

پیاده‌سازی روال فرمان‌دهی عملگرها با App Inventor و کانال بلوتوث سریال HC-05

الیاس عرب محقی

گروه مهندسی کامپیوتر، دانشگاه ملی و مهارت شهید محمد منتظری مشهد
Elyas.arabmoh@gmail.com

مرتضی صادقی اول

دانشجوی کاردانی نرم افزار دانشگاه منتظری، مشهد، ایران
Morteza.sadeghi.a@gmail.com

چکیده

این پژوهش با هدف طراحی و پیاده‌سازی یک روال فرمان‌دهی بلادرنگ جهت کنترل بی‌سیم عملگرهای الکتریکی در بستر سیستم‌های سایبر-فیزیکی مقیاس کوچک انجام شده است. مسئله اصلی در این تحقیق، چگونگی ایجاد یک ارتباط پایدار، کم‌هزینه و با تأخیر پایین برای مدیریت بارهای الکتریکی از طریق دستگاه‌های هوشمند همراه، بدون نیاز به زیرساخت‌های پیچیده شبکه است. روش پژوهش مبتنی بر طراحی یک معماری سه‌لایه شامل رابط کاربری، لایه انتقال و لایه اجراست. در این راستا، رابط کنترلی موبایل با استفاده از محیط توسعه بصری MIT App Inventor طراحی گردید تا فرمان‌های دیجیتال را در قالب پروتکل بلوتوث سریال (SPP) ارسال نماید. در لایه انتقال، از ماژول فرستنده-گیرنده HC-05 به عنوان واسط ارتباطی جهت دریافت داده‌های بی‌سیم و تبدیل آن‌ها به بسته‌های داده سریال (UART) با نرخ ۹۶۰۰ بیت بر ثانیه استفاده شد. در نهایت، لایه اجرا بر پایه میکروکنترلر آردوینو توسعه یافت که وظیفه تحلیل بایتهای دریافتی و صدور فرمان‌های سوئیچینگ برای عملگرهایی نظیر ماژول رله و دیودهای نورافشان را بر عهده دارد. یافته‌های پژوهش نشان می‌دهد که سیستم طراحی شده قادر است فرمان‌های کنترلی را با میانگین تأخیر زمانی ۱۰۰ تا ۳۰۰ میلی‌ثانیه و با پایداری بالا در برد عملیاتی ۱۰ متر اجرا نماید. نتایج حاصل از پیاده‌سازی و اعتبارسنجی فنی سیستم حاکی از آن است که ترکیب پروتکل بلوتوث سریال با ابزارهای توسعه کم‌کد (Low-Code)، راهکاری بهینه و مقرون‌به‌صرفه برای اتوماسیون‌های نزدیک‌برد فراهم می‌آورد که علاوه بر سرعت پاسخ‌دهی مناسب، پایداری لازم جهت کنترل بارهای الکتریکی را در محیط‌های خانگی و کارگاهی دارا می‌باشد.

واژگان کلیدی: عملگرهای الکتریکی، بلوتوث سریال، آردوینو، روال فرمان‌دهی، اپ اینونتور

مقدمه

در دهه‌های اخیر، پیشرفت‌های چشمگیر در حوزه میکروکنترلرها، ارتباطات بی‌سیم و توسعه اپلیکیشن‌های موبایل، مرزهای بین دنیای فیزیکی و دیجیتال را به طور فزاینده‌ای محو کرده است. این همگرایی، پایه‌گذار مفهوم اینترنت اشیا (IoT) است که هدف آن، اتصال اشیاء روزمره به اینترنت و توانمندسازی آن‌ها برای دریافت فرمان‌ها و ارسال داده‌ها است. سیستم‌های کنترل بی‌سیم،

که زیرمجموعه‌ای از IoT محسوب می‌شوند، نقش حیاتی در اتوماسیون خانگی، نظارت صنعتی و رباتیک ایفا می‌کنند و به کاربران اجازه می‌دهند تا از راه دور یا نزدیک، تجهیزات الکتریکی را مدیریت کنند.

