JCWR2020

کنفرانس بین المللی وب پژوهی
ایران، تهران، ۲۲ خرداد ۱۳۹۹

Side-Effects of Reliable IoT Protocols by Hardware Implementation

Mahdie HajilooVakil[1], Zahra Shirmohammadi[2]

[1] Student of Computer Engineering, Shahid Rajaee Teacher Training University, Tehran, Iran
hajiloo@sru.ac.ir

[2] Assistant Professor of Computer Engineering, Shahid Rajaee Teacher Training University, Tehran, Iran
shirmohammadi@sru.ac.ir

Abstract

One of the emerging technologies that have entered the daily life of human beings with the development of wireless networks is the Internet of Things (IoTs). IoTs is the invention of new intelligent devices, which is rapidly developing. In this paper, we implement the application layer protocols and two of the most trusted IoT protocols, MQTT and CoAP in terms of power consumption and implement them on Nodemcu hardware. The results in the graphs show that the average power consumption of the MQTT protocol is higher than the CoAP protocol, which is due to the high reliability of the MQTT protocol, which imposes additional overhead on the network, resulting in additional overhead. It is a control flow mechanism that is used in the TCP sub-layer protocol as the MQTT network transmission layer protocol.

Keywords: Internet of Things, Protocol, Power Consumption, Reliability, Application Layer, MQTT, CoAP,

JCWR2020
کنفرانس بین المللی وب پژوهی
ایران، تهران، ۲۶ خرداد ۱۳۹۹

پیاده سازی و مقایسه سخت افزاری پروتکل‌های قابل اطمینان اینترنت اشیا

مهدیه حاجیلووکیل[۱]، زهرا شیرمحمدی[۲]

[۱] دانشجو، دانشکده مهندسی کامپیوتر دانشگاه شهید رجایی، تهران
hajiloo@sru.ac.ir

[۲] استادیار، گروه معماری سیستم‌های کامپیوتر، دانشکده مهندسی کامپیوتر دانشگاه شهید رجایی، تهران
Shirmohammadi@sru.ac.ir

چکیده

یکی از فن‌آوری‌های نو ظهور که با توسعه شبکه‌های بی سیم و ابداع وسایل هوشمند جدید به زندگی روزانه انسان‌ها ورود پیدا نموده، اینترنت اشیاء می‌باشد، که روند توسعه آن با سرعت زیادی در حال انجام است. ما در این مقاله به بررسی پروتکل‌های لایه‌ی کاربرد ومقایسه‌ی دو پروتکل از پروتکل‌های قابل اطمینان در اینترنت اشیا یعنی MQTT وCoAP از نظر توان مصرفی پرداختیم. این مقایسه بر روی سخت افزار Nodemcu پیاده سازی شده است. نتایج حاصل در نمودارها نشان می‌دهد که توان مصرفی پروتکلMQTT به طور میانگین از پروتکلCoAP بیشتر است که این مساله درMQTT به دلیل ماهیت قابلیت اطمینان بالا در این پروتکل است که سربار اضافی را به شبکه تحمیل می‌کند و این سربار اضافی ناشی از مکانیزم جریان کنترل است که در پروتکل زیر لایه‌یTCP به عنوان پروتکل لایه‌ی انتقال شبکه‌ی پروتکلMQTT به کار گرفته شده است.

کلمات کلیدی

اینترنت اشیاء، پروتکل،توان مصرفی، قابلیت اطمینان ،لایه کاربرد، MQTT،CoAP ،Nodemcu.

۱-مقدمه

تعداد کل دستگاه‌های متصل به هم اینترنت اشیا در سراسر دنیا در حال افزایش هستند و در سال ۲۰۲۵ به ۷۵ بیلیون دستگاه خواهند رسید. این رشد ارتباطات سازمان‌های استاندارد بین المللی را بر آن داشت که پروتکل‌های جدیدی را برای رعایت الزامات و قابلیت اطمینان استاندارد گذاری کنند، زیرا انتقال داده در محیط‌های خشن از طریق لینک‌های پویا غیر قابل اعتماد هستند و باعث ایجاد مصرف انرژی بالا و تاخیر طولانی می‌شوند، درواقع امروزه بحث قابلیت اطمینان از اهمیت بالایی برخوردار است [2].

اینترنت اشیا زمینه‌های متفاوت مانند نظارت بر مراقبت‌های بهداشتی، خانه‌های هوشمند و ساختمان‌ها را پوشش می‌دهد. استفاده برای ارتباطات راه دور، ارتباط با افراد سالخورده که امکان رفتن به پزشک را ندارند، برای کسانی که مشکلات مالی دارند و افراد دور از دسترس از طریق مانیتورینگ، شهرهای هوشمند، خانه‌های هوشمند، نیازمندی‌ها به زمان واقعی، ارتباطات با انرژی کم که به کمک آن می‌توان کاربردهایی مانند مدیریت ترافیک، پیش بینی وضع هوا، زیر ساخت‌های تسویه آب، تولید برق و... را داشت [9] .

