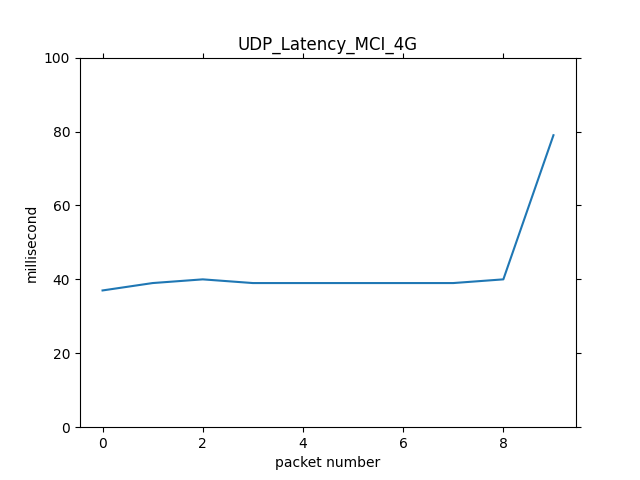
نمودار‌های دانلود، آپلود و تاخیر برای پروتکل TCP را در زیر مشاهده می‌کنید. میانگین دانلود 50 کیلوبایت، آپلود 15 کیلوبایت و تاخیر ۶۰ میلی‌ثانیه است.



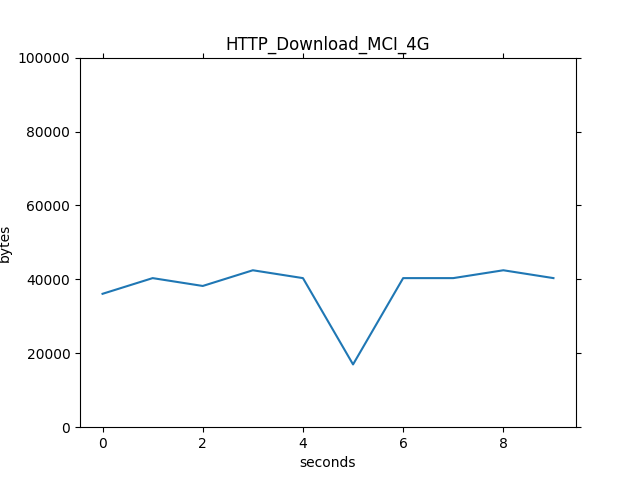


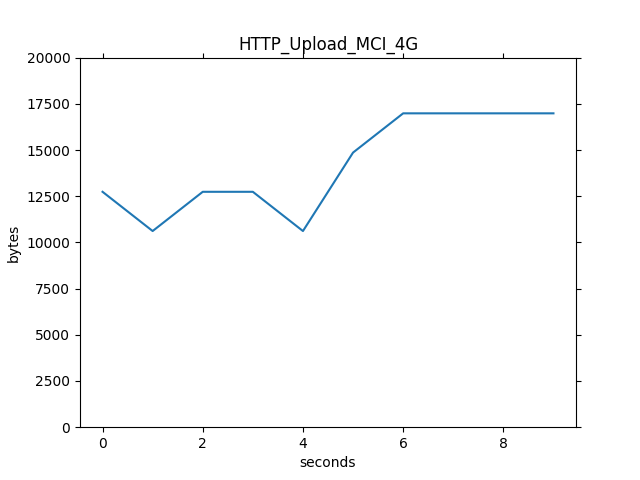
نمودارهای زیر پروتکل UDP را نشان می‌دهند. میانگین دانلود ۶۰۰ کیلوبایت و throughput آن حدود ۱/۲ مگابایت است. میانگین آپلود ۱۰ کیلوبایت و میانگین تاخیر ۴۳ میلی‌ثانیه است.



سه نمودار آخر این بخش نیز متعلق به پروتکل HTTP است. میانگین دانلود ۳۸ کیلوبایت بر ثانیه، آپلود ۱۵ کیلوبایت و تاخیر حدود ۱۰۰ میلی‌ثانیه است.