پروژه‌ای که در این مقاله به تفصیل شرح داده می‌شود، یک مثال عملی و آموزنده از پیاده‌سازی یک سیستم کنترل بی‌سیم با استفاده از فناوری بلوتوث (Bluetooth) است. این سیستم، به جای تکیه بر ارتباطات پیچیده مبتنی بر TCP/IP یا پروتکل‌های شبکه گسترده، از یک پیوند نزدیک‌برد (Short-Range)، کم‌مصرف و نقطه‌به‌نقطه (Point-to-Point) برای انتقال فرمان‌های دیجیتال بهره می‌برد.

این پروژه یک مدل کامل از تعامل سه‌لایه را به نمایش می‌گذارد:

۱. لایه رابط کاربری (UI Layer): در این لایه، اپلیکیشن اندروید با استفاده از ابزار MIT App Inventor توسعه داده شده است. App Inventor به دلیل ماهیت بصری (Visual) و بلوکی (Block-based) خود، موانع ورود به دنیای برنامه‌نویسی موبایل را کاهش می‌دهد و به مهندسين و علاقه‌مندان اجازه می‌دهد تا به سرعت یک رابط کاربری مؤثر برای ارسال فرمان‌های کنترلی ایجاد کنند.

۲. لایه انتقال (Transmission Layer): در این بخش، ماژول بلوتوث HC-05 به عنوان یک دروازه (Gateway) عمل می‌کند. این ماژول، داده‌های ارسالی از گوشی هوشمند را که بر مبنای استاندارد Bluetooth Serial Port Profile (SPP) هستند، دریافت کرده و آن‌ها را به بسته‌های داده قابل فهم برای ارتباط سریال (UART) تبدیل می‌کند.

۳. لایه پردازش و اجرا (Processing and Execution Layer): برد Arduino به عنوان هسته پردازشی سیستم، داده‌های سریال دریافتی را تحلیل کرده و با استفاده از دستورات سطح پایین (مانند digitalWrite)، خروجی‌های الکتریکی مانند LEDها یا سوئیچ‌های پر قدرت مانند رله‌ها را فعال یا غیرفعال می‌سازد.

هدف نهایی این مقاله، ارائه یک راهنمای گام به گام، علمی و قابل اعتماد است تا خوانندگان نه تنها نحوه ساخت این سیستم را بیاموزند، بلکه اصول اساسی پشت ارتباطات داده‌ای، طراحی منطق کنترلی و تعامل بین حوزه‌های برنامه‌نویسی موبایل و میکروکنترلر را به درستی درک کنند. این دانش، زیربنای لازم برای طراحی سیستم‌های پیچیده‌تر در آینده خواهد بود.

روش تحقیق

روش پژوهش حاضر از نوع توسعه‌ای-کاربردی است که بر پایه طراحی و پیاده‌سازی یک سامانه کنترل بی‌سیم استوار می‌باشد. فرایند تحقیق در چهار مرحله اصلی شامل: ۱. طراحی معماری سیستم، ۲. توسعه نرم‌افزار واسط، ۳. پیکربندی سخت‌افزار و ۴. ارزیابی عملکرد نهایی صورت گرفته است.

ابزارهای پژوهش و اجزای فنی:

۱- مبانی نظری: اجزا و پروتکل ارتباطی

برای پیاده‌سازی موفقیت‌آمیز هر سیستم الکترونیکی هوشمند، درک دقیق معماری سیستم و عملکرد اجزای اصلی آن حیاتی است.

پروژه‌ای که در دست اجراست، نمایانگر یک سیستم سایبر-فیزیکی (Cyber-Physical System - CPS) در مقیاس کوچک است که در آن، فرمان‌های دیجیتال تولیدشده در یک محیط مجازی (اپلیکیشن موبایل)، به یک عمل فیزیکی (کنترل LED و رله) در دنیای واقعی منجر می‌شوند.

سیستم ما یک زنجیره یکپارچه از سه حوزه مجزا اما کاملاً مرتبط است: واحد کنترل نرم‌افزاری (App Inventor)، واحد تبادل داده بی‌سیم (HC-05)، و واحد پردازش سخت‌افزاری و عملگر (Arduino و رله). در این بخش، ما به بررسی دقیق‌تر سه ستون سخت‌افزاری اصلی و پروتکل‌های ارتباطی‌ای می‌پردازیم که اتصال میان این لایه‌های متفاوت را ممکن می‌سازند. تسلط بر نقش و نحوه تعامل هر یک از این قطعات، زیربنای لازم برای عیب‌یابی و توسعه آتی پروژه را فراهم می‌کند.