از بین چالش‌های اینترنت اشیا همواره دو مساله مهم مورد بحث قرار می‌گیرد:

-چالش‌های قابلیت اطمینان

-چالش‌های توان مصرفی

قابلیت اطمینان به‌عنوان توانایی تجهیز برای انجام در کار موردنیاز، بدون شکست، برای یک دوره زمانی مشخص‌شد، تحت شرایط داده‌شده تعریف می‌شود.

با توجه به اهمیت اندازه گیری انرژی و پایداری انرژی در اینترنت اشیا در هر مکان و موقعیت، نظارت و کنترل منابع، از راه دور اهمیت پیدا می‌کند. مدیریت انرژی (ادغام شدن سیستم‌های حسگر و فعال کننده متصل به اینترنت پتانسیل بهینه سازی مصرف انرژی را ایجاد کرده‌است در نتیجه با توجه به اینکه بسیاری از دستگاه‌ها یا اینترنت اشیا با باتری عمل می‌کنند یا به انرژی مصرفی با منابع محدود محدودیت دارند مصرف توان فوق العاده کم و مدیریت انرژی کارآمد اهداف اصلی در هر دو طراحی سخت‌افزاری و نرم‌افزاری کاربردی است [10,9].

برای حل این مشکلات از معماری اینترنت اشیا استفاده می‌شود. معماری اینترنت اشیاء مدل چهارگانه است که به عنوان مبنا از آن استفاده نموده اند، در این مدل چهار لایه وجود دارد که به ترتیب لایه حسی، لایه شبکه، لایه میان پوششی ولایه کاربردی است که هر لایه پروتکل مختص به خود را دارد. در این مقاله تمرکز خود را بر روی بررسی پرکاربردترین پروتکل‌های لایه کاربرد اینترنت اشیاء گذاشته‌ایم و در انتها دو پروتکل CoAPوMQTT که از پروتکل‌هایی با قابلیت اطمینان بالا در اینترنت اشیا هستند را برروی سخت افزارNodemcu پیاده سازی و از نظر توان مصرفی بایکدیگر مقایسه می‌کنیم.

با توجه به اینکه قابلیت اطمینان در ارسال ودریافت پیام گره‌های اینترنت اشیا بسیار مهم است، دو پروتکلMQTT وCoAP موجود در لایه کاربردقابلیت اطمینان لازم را با توجه به ساختارشان در تحویل بسته‌های اطلاعاتی بین گره‌های اینترنت اشیا فراهم می‌کنند.

توان مصرفی نیز در ارسال و دریافت بسته‌های اطلاعاتی جز چالش‌های بسیار مهم اینترنت اشیا می‌باشد که در پروتکل‌های MQTTوCoAP این ویژگی بررسی و اندازه گرفته شده است که پروتکلMQTT با اینکه قابلیت اطمینان بالایی در ارسال و دریافت پیام ایجاد می‌کند اما بدلیل استفاده از پروتکل زیرلایه TCP توان مصرفی بالایی دارد و باتوجه به اینکه توان گره‌ها در اینترنت اشیا محدود هستند باید به این امر مهم توجه شود.

سخت افزار مورد استفاده در این مقاله میکروکنترلر Nodemcu ESP8266است که بدلیل داشتن هسته ی ESP8266 به راحتی بهwifi وصل شده و اطلاعات را ارسال و دریافت می‌کند هم چنین این میکروکنترلر یک پردازنده ۳۲ بیتی RISC با سرعت ۸۰ مگاهرتز ، یک RAM مکمل و پشتیبانی از حداکثر ۱۶ مگابایت حافظه فلش خارجی دارد. این دستگاه به دلیل اندازه کوچکش و پشتیبانی از WiFi برای صنعت اینترنت اشیا بسیار مفید است.

ساختار این مقاله به این گونه است که ابتدا چالش‌های اینترنت اشیا که تا کنون با آن روبرو بوده نام برده شده و توضیح داده می‌شوند و سپس کارهای پیشین در این زمینه که شامل معرفی معماری اینترنت اشیا و پروتکل‌های لایه کاربرد است شرح داده می‌شوند .در انتها پروتکل‌های قابل اطمینان انتخابی از لایه کاربرد که پیشنهاد شده اند بر روی سخت افزارهای معرفی شده پیاده سازی می‌شوند و بایکدیگر از لحاظ توان مصرفی مقایسه می‌شوند.

۲-چالش‌های اینترنت اشیا

با توجه به افزایش روز افزون کاربرد اینترنت اشیا در زمینه‌های مختلف همواره این فناوری با چالش‌های بسیاری روبرو بوده است که برای تحقق اهداف این فناوری نو ظهور باید اقدام لازم را برای غلبه بر این چالش‌های مطرح شده برداشت. در زیر به بررسی این چالش‌ها می‌پردازیم:

- چالش‌های قابلیت اطمینان: همانطور که می‌دانید قابلیت اطمینان،احتمال کارکرد صحیح و بدون اشکال یک سیستم یا آیتم، در شرایط مشخص و از پیش تعیین شده برای یک زمان مشخص را مطرح می‌کند. منابع موثر در به خطر افتادن قابلیت اطمینان می‌تواند شامل تغییرات سطح ولتاژ، نویز، تابش و سالخوردگی باشند[11,12].