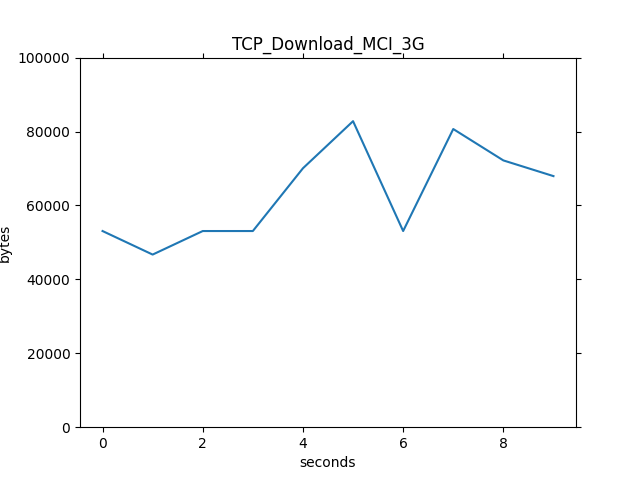


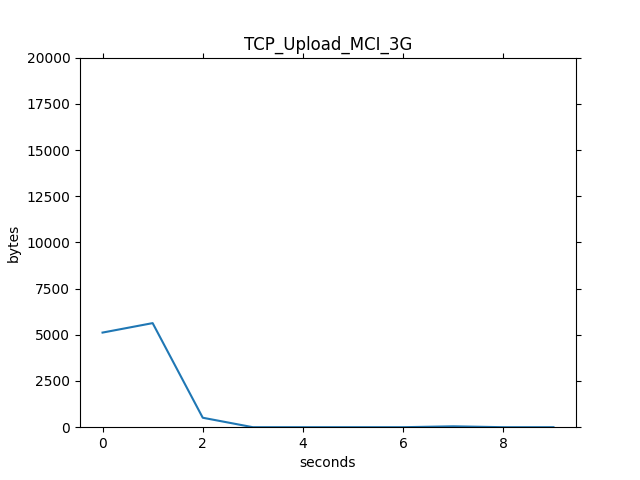


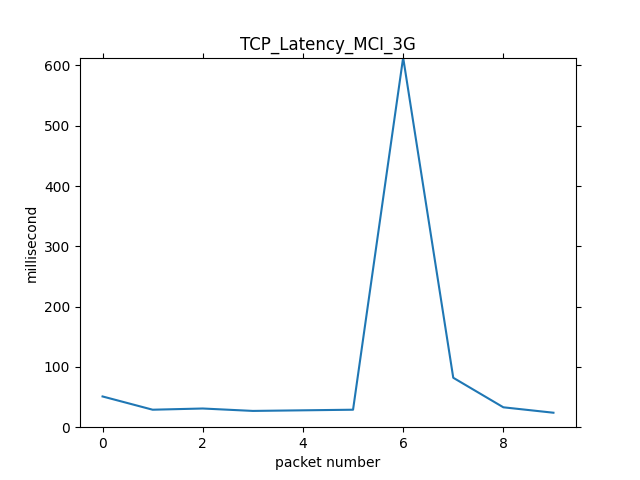


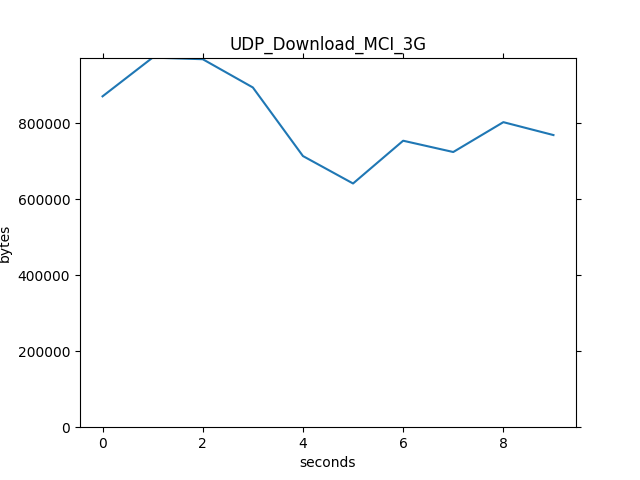
### **۶.۲.۲. اینترنت نسل ۳**

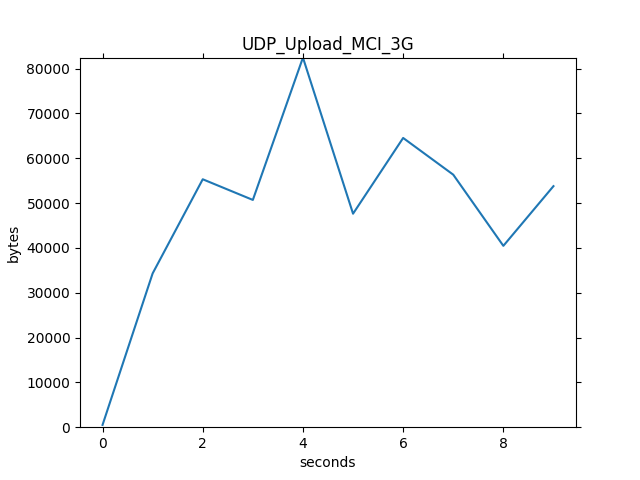
نمودارهای زیر مربوط به پروتکل TCP است. میانگین سرعت دانلود ۶۵ کیلوبایت، آپلود ۱ کیلوبایت و تاخیر،‌با صرف از نظر از افزایش ناگهانی در نقطه ۶ حدود ۴۰ میلی ثانیه است.

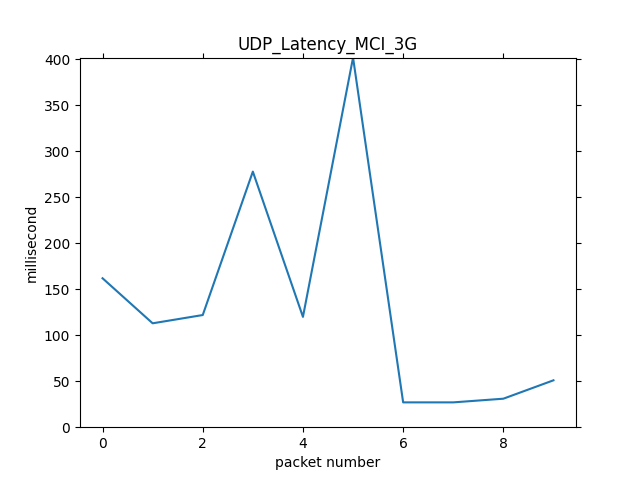




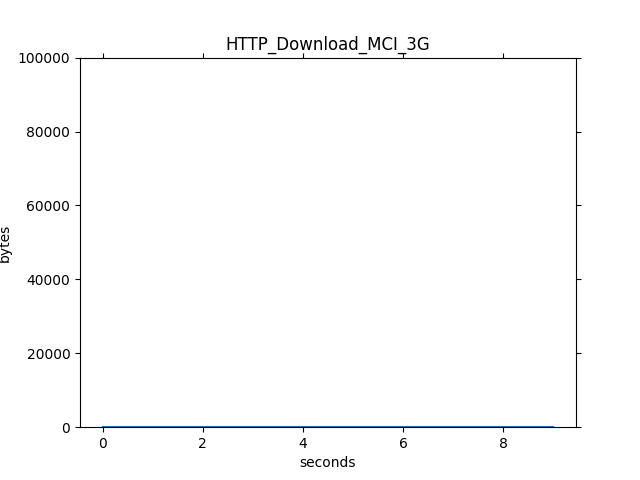
نمودارهای زیر مربوط به روش UDP هستند.  میانگین دانلود ۸۰۰ کیلوبایت، آپلود ۴۰ کیلوبایت و تاخیر ۲۰۰ میلی ثانیه می‌باشد.

