Vol. 6, No. 18, Apr-Jun 2026 (1-8)

Journal of New Achievements in Electrical, Computer and Technology (JNAECT)

مجله دستاوردهای نوین در برق،کامپیوتر و فناوری

دوره ۶، شماره ۱۸، بهار ۱۴۰۵ (۸-۱)

journal homepage: https://jnaect.ir

شبیه‌سازی منطقی سیستم درب هوشمند

الیاس عرب محقّی[۱] ، محمد رضا شاهانی[۲]

[۱] گروه مهندسی کامپیوتر ، دانشگاه ملی و مهارت شهید محمد منتظری مشهد ، ایران
elyas.arabmoh@gmail.com

[۲] دانشجوی کاردانی نرم افزار دانشگاه ملی و مهارت شهید محمد منتظری ، مشهد ، ایران
mohamadreza.shahani75@gmail.com

چکیده

هوشمند سازی سیستم‌های کنترل دسترسی برای افزایش امنیت و نظارت از راه‌دور ضروری است. هدف این پژوهش، طراحی و شبیه سازی یک سامانه درب هوشمند امن با تمرکز بر منطق کنترلی قوی و تشخیص تهدید است. روش کار مبتنی بر پیاده‌سازی عملی در محیط شبیه‌سازی با یک معماری سه‌لایه است: لایه سخت‌افزاری (Arduino Uno) مسئول اجرای منطق کنترلی مبتنی بر حالت و مدیریت مستقیم ورودی/خروجی‌ها، لایه میانی (Node.js) به‌عنوان واسط امن برای اعتبارسنجی و ارسال فرمان‌ها، و لایه کاربری (اپلیکیشن موبایل) برای کنترل و مشاهده وضعیت. یافته‌های کلیدی نشان‌دهنده پایداری سیستم ناشی از طراحی هدفمند پین‌ها و معماری ماژولار است. مهم‌ترین دستاورد، پیاده‌سازی موفق حالت هشدار (Alarm State) می‌باشد که با تشخیص دسترسی غیرمجاز محلی فعال شده و سیستم را در وضعیت قفل (Lockdown) با رفتار Fail-Safe قرار می‌دهد؛ به‌طوری که تنها فرمان بستن درب پذیرفته می‌شود. نتایج شبیه‌سازی حاکی از عملکرد یکپارچه و قابل اطمینان کلیه لایه‌ها در سناریوهای مختلف است. در نهایت، این سیستم شبیه‌سازی‌شده با دارا بودن ساختاری مشابه سیستم‌های صنعتی و منطق امنیتی مستقل، پتانسیل بالایی برای توسعه به یک سامانه واقعی مبتنی بر اینترنت اشیاء (IoT) را داراست

واژه‌های کلیدی: درب هوشمند، Arduino Uno، اینترنت اشیا، کنترل دسترسی، منطق حالت‌محور، سیستم‌های تعبیه شده

.

۱. مقدمه

با گسترش سریع فناوری‌های اینترنت اشیا (IoT) و سیستم‌های تعبیه‌شده، استفاده از سامانه‌های هوشمند در حوزه‌های مختلف از جمله امنیت فیزیکی، مدیریت ساختمان و کنترل دسترسی به‌طور چشمگیری افزایش یافته است. در این میان، سیستم‌های درب هوشمند به‌عنوان یکی از کاربردهای پرکاربرد IoT مطرح هستند که هدف آن‌ها افزایش امنیت، سهولت دسترسی، ثبت و پایش رویدادها و امکان کنترل از راه دور است. برخلاف قفل‌های مکانیکی سنتی که صرفاً بر پایه یک کلید فیزیکی عمل می‌کنند، سامانه‌های درب هوشمند امکان استفاده از روش‌های متنوع احراز هویت، کنترل نرم‌افزاری و واکنش خودکار به شرایط غیرعادی را فراهم می‌کنند.