- چالش‌های توان مصرفی: به دلیل تقاضای تامین انرژی از میلیاردها سنسور، دروازه‌های ورودی و سرورهای ذخیره وپردازش اطلاعات، نیازمند مصرف انرژی الکتریکی فراوانی می‌باشند و افزایش تقاضا برای مصرف و تولید الکتریسیته بیشتر، خود می‌تواند آسیب‌های زیست محیطی بیشتری همانند افزایش گرمای جهانی را تشدید نماید.مواردی که در اینترنت اشیا لزوم مدیریت توان مصرفی را قابل اهمیت می‌کند شامل موارد زیر است [10,12]:

- افزایش داده‌های انتقالی و تحلیل آن‌ها
- افزایش سرعت درخواست و پاسخ و کاهش تاخیر در زمان
- اتمام منابع انرژی مورد استفاده
- استفاده از پلتفرم‌ها و سخت افزارهای نامناسب

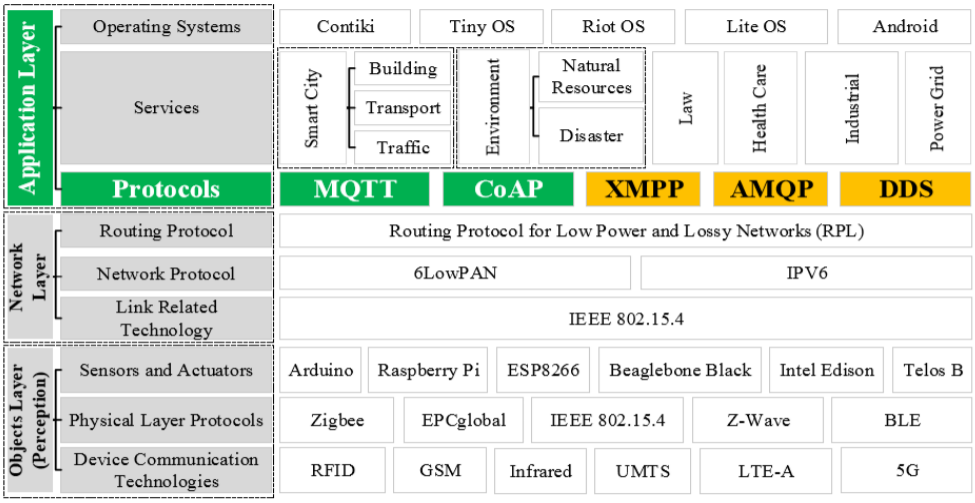
۳-کارهای پیشین

۳-۱معماری اینترنت اشیا

معماری ارائه شده برای اینترنت اشیاء در مطالعه‌ی Engin Leloglu درسال ۲۰۱۷ معرفی شده است، یک مدل چهارگانه است که به عنوان مبنا از آن استفاده نموده‌اند، در این مدل چهار لایه وجود دارد که در شکل (۱) مشاهده می‌کنید:

لایه حسی: این لایه شامل انواع متفاوتی از حس‌گرها می‌باشد که وظیفه تعامل با محیط اطراف به عهده دارند و اطلاعات به‌صورت خام دریافت می‌کنند، میلیاردها حسگر در این لایه نصب خواهند شد، این حسگرها به‌صورت بیسیم با همدیگر در ارتباط هستند[8] .

لایه شبکه: این لایه شامل تمام دستگاه‌ها و ابزارهایی است که وظیفه تامین شبکه‌ی اتصالات حسگرها را به عهده دارند، دروازه‌های ورودی، سوئیچ‌ها و دیوارهای آتش نمونه‌هایی از این ابزارها می‌باشند [8].



شکل(۱):لایه‌های اینترنت اشیا وپروتکل‌های هر لایه[2]

لایه میان پوششی: این لایه شامل دریافت اطلاعات از لایه شبکه و پردازش و ذخیره اطلاعات می‌باشد، سرورها و رایانش ابری نمونه‌ای از ابزارهای این لایه می‌باشند[8] .

لایه کاربرد: این لایه وظیفه ارتباط با کاربر و یا کاربران را که طیف وسیعی را شامل می‌شوند. برنامه‌های متفاوتی که در گوشی‌های هوشمند وکامپیوترها برای کنترل اشیاء همانند سیستم‌های گرمایشی و سرمایشی هوشمند بر اساس سیستم عامل‌های متفاوت مثل اندروید، ویندوز، مک اینتاش تهیه شده است[8]

۲-۳معرفی پروتکل‌های لایه کاربرد

هرلایه در اینترنت اشیا دارای پروتکل‌های مخصوص به خود است که در ادامه به معرفی پروتکل‌های لایه کاربرد و مقایسه‌هایی که انجام شده است می‌پردازیم:

پروتکل XMPP یک پروتکل لایه کاربرد که درسال۱۹۹۹منتشر شد و توسط IETFاستاندارد شده است. این پروتکل برای تبادل پیام با فرمت xml طراحی شده است وپروتکل زیر لایه ی آن TCPاست. در این پروتکل امنیت توسط پروتکل TLS/SSLتامین می‌شود. دو نوع معماری انتشار/اشتراک و درخواست/پاسخ را پشتیبانی می‌کند. در این پروتکل به دلیل استفاده ازXML سرباراضافی را در شبکه ایجاد می‌کند ودر محیط اینترنت اشیا باعث افزایش مصرف توان می‌شود وهم چنین در این پروتکل هیچ گونه گارانتی QOS وجود ندارد در نتیجه قابلیت اطمینان در ارسال بسته‌ها را ایجاد نمی‌کند[4].