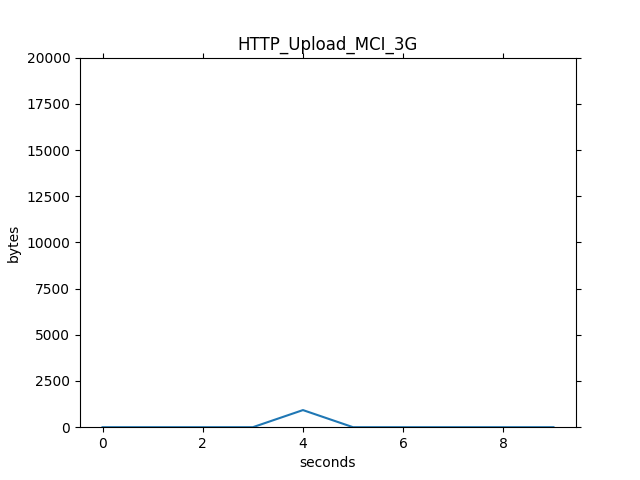


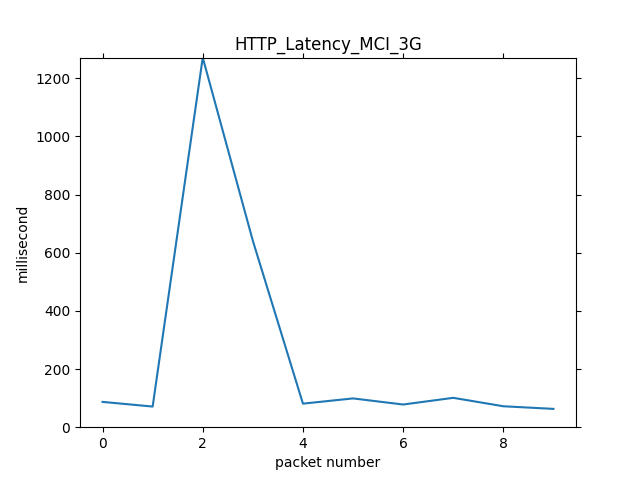




نمودارهای زیر مربوط به پروتکل HTTP هستند. سرعت آپلود و دانلود نزدیک صفر است و بیشترین تاخیر تا به اینجا،‌ یعنی ۱۲۰۰ میلی ثانیه را در نمودار این بخش شاهد هستیم.