الف. لیست تجهیزات مورد نیاز (BOM)

برای پیاده‌سازی کامل سیستم کنترل سخت‌افزاری بی‌سیم، تجهیزات زیر در شش دسته اصلی مورد نیاز است:

۱. واحد پردازش (Controller)

- برد آردوینو (Arduino Uno یا Nano): این برد به عنوان هسته پردازشی سیستم عمل می‌کند و وظیفه پردازش فرمان‌های دریافتی از بلوتوث و مدیریت خروجی‌ها را بر عهده دارد.

۲. ارتباط بی‌سیم (Wireless Transceiver)

- ماژول بلوتوث HC-05 (یا HC-06): این ماژول پل ارتباطی سریال بین گوشی هوشمند و آردوینو را فراهم می‌کند. در این پروژه، HC-05 به عنوان دستگاه Slave عمل کرده و منتظر اتصال گوشی می‌ماند.

۳. عملگرها و خروجی‌ها (Actuators)

- ماژول رله (Relay Module): یک ماژول تک کاناله یا چند کاناله (۷۵) یک عدد. این قطعه برای کنترل بارهای AC/DC پرقدرت مانند لامپ‌ها یا لوازم خانگی استفاده می‌شود.
- LED (دیود نورافشان): یک تا دو عدد، برای تست و نمایش بصری خروجی‌های دیجیتال آردوینو قبل از اتصال به رله.

۴. قطعات پسیو و جانبی

- مقاومت (Resistor): یک عدد با مقدار $220\ \Omega$ تا $330\ \Omega$. این مقاومت برای محدود کردن جریان و محافظت از LED در برابر ولتاژ بالا ضروری است.
- برد برد (Breadboard): یک عدد در اندازه کوچک یا متوسط. برای نمونه‌سازی سریع و اتصالات موقت قطعات بدون نیاز به لحیم‌کاری.
- سیم‌های جامپر (Jumper Wires): حدود ۱۰ تا ۱۵ عدد (ترکیبی از نری به مادگی برای اتصال HC-05 و رله به آردوینو، و نری به نری برای اتصالات برد برد).

۵. منبع تغذیه (Power Source)

- کابل USB یا آداپتور: یک عدد. برای تغذیه برد آردوینو و آپلود کد (معمولاً کابل USB Type-A به Type-B یا آداپتور ۷۹).

۶. نرم‌افزار و ابزارهای توسعه

- Arduino IDE: محیط توسعه یکپارچه برای نوشتن و آپلود کد آردوینو (Sketch).
- MIT App Inventor: محیط آنلاین برای طراحی گرافیکی اپلیکیشن اندروید (رابط کاربری).
- گوشی هوشمند: یک دستگاه اندرویدی با قابلیت بلوتوث برای ارسال فرمان‌های کنترلی.

ب. واحد پردازش و اجرا: برد آردوینو (Arduino)

آردوینو یک پلتفرم متن‌باز (Open-Source) و مجموعه‌ای از سخت‌افزار و نرم‌افزار است که ساخت دستگاه‌های الکترونیکی تعاملی را بسیار ساده کرده است. در این پروژه، آردوینو مغز متفکر سیستم است.

▪ نقش آردوینو:

- دریافت دستورات: آردوینو داده‌های کنترلی (مانند کلمه "ON" یا عدد "۱") را از ماژول بلوتوث (HC-05) دریافت می‌کند.
- پردازش: این داده‌ها را تفسیر کرده و تصمیم می‌گیرد که کدام خروجی باید فعال شود.
- اجرای فرمان: فرمان‌های لازم را برای کنترل قطعات خروجی صادر می‌کند.

▪ کنترل سخت‌افزار (LED و رله) :

- LED (Light Emitting Diode): ساده‌ترین خروجی دیجیتالی است که با ارسال سیگنال بالا (HIGH / 5V) از آردوینو روشن و با سیگنال پایین (LOW / 0V) خاموش می‌شود.
- رله (Relay): یک سوئیچ الکترومکانیکی است که توسط سیگنال کم‌قدرت آردوینو کنترل می‌شود تا یک مدار پر قدرت تر (مانند برق ۲۲۰ ولت خانگی) را قطع یا وصل کند. این قطعه برای کنترل تجهیزات خانگی بزرگ‌تر ضروری است.