پروتکل AMQP در سال۲۰۰۳ ارائه شد و هنوز استاندارد نشده است. مخصوص محیطهای پیام‌گراست پروتکل زیر لایه آنTCP است ودر این پروتکل امنیت توسط پروتکل TLS/SSL تامین می‌شود.معماری این پروتکل ازنوع انتشار/اشتراک است که دارای یک بارگزار است که خود شامل دوقسمت صف و تبادل پیام می‌باشد. از معایب این پروتکل ارتباطی می‌توان به نامناسب بودن برای برنامه‌های کاربردی با زمان واقعی و محیطهای محدود اشاره کرد [3,5].

پروتکل DDS در سال۲۰۰۴ارائه شده است این پروتکل از معماری انتشار/اشتراک بدون کارگزار استفاده می‌کند. طراحی شده برای ارتباطات ماشین با ماشین است. پروتکل زیر لایه آنTCPاست.پیامی که بین منتشر کنندگان ومشترکان منتقل می‌شود فظاTopic dataنامید هم چنین در این پروتکل بالاترین سطحQoS را فراهم می‌شود. در این پروتکل ساختار پیچیده است که باعث می‌شود انرژی وتوان زیادی را مصرف کند و برای محیطهای محدود اینترنت اشیا مناسب نیست[5,2] .

پروتکلMQTTدر سال۱۹۹۹ارائه و درسال۲۰۱۳توسطOASIS استاندارد شده است. هدف از طراحی این پروتکل برای ارائه اتصالات بین برنامه‌های کاربردی در یک طرف و شبکه‌ها و ارتباطات در طرف دیگر می‌باشد. پروتکل زیر لایه آن TCPاست و امنیت آن توسط TLS/SSLتامین می‌شود. دارای معماری انتشار/اشتراک است و از یک بارگزار هم استفاده می‌شود که منتشر کنندگان گره‌های اینترنت اشیاهستندکه اطلاعات جمع آوری شده را به بارگزارها می‌دهند و آن‌ها نیز پیام‌ها را به اشتراک مشترکان می‌گذارند[3,5,6] .

از مزایای استفاده این پروتکل در اینترنت اشیا می‌توان به انعطاف پذیری در انتقال وسادگی اجرا، مناسب برای دستگاه‌ها با منابع محدود، پروتکل مناسب پیام رسانی برای ارتباطات اینترنت اشیاء، ارائه مسیر یابی برای دستگاه‌های کوچک و ارزان، سربار کم، نیاز به پهنای باند و پردازش کم، طول عمر باطری بیشتر، اطمینان از تحویل بسته، تاخیرکم، به حداقل رساندن ترافیک شبکه، متن بازبودن، پشتیبانی از انواع پلتفرم‌ها ومعماری ساده این پروتکل اشاره کرد. در مقابل این مزایا این پروتکل تنها برای محیطهای محدود با توان کم قابل استفاده است وقابلیت محاسبه، حافظه و پهنای باند محدود را دارد. هم چنین در این پروتکل ۳ سطح کیفیت خدمات رسانی شامل Qos(0)ارسال بیشتر از یک بار پیام،Qos(1)ارسال حداقل یک بار پیام،Qos(2) ارسال دقیقا یک بار پیام وجود دارد که قابلیت اطمینان را در لایه کاربرد ضمانت می‌کند[3,5,6] .

پروتکلCoAP در سال۲۰۱۰ توسطIETF ابداع شد. برای انتقال وب با گره‌ها و شبکه‌های محدود شده در اینترنت اشیاتوسعه‌یافته‌است. جایگزین مناسب برای HTTPاست. پروتکلHTTP به دلیل استفاده از توابع بسیار زیاد در ارسال و دریافت بسته‌های اطلاعاتی بسیار سنگین است و برای محیط اینترنت اشیا مفید نیست در نتیجه پروتکلCoAP جایگزین مناسبی در محیط اینترنت اشیا برای این پروتکل است. پروتکل زیر لایه‌ی آنUDP است.پروتکل امنیتی آن DTLS است.دارای معماری Resource/Observeکه یک نمونه از معماری انتشار/اشتراک است و همین طور معماری سرویس دهنده/سرویس گیرنده را نیز پشتیبانی می‌کند. مقید به محیط Restfulهست. دارای دو لایه به نام‌های:۱-Messaging ۲-Request/Response که لایه اول مسئول قابلیت اطمینان و ارسال مجدد پیام را داردو لایه دوم مسئول ارتباطات است[3,5,6] .