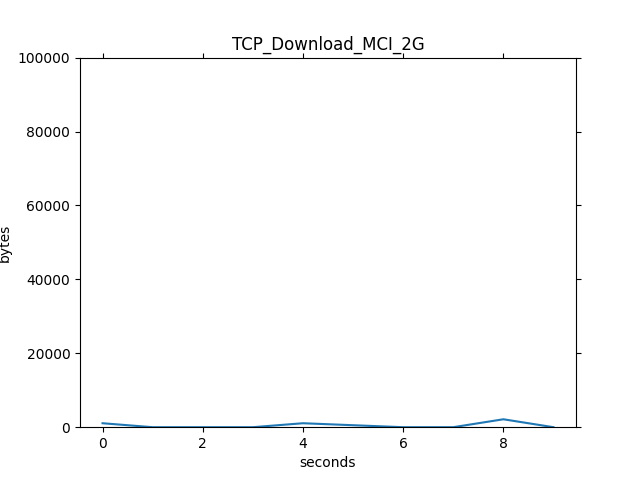


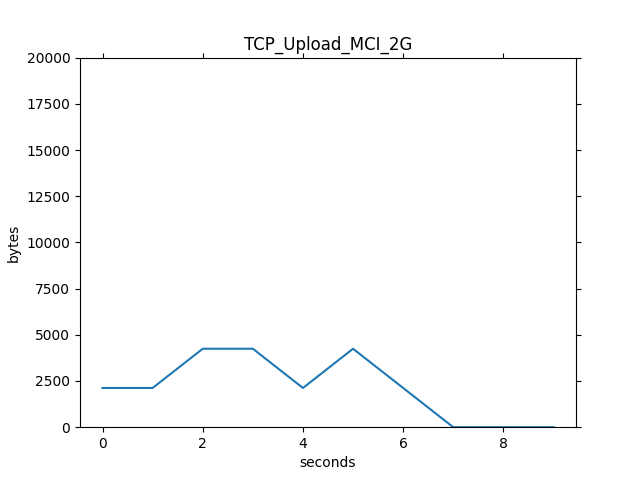


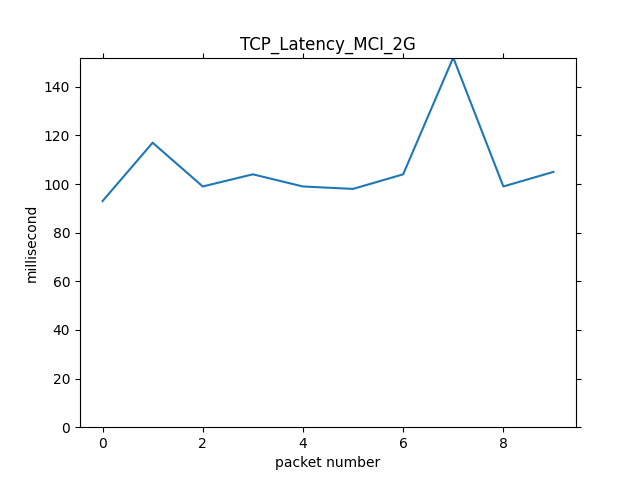


### **۶.۲.۳. اینترنت نسل ۲**

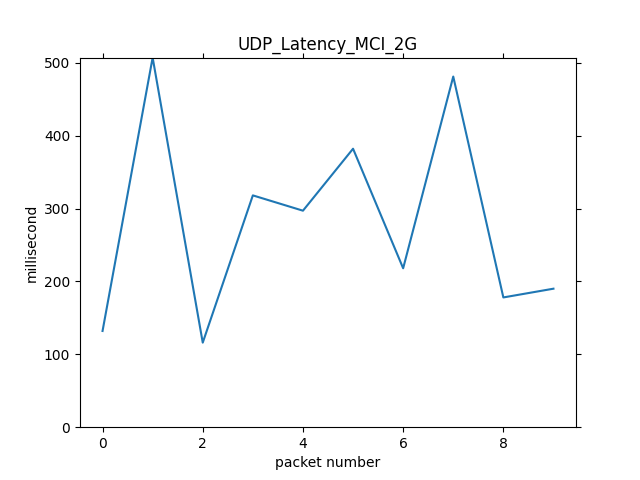
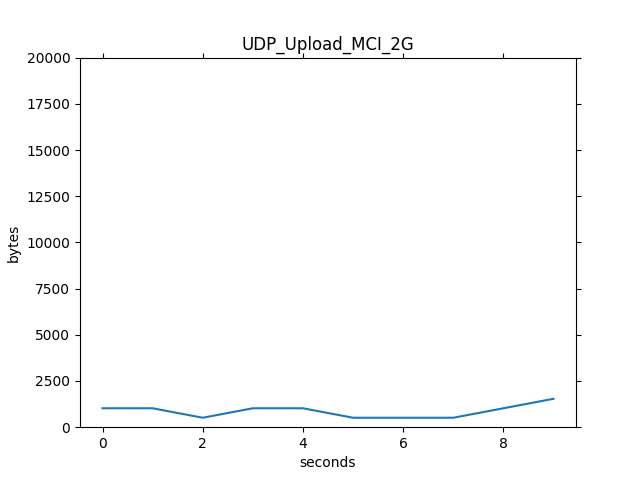
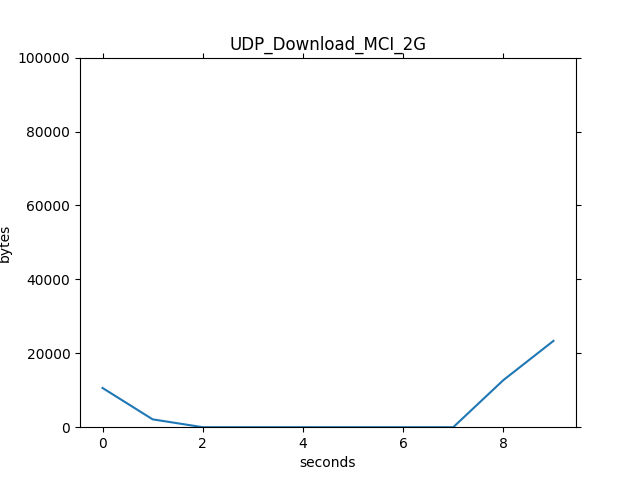
نمودارهای پروتکل TCP را در بخش زیر مشاهده می‌کنیم. میانگین سرعت آپلود و دانلود بسیار کم (حدود ۰ و ۲ کیلوبایت) و تاخیر نیز ۱۲۰ میلی‌ثانیه است.



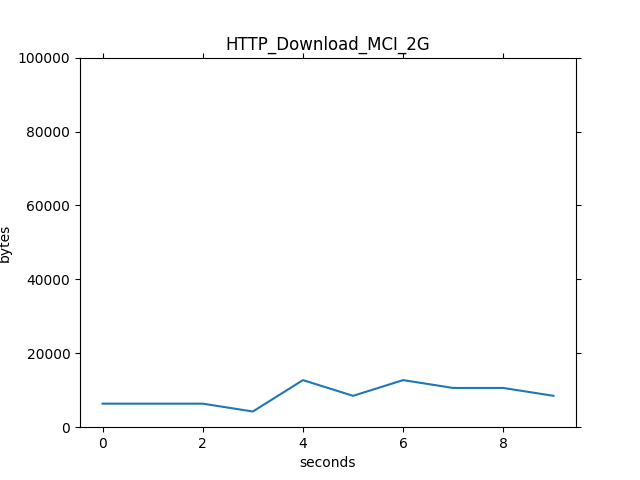


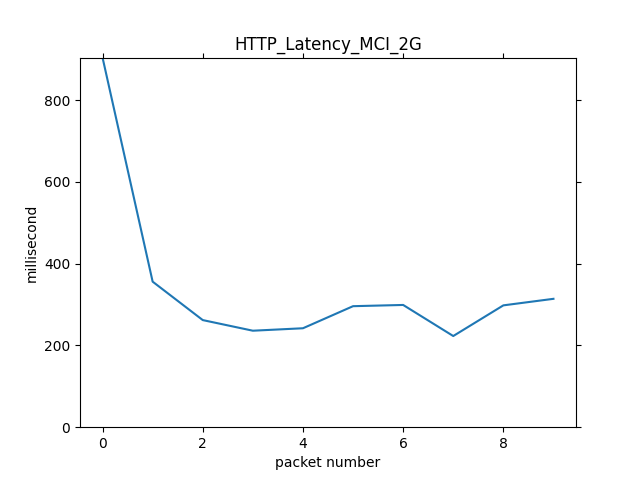
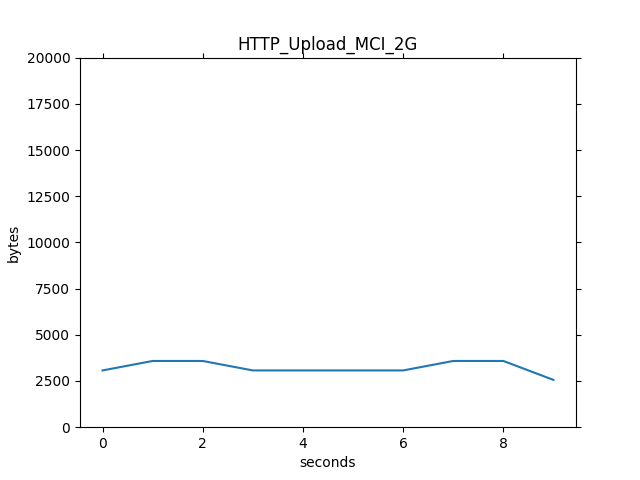


نمودارهای UDP این نسل را در زیر مشاهده می‌کنید. میانگین دانلود حدود ۱۰ کیلوبایت، آپلود ۱ کیلوبایت و تاخیر حدود ۳۵۰ میلی‌ثانیه است.



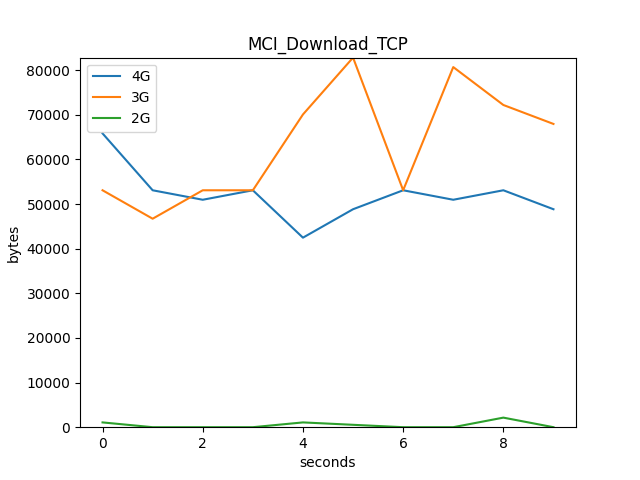
نمودارهای HTTP این نسل را در زیر مشاهده می‌کنید. میانگین دانلود ۵ کیلوبایت، آپلود ۲/۵ کیلوبایت و تاخیر حدود ۵۰۰ میلی ثانیه است.