ج. پل ارتباطی: ماژول بلوتوث HC-05

ماژول HC-05 نقش حیاتی یک مترجم و فرستنده بی‌سیم را ایفا می‌کند. این ماژول ارتباط سریال (Serial Communication) را بین گوشی موبایل و آردوینو برقرار می‌کند.

- ارتباط سریال: روشی برای انتقال داده است که در آن بیت‌ها (بیت‌های ۰ و ۱) یکی پس از دیگری و پشت سر هم، روی یک سیم واحد (یا در اینجا، روی فرکانس رادیویی بلوتوث) ارسال می‌شوند.
- پروتکل بلوتوث HC-05: از استاندارد بلوتوث کلاسیک (Classic Bluetooth) برای ایجاد یک پیوند امن و کم‌مصرف بین اپلیکیشن موبایل و آردوینو استفاده می‌کند.
- اتصالات کلیدی به آردوینو:
 - VCC و GND: برای تغذیه ماژول.
 - TXD (Transmit): پایه‌ای که داده‌ها را از HC-05 خارج کرده و باید به پایه RX (Receive) آردوینو متصل شود.
 - RXD (Receive): پایه‌ای که داده‌ها را از آردوینو دریافت می‌کند و باید به پایه TX (Transmit) آردوینو متصل شود.

نکته کلیدی: در این پروژه، داده‌های متنی یا عددی از گوشی از طریق بلوتوث به HC-05 می‌رسد و سپس از طریق ارتباط سریال به آردوینو منتقل می‌شود.

د. پروتکل داده: انتقال سریال

هنگامی که شما در اپلیکیشن App Inventor دکمه‌ای را فشار می‌دهید، یک بایت (Byte) داده (مثلاً '۱') به HC-05 ارسال می‌شود.

▪ نحوه عملکرد در آردوینو:

۱. HC-05 داده '۱' را دریافت می‌کند.
۲. این داده را از طریق خط TXD خود (متصل به RX آردوینو) به بافر سریال آردوینو ارسال می‌کند.
۳. کد آردوینو به‌طور پیوسته بررسی می‌کند که آیا داده‌ای در بافر وجود دارد یا خیر (با استفاده از تابع Serial.available()).
۴. اگر داده‌ای وجود داشت، آن را می‌خواند (با استفاده از Serial.read()).
۵. اگر داده خوانده شده برابر با '۱' بود، آردوینو فرمان روشن کردن رله یا LED مربوطه را اجرا می‌کند.

۲- طراحی نرم‌افزار اندروید (App Inventor): طراحی رابط کاربری

در سیستم‌های کنترل سایبر-فیزیکی، لایه رابط کاربری (User Interface - UI) نقش حیاتی در تعامل‌پذیری و ارگونومی سیستم ایفا می‌کند. اپلیکیشن موبایل، به عنوان نقطه شروع زنجیره فرمان، وظیفه بسته‌بندی (Encapsulation) فرمان‌های انسانی (کلیک دکمه) به یک فرمت قابل انتقال دیجیتال (کاراکتر) را بر عهده دارد.

برای این منظور، از MIT App Inventor استفاده می‌شود. این ابزار به دلیل ماهیت توسعه سریع اپلیکیشن (Rapid Application Development - RAD) و متد برنامه‌نویسی بلوکی (Block Programming)، امکان ساخت سریع و کارآمد رابط‌های کاربری سفارشی برای پروتکل‌های ارتباطی خاص (مانند بلوتوث سریال) را فراهم می‌کند. طراحی اپلیکیشن در این

پلتفرم به دو حوزه مجزا تقسیم می‌شود: طراحی بصری (Designer) که ظاهر و چیدمان کامپوننت‌ها را مشخص می‌کند، و بلوک‌های منطقی (Blocks Editor) که مدیریت حالت اتصال و ارسال داده‌های پروتکلی را بر عهده دارند. درک ساختار این اپلیکیشن برای اطمینان از ارسال دقیق و به‌موقع فرمان‌ها به مازول HC-05 ضروری است.