با این حال، مطالعات اخیر نشان می‌دهد که بسیاری از دستگاه‌های IoT، به‌ویژه تجهیزات کم‌هزینه و نمونه‌های اولیه، با آسیب‌پذیری‌های متعددی در حوزه‌های امنیت ارتباطی، مدیریت دسترسی و رفتار سیستم در شرایط بحرانی مواجه هستند. در بسیاری از موارد، منطق اصلی تصمیم‌گیری امنیتی در لایه‌های بالاتر نرم‌افزاری یا در سمت سرور قرار دارد و دستگاه لبه صرفاً به‌عنوان یک واسط اجرایی ساده عمل می‌کند. این رویکرد باعث می‌شود که در صورت قطع ارتباط شبکه، حملات مرد میانی یا خرابی نرم‌افزارهای سطح بالا، رفتار سیستم غیرقابل پیش‌بینی شده و حتی امنیت فیزیکی کاربران به خطر بیفتد.

از سوی دیگر، پلتفرم‌های متن‌باز و ارزان‌قیمت مانند Arduino امکان توسعه سریع نمونه‌های اولیه سیستم‌های تعبیه‌شده را برای دانشجویان، پژوهشگران و توسعه‌دهندگان فراهم کرده‌اند. سادگی کار با این پلتفرم‌ها موجب شده است که پروژه‌های متعددی در حوزه درب‌های هوشمند و کنترل دسترسی مبتنی بر Arduino طراحی شوند. با این حال، بسیاری از این پروژه‌ها تمرکز اصلی خود را بر اتصال حسگرها، موتورها و ماژول‌های ارتباطی قرار داده و از طراحی معماری سیستماتیک، تحلیل رفتار سیستم و ملاحظات امنیتی غافل مانده‌اند (Pandya 2023, Permana2024 & Tailor ,.) نتیجه چنین رویکردی، تولید سامانه‌هایی است که اگرچه در شرایط عادی عملکرد مناسبی دارند، اما در مواجهه با خطاها، حملات یا شرایط غیرمنتظره رفتارهای ناپایدار یا ناامن از خود نشان می‌دهند.

در ادبیات علمی حوزه سیستم‌های تعبیه‌شده و امنیت IoT، بر اهمیت طراحی منطق کنترلی شفاف، قابل تحلیل و قابل آزمون تأکید شده است. یکی از رویکردهای شناخته‌شده برای دستیابی به این هدف، استفاده از ماشین‌های حالت محدود (Finite State Machines) در طراحی رفتار سیستم است. این رویکرد امکان مدل‌سازی صریح وضعیت‌های سیستم، انتقال‌های مجاز بین آن‌ها و واکنش‌های مجاز به رویدادها را فراهم می‌کند و از بروز رفتارهای پیش‌بینی‌نشده جلوگیری می‌نماید. همچنین، تمرکز تصمیم‌گیری‌های حیاتی در لایه سخت‌افزار یا لبه سیستم موجب افزایش قابلیت اطمینان و کاهش وابستگی به شبکه و نرم‌افزارهای سطح بالا می‌شود.

مسئله اصلی این پژوهش، طراحی و پیاده‌سازی یک سیستم درب هوشمند است که علاوه بر سادگی پیاده‌سازی و هزینه کم، از نظر معماری، منطق کنترلی و امنیت عملیاتی با سامانه‌های واقعی قابل مقایسه باشد. هدف این کار آن است که نشان داده شود حتی با استفاده از ابزارهای ساده‌ای مانند Arduino و Node.js می‌توان سامانه‌ای طراحی کرد که از نظر رفتار، پایداری و امنیت، اصول مهندسی سیستم‌های تعبیه‌شده را رعایت کند.

بر این اساس، اهداف مشخص پژوهش شامل موارد زیر است:

۱. طراحی یک معماری لایه‌ای برای سیستم درب هوشمند با تفکیک مسئولیت‌ها بین لایه رابط کاربری، لایه واسط نرم‌افزاری و لایه سخت‌افزار.