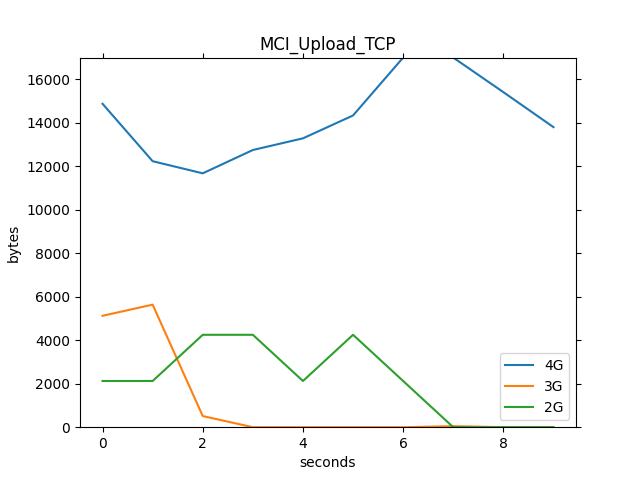


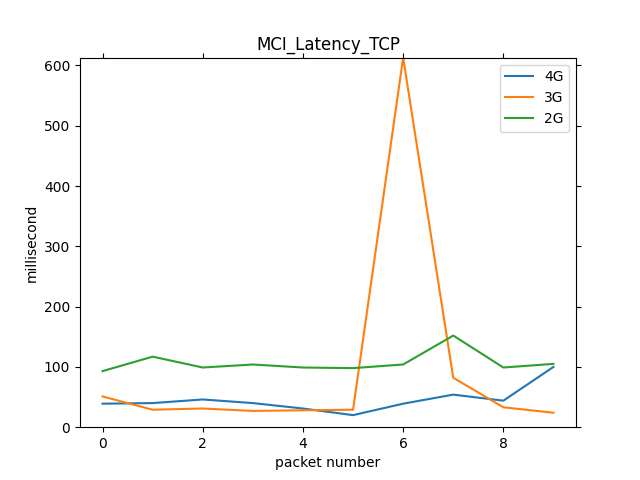


### **۶.۲.۴. مقایسه نسل‌های مختلف**

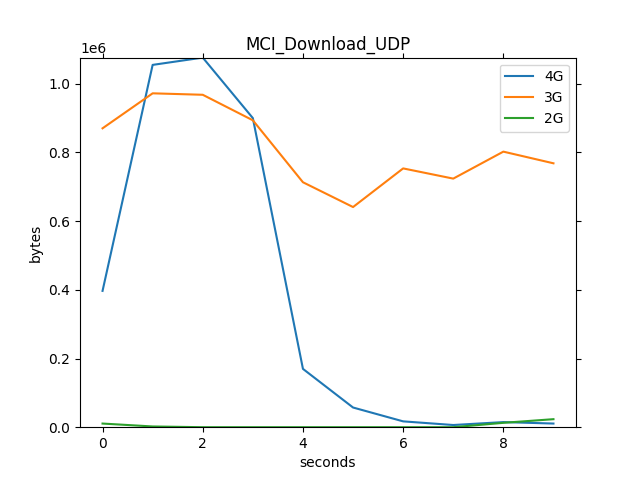
سرعت دانلود TCP را در سه نسل مختلف در نمودار زیر مشاهده می‌کنید.

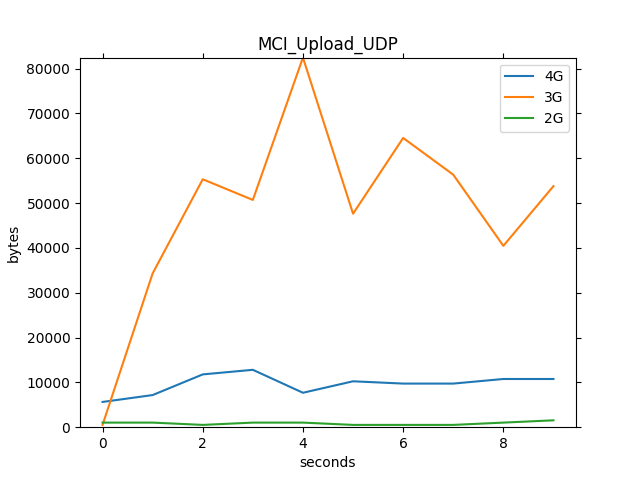
سرعت دانلود در نسل سوم و چهارم به شدت نسبت به نسل دوم افزایش پیدا کرده است.

سرعت آپلود را با استفاده از روش TCP در نمودار بالا مشاهده می‌کنید. نسل چهارم، افزایش چشمگیری در سرعت آپلود داشته است اما نسل دوم و سوم در حدود هم هستند.

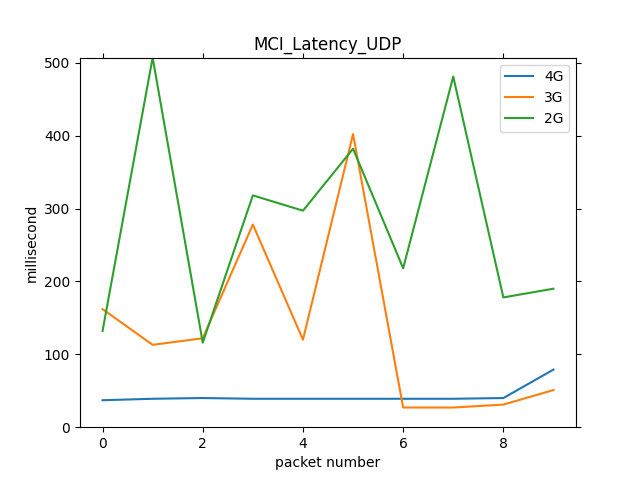


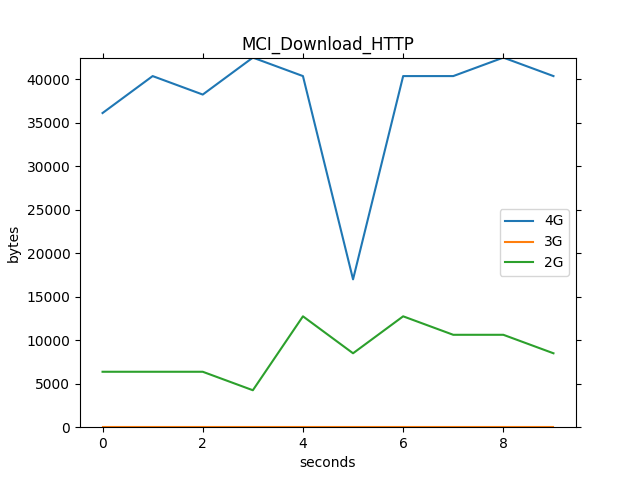
تاخیر سه نسل مختلف را با استفاده از روش TCP در نمودار بالا مشاهده می‌کنید. کمترین تاخیر مربوط به نسل چهارم است.