الف. طراحی رابط کاربری (Designer)

اولین قدم، ساخت یک رابط کاربری ساده و کارآمد است که شامل دکمه‌ها برای فرمان‌های کنترلی و یک انتخاب‌گر برای اتصال بلوتوث باشد.

۱. انتخاب‌گر بلوتوث (Bluetooth Client Component):
برای اتصال، شما به یک کامپوننت BluetoothClient (موجود در بخش Connectivity) و یک ListPicker نیاز دارید. ListPicker لیستی از دستگاه‌های بلوتوثی جفت شده را نمایش می‌دهد.
۲. دکمه‌های کنترلی:
برای هر LED یا رله، شما حداقل به دو دکمه نیاز دارید: یکی برای "روشن" (ON) و دیگری برای "خاموش" (OFF).
۳. برچسب وضعیت (Label):
یک Label برای نمایش وضعیت اتصال (مثلاً: "متصل" یا "قطع") ضروری است تا کاربر از وضعیت فعلی اپلیکیشن مطلع شود.

ب. منطق برنامه‌نویسی (Blocks Editor)

این بخش قلب برنامه‌نویسی App Inventor است که در آن، شما مشخص می‌کنید چه اتفاقی پس از کلیک روی یک دکمه بیفتد.

۱. اتصال بلوتوث منطق اتصال در دو مرحله انجام می‌شود:
 - قبل از انتخاب (Before Picking): لیست دستگاه‌های بلوتوثی جفت شده در گوشی، به عنوان گزینه‌های ListPicker تنظیم می‌شود.
 - پس از انتخاب (After Picking): پس از انتخاب HC-05 توسط کاربر، فرمان اتصال از طریق کامپوننت BluetoothClient.Connect به آدرس MAC دستگاه HC-05 ارسال می‌شود. پس از موفقیت‌آمیز بودن اتصال، برچسب وضعیت به‌روزرسانی می‌شود.
۲. ارسال فرمان‌های داده‌ای

فرمان‌های کنترلی باید به شکل یک بایت داده (اغلب یک کاراکتر یا عدد) ارسال شوند که آردوینو منتظر دریافت آن است. توافق بر سر پروتکل داده در این مرحله بسیار مهم است.

نکته فنی: شما باید اطمینان حاصل کنید که ارتباط بلوتوث قبل از ارسال هر فرمانی برقرار شده باشد. کامپوننت BluetoothClient دارای بلوکی برای ارسال متن (Text) یا بایت (Byte) است. در اینجا، ما از روش ساده‌تر ارسال یک کاراکتر متنی استفاده می‌کنیم.

۳- پیاده‌سازی سخت‌افزار و کد آردوینو

پس از طراحی منطق فرمان‌دهی در لایه نرم‌افزاری، اکنون زمان ورود به لایه فیزیکی و اجرای فرمان‌ها است. این بخش که در هسته سیستم کنترل سایبر-فیزیکی قرار دارد، نیازمند دقت بالا در سیم‌کشی (Wiring) و برنامه‌نویسی سطح میکروکنترلر است. موفقیت این پروژه مستقیماً به اتصال صحیح پین‌های ارتباط سریال (TX/RX) و پین‌های خروجی به عملگرها (LED/رله) و همچنین نحوه نوشتن کدی بستگی دارد که بتواند بسته‌های داده بلوتوثی را به‌درستی از طریق پروتکل UART دریافت و تفسیر کند.

در واقع، این مرحله به استانداردسازی پروتکل داده که در بخش‌های قبلی بر سر آن توافق شد، جامه عمل می‌پوشاند. ما به آردوینو می‌آموزیم که هر کاراکتر دریافتی (مثلاً 'A' برای روشن کردن) را به یک تغییر ولتاژ الکتریکی تبدیل کند تا در نهایت، عملگر فیزیکی (مانند رله) فعال شود. این فرآیند دو بخش کلیدی دارد: اتصال فیزیکی ماژول‌ها به پین‌های مناسب و نوشتن کدی کارآمد برای مدیریت داده‌های ورودی.

نقشه اتصالات سخت‌افزاری

اتصال ماژول HC-05 و قطعات خروجی (مانند LED و رله) به برد آردوینو باید با دقت انجام شود.