از مزایای این پروتکل می‌توان به این موارد اشاره کرد: مناسب برای برنامه‌های‌اینترنت اشیا، ارتباط ماشین با ماشین در محیط‌های محدود وپشتیبانی از محیط‌های چند بخشی و تک بخشی پیام، سرآیند با اندازه ۴بایت، قابلیت اطمینان با پیام CON، معماری ساده، مصرف کم پردازنده و حافظه، انعطاف پذیری بالا[3,5,6] .

باتوجه به اینکه همه‌ی پروتکل‌های لایه‌ی کاربرد باید قابلیت اطمینان، مصرف انرژی کم و تاخیر پایین در ارسال و دریافت پیام را داشته باشند طبق معرفی‌های شده از پروتکل‌های لایه‌ی کاربرد به این نتیجه می‌رسیم که پروتکل AMQP بدلیل داشتن ۸بایت سرآیند و سربار بالایی در شبکه ایجاد و انرژی زیادی نیز مصرف می‌کند که برای محیط‌های محدود مانند اینترنت اشیا مناسب نیست. پروتکلXMPP هیچ گونه کیفیت خدمات رسانی را پشتیبانی نمی‌کند در نتیجه هیچ ضمانتی ایجاد قابلیت اطمینان ندارد و به دلیل استفاده از یک نوع فرمت خاص در ارسال پیام سربار زیادی در شبکه ایجاد می‌کند و ترافیک شبکه را بالا برده باعث افزایش مصرف توان و انرژی می‌شود. پروتکل DDS نیز به دلیل ساختار پیچیده مناسب محیط‌های محدود اینترنت اشیا نیست[2]. بنابراین ازبین پروتکل‌های معرفی شده ی لایه کاربرد تنها پروتکل‌های MQTTوCoAP برای تحقق ویژگی‌های قابلیت اطمینان و توان مصرفی کم پیشنهاد می‌شوند.

پیاده سازی پروتکل MQTT بر روی سخت افزار Nodemcu و اتصال به کارگزارhivemqکه در مطالعه ای درسال۲۰۱۸ انجام شده و هدف از انجام این پیاده سازی ایجاد یک محیط هوشمند وتبادل پیام بین اشیا تحت پروتکلMQTT است[7].

پیاده سازی پروتکل‌هایMQTT وCoAP بر روی ۲عدد سخت افزار ArduinoUNO و استفاده از ماژولESP8266 برای اتصال به wifi و ارتباطات بیسیم بین گره‌ها که در سال ۲۰۱۷ انجام شده است که هدف از انجام این کار اندازه گیری توان مصرفی هر پروتکل، تاخیر در دریافت و ارسال پیام بین گره‌ها، قابلیت پیش گویی نسبت به تغییرات محیطی و تاثیر گذاری پروتکل که منظور از آن ارتباط بین میزان تاخیر با مصرف توان پروتکل است که هرچه سریع تر واکنش نشان دهد و تاخیر کمتری داشته باشد توان بیشتری نیز مصرف می‌کند. نتایج این تحقیق نشان می‌دهد که توان مصرفی پروتکل MQTT از پروتکلCoAP بیشتر است هر چند که این پروتکل قابلیت اطمینان بیشتری در ارسال و دریافت پیام برای محیط اینترنت اشیا ایجاد می‌کند[2].

در این مقاله ما دو پروتکلMQTTوCoAP را بر روی سخت افزار Nodemcu پیاده سازی کرده و جریان و ولتاژ مصرفی مدار را به کمک سنسور جریان اندازه گیری و ویژگی توان مصرفی هر دو پروتکل به هنگام ارسال و دریافت پیام را اندازه می‌گیریم.

٤-روش پیاده سازی

٤-١-معرفی سخت افزارهای مورداستفاده

NodeMCUیک پلت فرم متن باز در زمینه‌ی اینترنت اشیا می‌باشد. هسته‌ی NodeMCU ، چیپ ESP8266 ورژن ESP12می‌باشد که از wifi پشتیبانی می‌کند و به وسیله‌ی آن می‌توان به شبکه‌ی wifiمتصل شد و اطلاعات را میان اینترنت یا سایر دستگاه‌ها جابه‌جا نمود. زبان برنامه‌نویسی این ماژول ، Luaمی‌باشد. از امکانات بسیار عالی این ماژول این است که می‌توان با استفاده از IDE آردوینو و با استفاده از دستوارات آردوینو بر روی آن به سادگی برنامه نویسی کرد. برای پیاده سازی پروتکل‌های CoAPوMQTTاز این IDE استفاده شده است. تصویر این سخت افزار را در شکل (۲) مشاهده می‌کنید.

در تحقیق‌های پیشین برای پیاده سازی این پروتکل‌ها از سخت افزار Arduino به عنوان گره ارتباطی نیز استفاده شده است اما این سخت افزار به طور مستقیم نمی‌تواند به اینترنت وصل شود و باید در کنار آن به صورت جداگانه سخت افزار دیگری به نام ماژولESP8266 نیز نصب کرد و هردوی آن‌ها را برنامه ریزی کرده تا به اینترنت وصل شود، در حالی که این قابلیت به طور مستقیم در Nodemcu فراهم است. همین طور ولتاژ قابل استفاده در Nodemcu ، ۳/۳ولت است در صورتی که بیشترArduinoها با ولتاژ ۵ولت کار می‌کنند. تمامی پروتکل‌های ارتباطی ،SPI، UART وI2C که بر رویArduino کار می‌کنند، توسط Nodemcuنیز پشتیبانی می‌شوند. دلیل استفاده از Nodemcuدر پیاده سازی پروتکل‌ها داشتن ماژول ESP8266برای اتصال مستقیم به اینترنت است و باعث راحتی و سادگی در پیاده سازی شبکه ای از گره‌ها در اینترنت اشیا و ارتباطات بی سیم بین آن‌ها می‌شود.