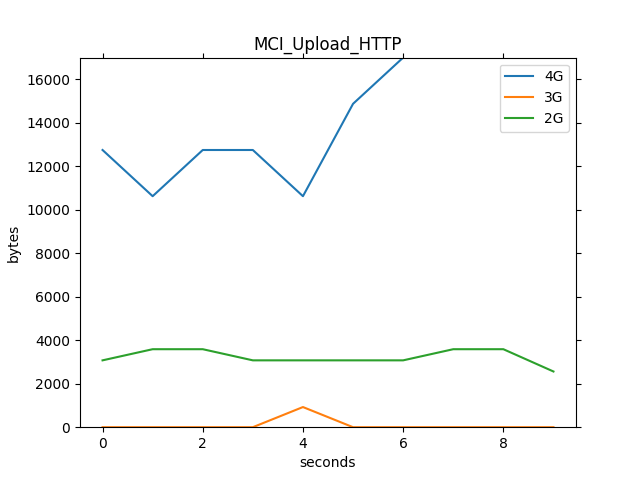
مقایسه سرعت دانلود در سه نسل مختلف با استفاده از پروتکل UDP در نمودار بالا انجام شده است. مشابه TCP، افزایش سرعت دانلود نسل سوم و چهارم نسبت به نسل دوم مشهود است.

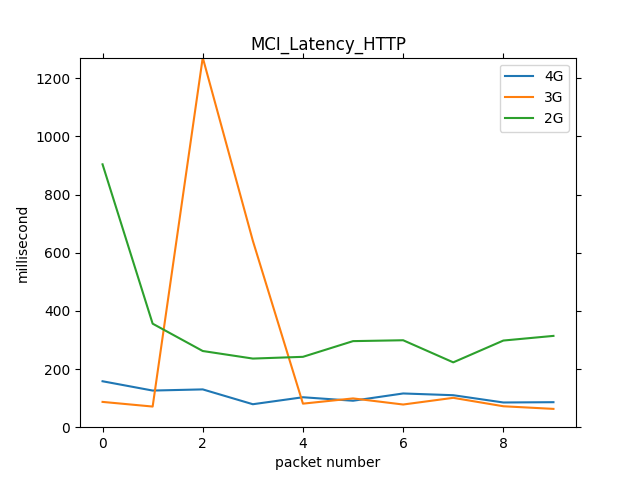


سرعت آپلود پروتکل UDP در نسل سوم در بیشترین مقدار خود است و در نسل چهارم کاهش یافته است.

مقایسه تاخیر را در تصویر بالا برای پروتکل UDP مشاهده می‌کنید. مشابه پروتکل قبل، کمترین تاخیر مربوط به نسل چهارم است.

سرعت دانلود HTTP در نسل چهارم، با اختلاف از دو نسل قبلی بهتر است.

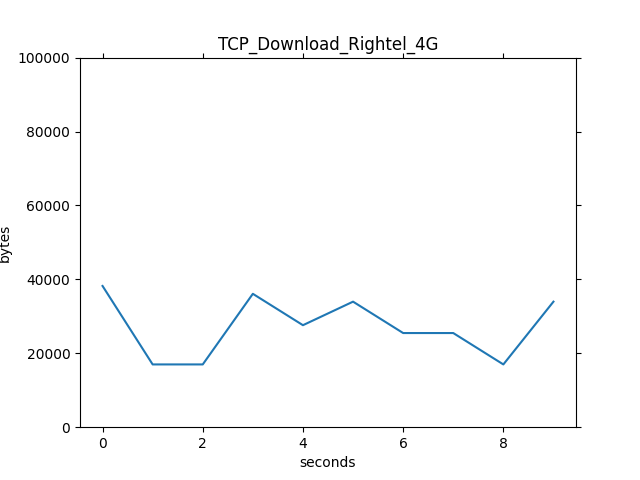
سرعت آپلود HTTP نیز در نسل چهارم پیشرفت زیادی داشته است.

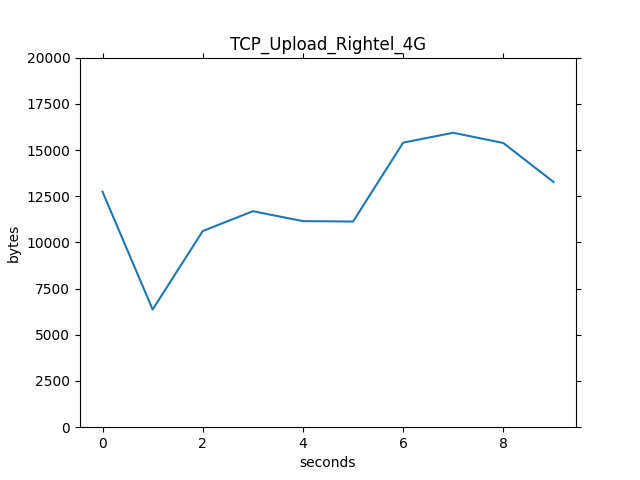
مشابه دو پروتکل قبل، برای HTTP نیز کمترین تاخیر در نسل چهارم ثبت شده است.