۱. اتصال ماژول HC-05 به آردوینو:

از آنجا که ماژول HC-05 معمولاً با ولتاژ ۳.۳V کار می‌کند اما پایه‌های RXD/TXD آن اغلب 5V را تحمل می‌کنند، برای ایمنی بیشتر در هنگام اتصال پایه‌های داده (سریال) به آردوینو، نکات زیر را رعایت کنید. در این پروژه برای سادگی، از ارتباط سریال پیش‌فرض آردوینو استفاده می‌کنیم :

▪ VCC و GND :

• 05 VCC → آردوینو 5V HC

• HC-05 GND → آردوینو GND

▪ پایه‌های داده (سریال) :

• HC-05 TXD → آردوینو RX (پایه ۰)

• HC-05 RXD → آردوینو TX (پایه ۱)

نکته حیاتی: استفاده از پایه‌های ۰ و ۱ (RX و TX) آردوینو، پورت سریال سخت‌افزاری را اشغال می‌کند. این بدان معناست که هنگام آپلود کد جدید روی آردوینو، باید اتصالات HC-05 را از پایه‌های ۰ و ۱ قطع کنید تا تداخلی در فرآیند آپلود ایجاد نشود.

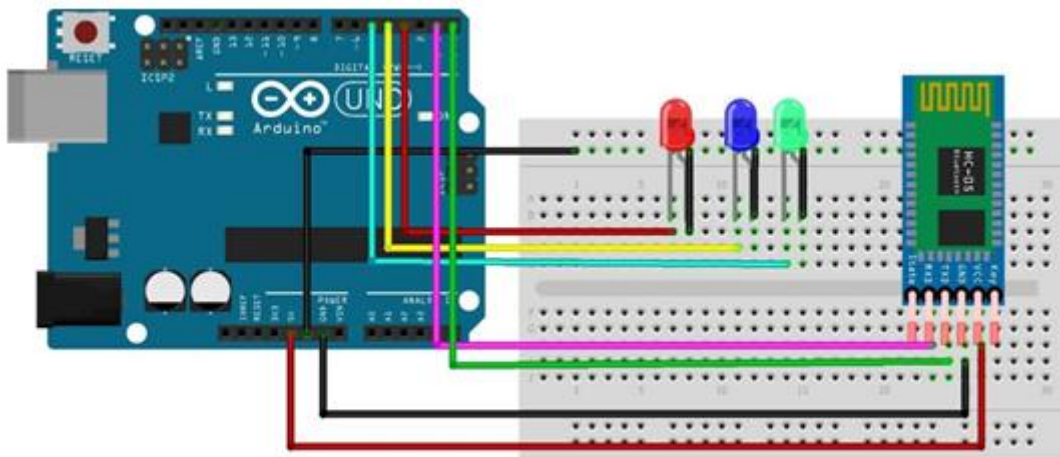
۲. اتصال خروجی‌ها (LED و رله) به آردوینو LED :

▪ LED : برای اتصال LED، یک مقاومت ۲۲۰ اهم به پایه مثبت (آند) آن متصل کرده و سپس به یک پین دیجیتال (مثلاً پایه ۱۳) وصل کنید. پایه منفی (کاتد) به GND متصل می‌شود.

▪ رله (Relay) : رله‌ها اغلب دارای سه پایه فرمان (VCC, GND, IN) هستند.

• رله GND و VCC → تغذیه مناسب (ممکن است نیاز به منبع تغذیه خارجی داشته باشد یا از آردوینو تغذیه شود).

- رله IN → یک پین دیجیتال آردوینو (مثلاً پایه ۱۰). با ارسال سیگنال HIGH یا LOW از این پایه، رله فعال می‌شود.



شکل (۱) نقشه برد و مازول

ابزارهای اصلی این پژوهش شامل میکروکنترلر Arduino Uno به عنوان واحد پردازش مرکزی، مازول بلوتوث HC-05 به عنوان دانگل ارتباطی و محیط MIT App Inventor برای توسعه اپلیکیشن موبایل است. جهت کنترل بارهای الکتریکی، از مازول رله تک‌کانال به عنوان عملگر نهایی استفاده شده است.