همین طوراز سنسور جریانAdafruit ina219 برای اندازه گیری جریان مدار و در نهایت محاسبه‌ی توان مصرفی پروتکل‌ها استفاده شده است. این سنسور با دقت بالا و خروجی I2C است. تصویر این سخت افزار را در شکل (۳) مشاهده می‌کنید.

SD adapter یا ماژول کارت SD این ماژول با پروتکلSPI به میکرو کنترلر متصل می‌شود و داده‌هایی که از میکروکنترلر می‌گیرد را در یک رم ذخیره می‌کند.

سخت افزار بعدی مورد استفاده برای اندازه گیری توان مصرفی هر پروتکل ArduinoUNO است که این میکروکنترلر بر پایه ATmega328می‌باشد که در اینجا تنها نقش یک منبع تغذیه و تامین کننده ی برق مدار را برعهده دارد که در شکل (۴) نمایش داده شده است.

۲-٤پیاده سازی پروتکل‌ها

در این قسمت با استفاده از سخت افزار Nodemcu پروتکل‌های MQTTوCoAP را پیاده سازی و توان مصرفی هر کدام را به صورت جداگانه حین ارسال و دریافت پیام اندازه گیری می‌کنیم. هم چنین از محیط ArduinoIDE برای برنامه ریزی سخت افزارهای Nodemcu وArduino UNO استفاده شده است.

۲-۲-٤پیاده سازی پروتکلMQTT

پروتکل MQTT از مدل انتشار/اشتراک برای ارتباط بین گره‌ها استفاده می‌کند. در این روش هرگره می‌تواند منتشر کننده‌ی داده برای یک موضوع خاص باشد. این موضوع در پروتکل MQTT تاپیک نامیده می‌شود.برای

شکل(۲):Nodmcu ESP8266

شکل(۳):سنسور جریان و ولتاژAdafruit INA219

شکل(٤):ArduinoUNO

پیاده سازی این پروتکل تنها از یکNodemcu استفاده می شود که می‌تواند نقش یک منتشر کننده و یا یک مشترک را داشته باشد. یکی از سرورهای رایگان MQTT که در دسترس عموم قرار دارد، البته آدرس مربوط به خود سرور، broker.hivemq.comبر روی پورت 1883 است. با اتصال Nodemcuبه این کارگزار، ارسال و دریافت پیام انجام می‌شود.

۲-۲-٤پیاده سازی پروتکل CoAP

با توجه به معماری سرویس دهنده/سرویس گیرنده ی پروتکل COAP برای پیاده سازی این پروتکل ازدو عدد Nodemcu استفاده می‌کنیم که گره اول نقش سرویس دهنده و گره دوم نقش سرویس گیرنده را دارد و ارسال و دریافت پیام بینشان انجام می‌شود.

۳-۲-٤ اندازه گیری توان مصرفی پروتکل‌هایMQTT وCoAP

همان‌طور که می‌دانید توان برابر حاصل ضرب ولتاژ در جریان است بنابراین لازم است ابتدا ولتاژ و جریان مصرفی گره مان را با استفاده از سنسورجریان Adafruit INA219اندازه بگیریم. برای این کار به کمک سخت افزار ArduinUNOمعرفی شده مداری را ساختیم تا بتواند جریان و ولتاژ را برای هر پروتکل اندازه بگیرد و حاصل ضرب آن‌ها را در یک sd card در قالب فایل txt ذخیره کند که این فایل دارای ۴ ستون است که به ترتیب برابر زمان، ولتاژ، جریان و توان می باشد. در آخر نمودار مصرف توان را برحسب زمان را با استفاده از نرم افزار متلب رسم می‌کنیم. باید از اتصال پایه ی دیجیتال۴ ArduinoUNO به پایه یcs ماژولSD اطمینان حاصل کنیم.
برای اتصال و راه اندازی سخت افزارها به یکدیگر در مدار طراحی شده کتاب خانه‌های لازم باید در محیطArduino IDE نصب شوند که کتابخانه‌های زیر هستند:

کتابخانه‌یwire.h برای برقراری ارتباط بین سنسور ولتاژ و جریان، کتابخانه‌یAdafruit_INA219.h برای راه اندازی سنسور ولتاژ و جریان، کتابخانه‌یSPI.hبرای برقراری SPI بین Arduino UNO و ماژول SD و کتابخانه‌یSD.hبرای راه اندازی ماژولSD.