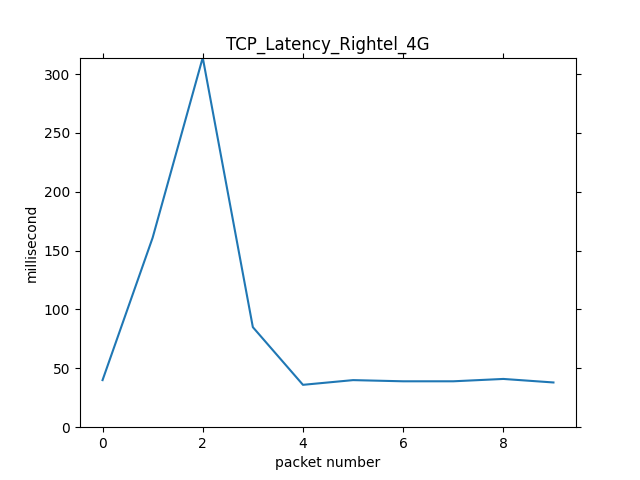
## **۶.۳. ارائه دهنده رایتل**

### **۶.۳.۱. اینترنت نسل ۴**

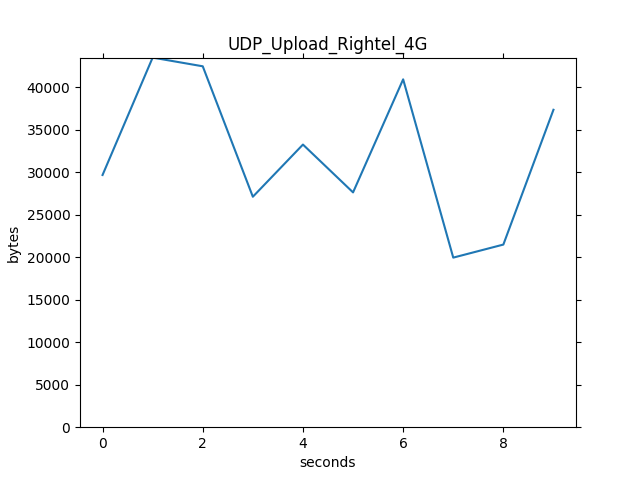
نمودارهای پروتکل TCP در زیر آمده است. میانگین دانلود ۳۰ کیلوبایت، آپلود ۱۲ کیلوبایت و تاخیر ۱۵۰ میلی‌ثانیه است.



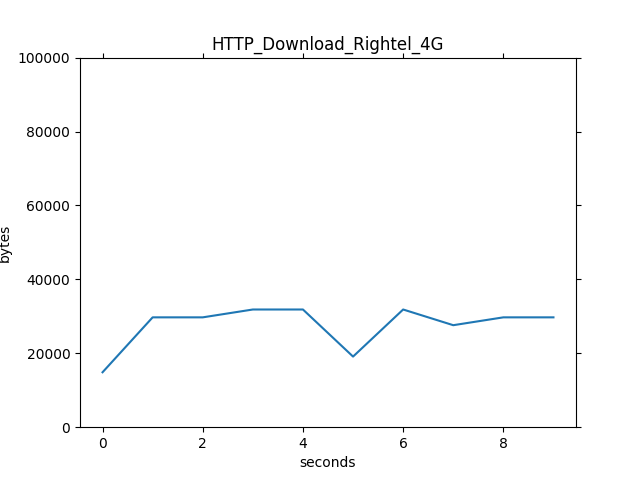


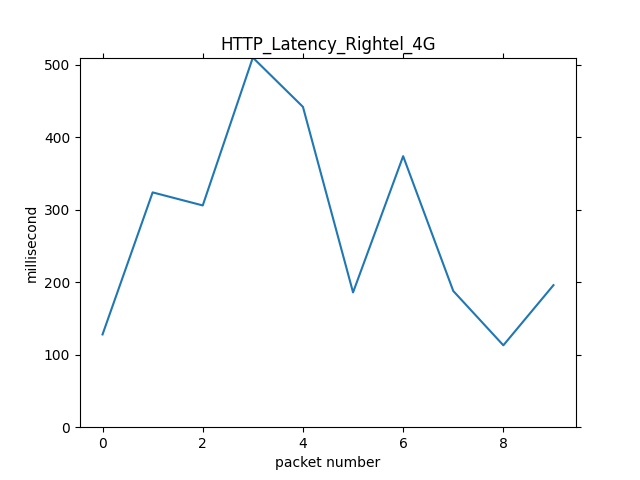
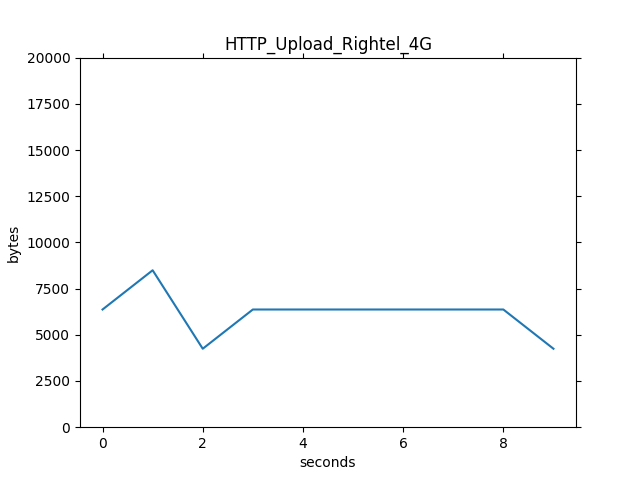


نمودارهای پروتکل UDP در زیر آمده است. این ارایه‌دهنده از دانلود با استفاده از پروتکل UDP پشتیبانی نمی‌کند بنابراین تنها اطلاعات آپلود در دسترس است. میانگین آپلود با استفاده از این پروتکل ۳۰ کیلوبایت است.



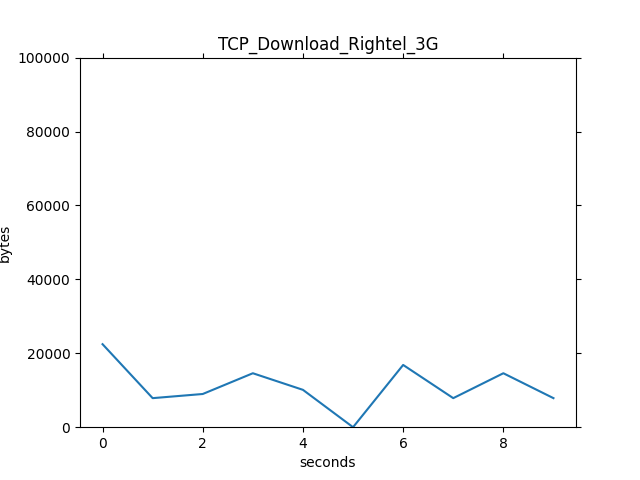
نمودارهای زیر مربوط به روش HTTP هستند. میانگین دانلود ۲۵ کیلوبایت، آپلود ۶ کیلوبایت و تاخیر ۳۰۰ میلی ثانیه‌ می‌باشد.