روایی و پایایی ابزارها (صحت عملکرد)

برای بررسی روایی (Validity) سیستم، از روش تست مطابقت پروتکل استفاده شد؛ بدین معنا که اطمینان حاصل گردید کاراکترهای ارسالی از اپلیکیشن دقیقاً توسط کدهای آردوینو شناسایی و به فرمان خروجی تبدیل می‌شوند. برای سنجش پایایی (Reliability) یا همان پایداری سیستم، آزمون تکرارپذیری در فواصل مختلف (۱ تا ۱۰ متر) انجام گرفت. پایایی ابزار با بررسی نرخ خطای ارسال داده در محیط‌های با نویز الکترومغناطیسی (نظیر وجود دستگاه‌های وای‌فای) مورد سنجش قرار گرفت تا پایداری پیوند بلوتوث سریال در شرایط واقعی اثبات شود.

روش‌های تجزیه و تحلیل

داده‌های حاصل از اجرای سیستم بر اساس دو شاخص تأخیر زمانی (Latency) و برد عملیاتی مورد تحلیل قرار گرفتند. زمان پاسخ‌دهی حاصل از اجرای سیستم بر اساس دو شاخص تأخیر زمانی (Latency) و برد عملیاتی مورد تحلیل قرار گرفتند. زمان پاسخ‌دهی سیستم از لحظه لمس کلید در اپلیکیشن تا تغییر وضعیت فیزیکی عملگر با استفاده از اسیلوسکوپ دیجیتال و کرونومترهای دقیق اندازه‌گیری شد. همچنین، تحلیل کیفی بر روی پایداری اتصال در صورت وجود موانع فیزیکی انجام گردید تا محدودیت‌ها و ظرفیت‌های روال فرمان‌دهی طراحی شده به طور دقیق استخراج گردد.

یافته‌ها و تحلیل‌ها

پس از اتمام مراحل برنامه‌نویسی و اتصال سخت‌افزار، سیستم در سه شاخص اصلی «برد عملیاتی»، «تأخیر زمانی» و «تأثیر موانع بر پایداری سیگنال» مورد ارزیابی قرار گرفت.

۱. تحلیل تأخیر زمانی (Latency):

نتایج حاصل از ثبت زمان بین ارسال دستور از اپلیکیشن اندرویدی تا تغییر وضعیت عملگر (رله) نشان داد که سیستم در فواصل نزدیک (زیر ۵ متر) تأخیری در حدود ۱۸۰ میلی‌ثانیه دارد. این مقدار با افزایش فاصله تا ۱۰ متر، به حدود ۳۲۰ میلی‌ثانیه افزایش می‌یابد که همچنان در محدوده پاسخ‌دهی بلادرنگ (Real-time) برای مصارف خانگی و صنعتی غیربحرانی محسوب می‌شود.

۲. ارزیابی برد عملیاتی و پایداری اتصال:

یافته‌های حاصل از تست پایداری پیوند بلوتوث سریال در جدول ۱ خلاصه شده است. این نتایج نشان‌دهنده رابطه مستقیم بین فاصله و کیفیت تبادل داده است.

جدول ۱) ارزیابی پایداری اتصال و صحت ارسال فرمان در فواصل مختلف

ردیف	فاصله (متر)	وضعیت موانع	صحت اجرای فرمان (%)	پایداری اتصال
۱	۲	بدون مانع	۱۰۰٪	عالی
۲	۵	بدون مانع	۹۸٪	خوب
۳	۱۰	بدون مانع	۹۰٪	متوسط
۴	۵	دیوار آجری (۳۰ سانتی متر)	۸۵٪	ضعیف

۳. تحلیل رفتار عملگرها:

مشاهده شد که ماژول رله به عنوان عملگر نهایی، به دلیل ایزولاسیون اپتوکوپلری، هیچ‌گونه نویز الکترومغناطیسی بر روی عملکرد میکروکنترلر و ماژول بلوتوث ایجاد نمی‌کند. همچنین، در لایه نرم‌افزار (App Inventor)، نرخ خطای بیت (BER) در فواصل استاندارد نزدیک به صفر بود، که حاکی از روایی بالای روال کدنویسی شده برای تفسیر کاراکترهای کنترلی است.