برای راه اندازی سخت افزارها لازم است تنها بهArduino UNO برق داده شود (با اتصال کابلUSB آن به یک کامپیوتر) و هرگز همزمان هردو به کامپیوترمتصل نباشند. آردوینو در این مدار نقش منبع تغذیه را بازی می‌کند و همین طور ما می‌توانستیم از باطری به جای آردوینو استفاده کنیم. مدار طراحی شده را در شکل (۵) مشاهده می‌کنید.

۵-یافته‌ها

با توجه به ماهیت محیط‌های اینترنت اشیا، مصرف توان در گره‌ها نقش بسیار مهمی دارد. بسیاری از فعالیت‌ها در یک سیستم اینترنت اشیا مثل: ماژول‌های فرستنده و گیرنده، فرآیندهای CPU، قابلیت‌های مسیریابی و پروتکل‌های ارتباطی منجر به مصرف انرژی می‌شود. در این قسمت میزان توان مصرفی

در یک ارتباط پروتکلMQTT به عنوان پروتکل کاربردی قابل اعتماد و CoAP به عنوان پروتکل ارتباطی با قابلیت اطمینان کمتر، مقایسه شده است .یافته‌های حاصل در نمودارهای شکل (۶)، (۷) و (۸) نشان می‌دهد که توان مصرفی پروتکلMQTT به طور میانگین از پروتکلCoAP بیشتر است .

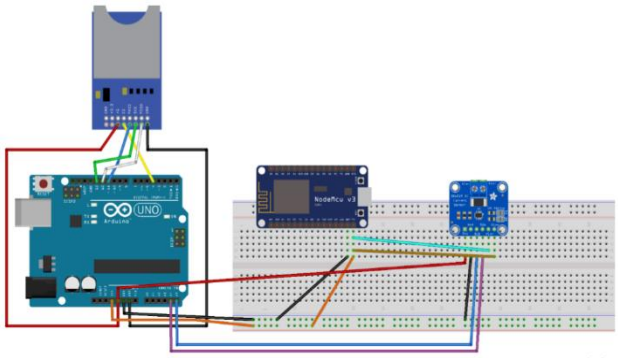
مصرف بالای توان پروتکل MQTT به دلیل ماهیت قابلیت اطمینان در این پروتکل است که سربار اضافی را به شبکه تحمیل می‌کند و این سربار اضافی ناشی از مکانیزم جریان کنترل است که در پروتکل زیر لایه یTCP به عنوان پروتکل لایه‌ی انتقال شبکه‌ی پروتکلMQTT به کار گرفته شده است. پروتکل زیر لایه‌ی TCPقبل از برقراری هر گونه ارتباطی ابتدا باید سه مرحله که اصطلاحا به Three Way Handshake معروف است را طی کند و سه بسته باید ارسال و دریافت شود تا برقراری ارتباط صورت بگیرد. این مکانیزم پروتکلTCPباعث افزایش مصرف توان و انرژی در پروتکل MQTT می‌شود در صورتی که در پروتکلCoAP از پروتکل UDP استفاده شده است که این قابلیت را ندارد. پس نتیجه می‌گیریم که باتوجه به اینکه قابلیت اطمینان در پروتکل MQTT بالاتر است اما این قابلیت اطمینان باعث مصرف بیشتر توان در این پروتکل می‌شود.

٦-نتیجه‌گیری وپیشنهاد

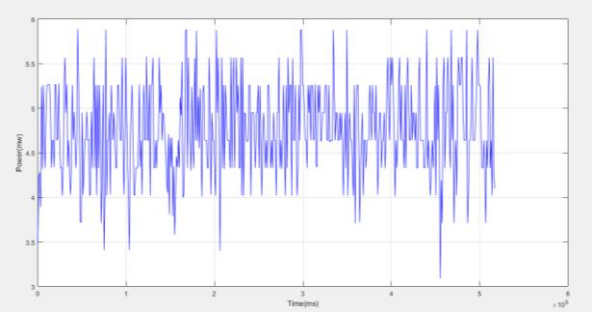
با توجه به اینکه تاثیر برنامه‌های IOT در زمینه‌های مختلف زندگی بشر روز به روز در حال افزایش است و بر اساس برآوردها در اواخر سال ۲۰۲۵ بیش از۹دستگاه برای هر نفر وجود خواهدداشت. اهمیت غلبه بر چالش‌های قابلیت اطمینان و توان مصرفی، بیشتر و بیشتر می شود. برای حفظ سطح مطلوبی از قابلیت اطمینان در برنامه‌های کاربردی IOT که سربار زیادی را در شبکه تحمیل می‌کنند و باعث افزایش مصرف انرژی می‌شوند، در این مقاله ما با ارزیابی دو پروتکل مشهور لایه‌ی کاربرد یعنی MQTT با بسیاری از ویژگی‌های قابل اعتمادتر وCoAP با قابلیت اعتماد کمتر، ولتاژ و جریان مصرفی ودر نهایت محاسبه‌ی توان مصرفی هر پروتکل را اندازه گیری کردیم. انجام این آزمایش برروی یک محیط سخت افزاری واقعی IOTیعنی مدار طراحی شده که در شکل ۵ می‌بینید، انجام شده است. مشاهدات ما نشان می‌دهد در حالی که MQTT زیر ساخت‌های قابل اعتمادتری را درIOT ارائه می‌دهد اما مصرف انرژی آن به طور میانگین از پروتکلCoAP بیشتر است.