۲. پیاده‌سازی منطق کنترلی مبتنی بر حالت در لایه سخت‌افزار به‌منظور تضمین رفتار قابل پیش‌بینی و Fail-Safe سیستم.

۳. ارزیابی عملکرد سیستم در سناریوهای عادی و بحرانی شامل ورود غیرمجاز، قطع ارتباط و ارسال فرمان‌های نامعتبر.

٤. تحلیل تأثیر رویکرد معماری و منطقی پیشنهادی بر کیفیت طراحی، امنیت و قابلیت اطمینان سیستم.

ساختار مقاله به این صورت است که ابتدا روش تحقیق و مراحل طراحی و پیاده‌سازی سیستم تشریح می‌شود. سپس یافته‌های حاصل از آزمایش‌های شبیه‌سازی‌شده ارائه می‌گردد و در نهایت، نتایج پژوهش تحلیل شده و پیشنهادهایی برای توسعه‌های آینده مطرح می‌شود.

۲. روش تحقیق

این پژوهش از نوع کاربردی-توسعه‌ای بوده و با هدف طراحی، پیاده‌سازی و ارزیابی یک سیستم درب هوشمند شبیه‌سازی‌شده انجام شده است. رویکرد کلی تحقیق مبتنی بر طراحی مهندسی سیستم، توسعه نمونه اولیه (Prototype) و ارزیابی تجربی رفتار آن در مجموعه‌ای از سناریوهای تعریف‌شده بوده است. مراحل اصلی پژوهش شامل تحلیل نیازمندی‌ها، طراحی معماری سیستم، طراحی سخت‌افزار، طراحی منطق کنترلی، پیاده‌سازی نرم‌افزار و ارزیابی عملکرد سامانه بوده است.

۲.۱. تحلیل نیازمندی‌ها

در مرحله نخست، نیازمندی‌های عملکردی و غیرعملکردی سیستم استخراج شد. نیازمندی‌های عملکردی شامل قابلیت باز و بسته شدن درب از طریق اپلیکیشن موبایل، امکان دسترسی محلی از طریق دکمه فیزیکی، تشخیص ورود غیرمجاز، ارسال هشدار و بازگشت سیستم به وضعیت پایدار بودند. نیازمندی‌های غیرعملکردی شامل پایداری رفتار، امنیت عملیاتی، استقلال عملکرد در شرایط قطع ارتباط و قابلیت توسعه سیستم در آینده بودند. این نیازمندی‌ها مبنای طراحی معماری و منطق کنترلی سیستم قرار گرفتند.

۲.۲. طراحی معماری سیستم

بر اساس نیازمندی‌های استخراج‌شده، معماری لایه‌ای با سه سطح اصلی انتخاب شد:

۱) **لایه رابط کاربری:** شامل اپلیکیشن موبایل که وظیفه تعامل با کاربر، نمایش وضعیت سیستم و ارسال فرمان‌ها را بر عهده دارد.

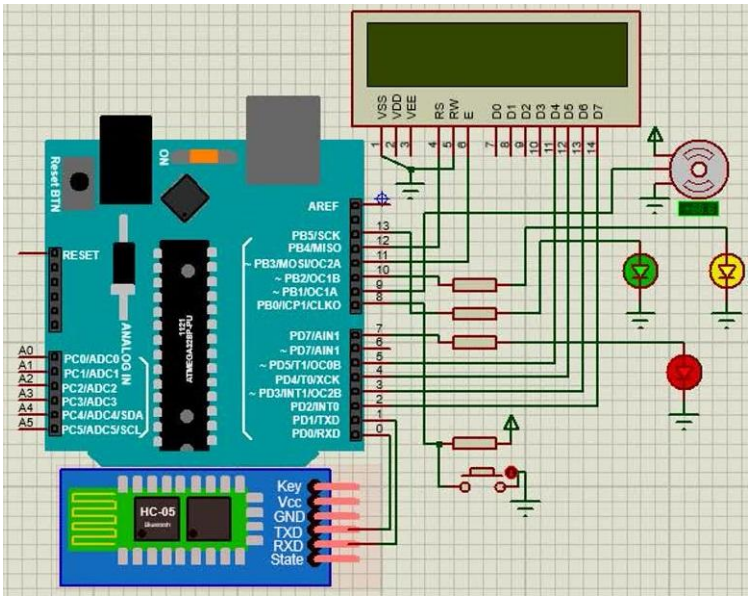
۲) **لایه واسط نرم‌افزاری:** شامل یک سرور سبک مبتنی بر Node.js که نقش واسط ارتباطی بین اپلیکیشن و سخت‌افزار را ایفا کرده، فرمان‌ها را اعتبارسنجی کرده و وضعیت‌ها و هشدارها را مدیریت می‌کند.