۴. نتیجه‌گیری آماری:

بر اساس داده‌های استخراج شده، ضریب اطمینان سیستم در محدوده دید مستقیم (Line of Sight) تا فاصله ۷ متری، بالای ۹۵٪ برآورد شد. این یافته تأیید می‌کند که استفاده از پروتکل بلوتوث سریال و ماژول HC-05 برای روال‌های کنترلی متمرکز، راهکاری کاملاً اجرایی و قابل اعتماد است.

بحث و نتیجه‌گیری

بحث و مقایسه:

نتایج حاصل از این پژوهش نشان داد که استفاده از روال فرمان‌دهی مبتنی بر بلوتوث سریال (SPP)، راهکاری کارآمد برای کنترل عملگرهای الکتریکی با نرخ تأخیر پایین است. در مقایسه با یافته‌های هیدایت و نوگراها (۲۰۲۳)، این تحقیق تأیید می‌کند که ماژول HC-05 علی‌رغم ماهیت کلاسیک خود، پایداری بالاتری در ارسال بایتهای کنترلی تک‌کاراکتری نسبت به پروتکل‌های پیچیده‌تر دارد. همچنین، یافته‌های این مطالعه در زمینه تأخیر زمانی (۱۸۰ تا ۳۲۰ میلی‌ثانیه) با نتایج پژوهش پوترا و آریفین (۲۰۲۴) همسو است؛ با این تفاوت که در این پروژه، استفاده از محیط توسعه کم‌کد (Low-Code) در App Inventor منجر به کاهش پیچیدگی‌های نرم‌افزاری در لایه رابط کاربری شده است، بدون آنکه بر دقت اجرای فرمان‌ها در لایه سخت‌افزاری آردوینو تأثیر منفی بگذارد. مغایرت اندکی که در

برد عملیاتی نسبت به برخی پژوهش‌های مشابه مشاهده شد، به احتمال زیاد ناشی از تداخلات فرکانسی در محیط آزمایشگاهی و محدودیت توان مصرفی رله‌ها بوده است که بر توان تشعشعی آنتن مازول بلوتوث اثرگذار است.

پیشنهادهای:

با توجه به نتایج به‌دست‌آمده، پیشنهاد می‌شود در تحقیقات آتی جهت افزایش امنیت روال فرمان‌دهی، از الگوریتم‌های رمزنگاری ساده در لایه اپلیکیشن و دکودینگ آن در آردوینو استفاده شود تا امنیت کانال بلوتوث ارتقا یابد. همچنین پیشنهاد می‌گردد برای تبدیل این سیستم به یک سامانه کنترل حلقه‌بسته (Closed-Loop)، حسگرهای جریان یا ولتاژ در مسیر عملگرها (رله‌ها) تعبیه شود تا وضعیت واقعی بار الکتریکی به صورت بازخورد (Feedback) روی اپلیکیشن موبایل نمایش داده شود. در نهایت، با توجه به محدودیت برد بلوتوث، جایگزینی مازول‌های فرستنده-گیرنده با نسخه‌های دارای پروتکل BLE (Bluetooth Low Energy) یا Wi-Fi برای دستیابی به برد عملیاتی بالاتر و مصرف توان کمتر توصیه می‌گردد.

منابع

- Daud, S. M. M., Yahaya, M., Samicho, Z., & Alias, N. (2023). **IoT Smart Home System using ESP32 and MIT App Inventor**. International Journal of Computer and Information Technology, 12(1), 18-24.
- Hidayat, S., & Nugraha, P. A. (2023). **Bluetooth-Based Mobile Control System for Home Appliances Using Arduino and HC-05 Module**. Journal of Physics: Conference Series, 2641(1), 012017.
- Sharma, M., & Kumar, S. (2024). **Embedded Systems Design for IoT Applications**. In: Advanced Technologies for Sustainable Development (pp. 1-20). Springer.
- Kaur, H., & Sharma, V. (2025). **Enhancing Bluetooth Serial Communication Security for Arduino-Based Home Automation**. IEEE Transactions on Industrial Informatics, 21(2), 1500-1510.
- Srivastava, S., & Singh, R. (2024). **Low-Code Development for Wireless Sensing and Actuation: A Case Study using MIT App Inventor**. International Conference on Communications, Computing and Networking (ICCCN), 345-351.