به طور کلی پروتکلMQTT برای سیستم‌های IOT مناسب تر است و دارای قابلیت اطمینان قابل توجهی است در حالی که پروتکلCoAP می‌تواند یک انتخاب مناسب برای محیط IOT درزمان واقعی باشد.

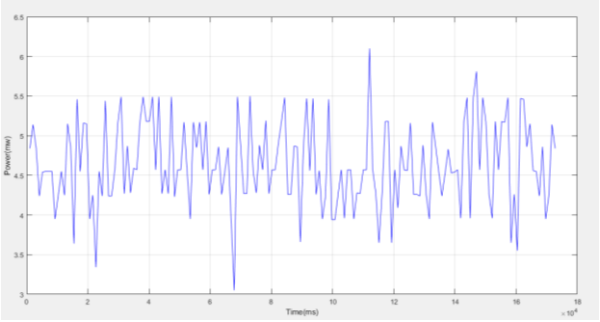
کار پیشنهادی در آینده با توجه با اهمیت چالش توان مصرفی در محیط‌هایIOT، پیاده سازی روش‌های کدگذاری کم توان در طراحی ساختار پروتکل‌ها و ایجاد بستری برای مصرف انرژی کمتر در این صنعت است.



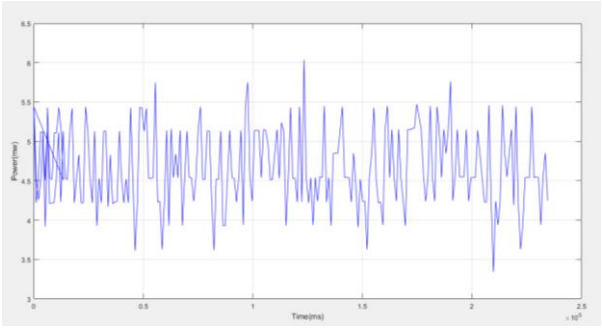
شکل(۵):مدار طراحی شده برای اندازه گیری جریان و توان مصرفی



شکل(٦):نمودار توان مصرفی پروتکلMQTT



شکل(۷):نمودار توان مصرفی پروتکلCoAP server



شکل(۸):نمودار توان مصرفی پروتکلCoAP client

مراجع

[۱] م. رفیقی، ن. معتضدی، "بررسی پروتکل‌ها و استانداردهای اینترنت اشیاء در لایه‌های مختلف"، علوم رایانشی، صفحه ۵۶، زمستان۱۳۹۶.

[2] B.Safaei, A. Hosseini Monazzah, M. Barzegar Bafroei, A. Ejlali," Reliability Side-Effects in Internet of Things Application Layer Protocols", 2nd International Conference on System Reliability and Safety, Milan, Italy, IEEE,pp . 207-212, 01 February 2018.

[3] N.Naik, " Choice of Effective Messaging Protocols for IoT Systems: MQTT, CoAP, AMQP and HTTP", IEEE International Systems Engineering Symposium (ISSE), Vienna, Austria, IEEE, pp. 1-6, 30 October 2017.

[4] A.Hornsby, R. Walsh, " From instant messaging to cloud computing, an XMPP review", IEEE International Symposium on Consumer Electronics (ISCE 2010), Braunschweig, Germany, pp. 1-2, 26 July 2010.

[5] T. Salman, R. Jain, " Networking protocols and standards for internet of things", Advanced Computing and Communications, Vol. 1, No. 1, pp. 15-20, March 2017.

[6] M. B. Yassein, M. Q. Shatnawi, D. Al-Zoubi, " Application layer protocols for the Internet of Things: A survey", 2016 International Conference on Engineering & MIS (ICEMIS), Agadir, Morocco, pp. 1-4, IEEE, 17 November 2016.

[7] M.Kashyap, V. Sharma, N. Gupta, " Taking MQTT and NodeMcu to IOT: Communication in Internet of Things", International Conference on Computational Intelligence and Data Science(ICCIDS2018), Procedia Computer Science 132 (2018) 1611–1618.

[8] Leloglu E. A Review of Security Concerns inInternet of Things, Journal of Computer and Communication 5, 121-136, 2017.

[9] Henkel, Jörg, Santiago Pagani, Hussam Amrouch, Lars Bauer, and Farzad Samie. "Ultra-low power and dependability for IoT devices (Invited paper for IoT technologies), " In 2017 Design, Automation & Test in Europe Conference & Exhibition (DATE), pp. 954-959. IEEE, 2017

[10] Bhat, Rashmi Chalia, Rahil Reyaz, and Najla Andrabi. "A Comprehensive Study on Power Consumption in IOT", (2018)

[11] Jorg Henkel, Santiago Pagani, Hussam Amrouch, Lars Bauer, and Farzad Samie, Ultra-Low Power and Dependability for IoT Devices, Published 2017 Design, Automation and Test in Europe (DATE) 954-959, 2017

[12] Mauro Conti, Ali Dehghantanha, Katrin Franke, Steve Watson. Internet of Things security and forensics: Challenges and opportunities.Available online 26 July 2017.