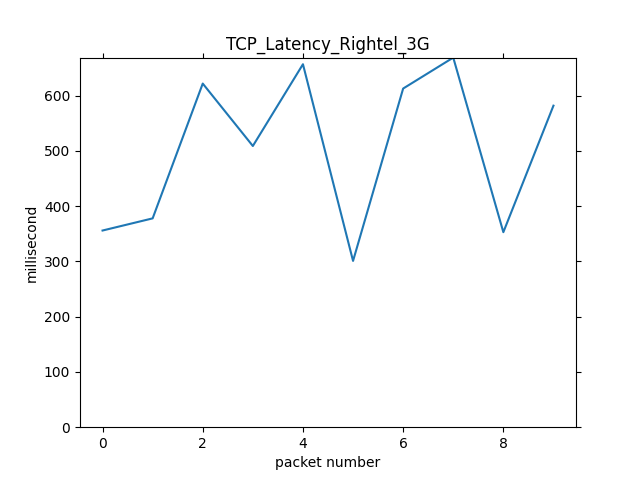


### **۶.۳.۲. اینترنت نسل ۳**

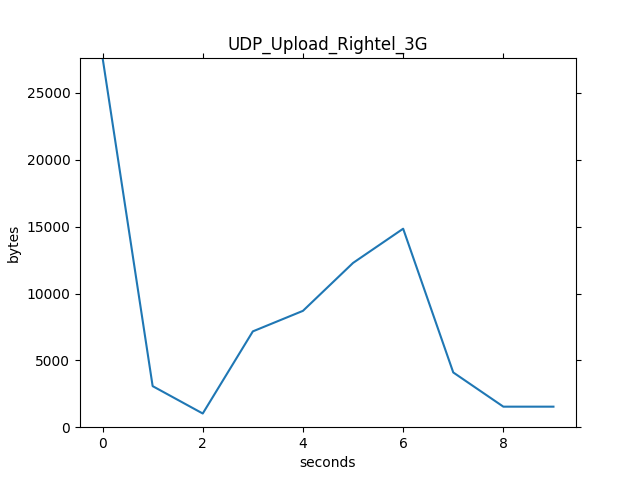
نمودارهای زیر برای پروتکل TCP هستند. میانگین دانلود ۱۰ کیلوبایت، آپلود ۴ کیلوبایت و تاخیر ۵۰۰ میلی ثانیه می‌باشد.



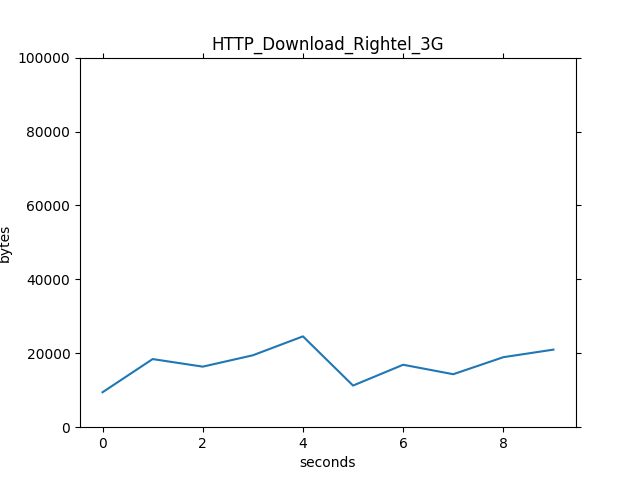


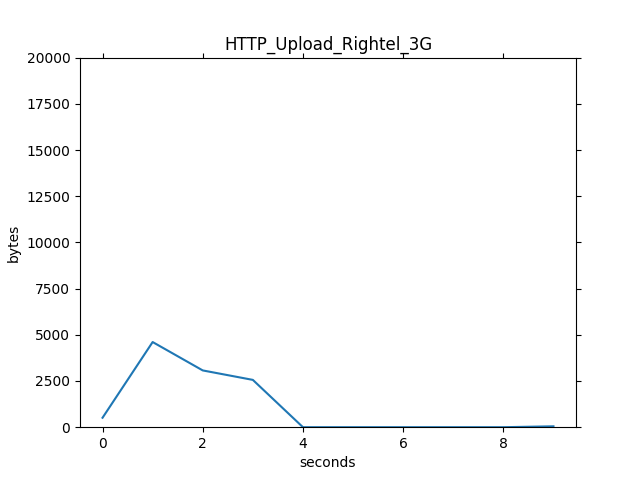


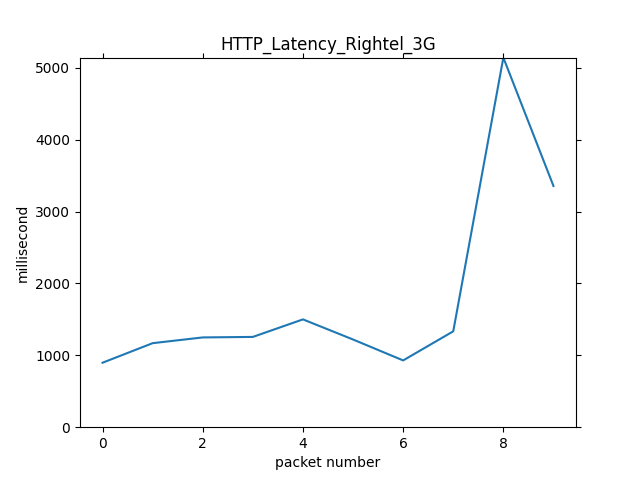
نمودار زیر، آپلود UDP است. میانگین سرعت آپلود با این روش ۷ کیلوبایت است.



نمودارهای زیر مربوط به HTTP است. میانگین دانلود ۱۵ کیلوبایت، آپلود ۲ کیلوبایت و تاخیر بیشتر از تمامی حالتهای دیگر ۲۰۰۰ میلی ثانیه است.







1. Packet loss [↑](#footnote-ref-1)
2. Congestion [↑](#footnote-ref-2)
3. Drop rate [↑](#footnote-ref-3)
4. latency [↑](#footnote-ref-4)