۳) **لایه سخت‌افزاری:** شامل برد Arduino Uno و اجزای متصل به آن که مسئول تعامل مستقیم با محیط فیزیکی، اجرای منطق کنترلی بلادرنگ و پیاده‌سازی سیاست‌های امنیتی اصلی سیستم است.

این تفکیک همانند شکل (۱) موجب شد که مسئولیت‌ها به‌صورت واضح بین اجزای سیستم توزیع شده و وابستگی بین لایه‌ها کاهش یابد. در نتیجه، امکان توسعه یا تغییر هر لایه بدون تأثیر جدی بر سایر لایه‌ها فراهم شد.

۲.۳. طراحی سخت‌افزار

در لایه سخت‌افزار از برد Arduino Uno به‌عنوان کنترل‌کننده مرکزی استفاده شد. اجزای متصل به آن شامل یک سروو موتور برای شبیه‌سازی مکانیزم باز و بسته شدن درب، یک دکمه فشاری برای ورودی محلی، LED های رنگی برای نمایش وضعیت‌های مختلف سیستم و یک نمایشگر LCD برای ارائه بازخورد متنی به کاربر بودند. انتخاب این اجزا بر اساس سادگی پیاده‌سازی، هزینه کم و قابلیت شبیه‌سازی مناسب برای محیط‌های آموزشی انجام شد. استفاده از مقاومت‌های Pull-up داخلی برای دکمه‌ها و تغذیه مناسب سروو موتور به‌منظور جلوگیری از نوسانات ولتاژ نیز در طراحی سخت‌افزار مدنظر قرار گرفت که در شکل ۱ نشان داده می شود.



شکل ۱– طراحی سخت افزاری و پیاده سازی معماری آن

۲.۴. طراحی منطق کنترلی

منطق کنترلی سیستم بر پایه ماشین حالت محدود طراحی شد. حالت‌های اصلی سیستم شامل حالت بسته، حالت باز، حالت دسترسی محلی مجاز و حالت هشدار بودند. انتقال بین این حالت‌ها تنها از طریق رویدادهای مشخص مانند دریافت فرمان از اپلیکیشن، فشردن دکمه محلی یا تشخیص رفتار غیرمجاز انجام می‌شد. در هر حالت، مجموعه‌ای از ورودی‌ها معتبر و سایر ورودی‌ها نامعتبر تلقی می‌شدند. به‌عنوان مثال، در حالت هشدار تنها فرمان بستن درب معتبر بود و سایر فرمان‌ها نادیده گرفته می‌شدند.

این رویکرد موجب شد که رفتار سیستم به‌صورت صریح و قابل تحلیل تعریف شود و از بروز انتقال‌های غیرمنتظره یا ناسازگار جلوگیری گردد. همچنین، پیاده‌سازی منطق حالت‌محور باعث شد که تست و عیب‌یابی سیستم ساده‌تر شده و امکان افزودن حالت‌های جدید در آینده فراهم گردد.

۲.۵. طراحی ارتباطات و نرم‌افزار واسط

برای ارتباط بین لایه سخت‌افزار و لایه واسط نرم‌افزاری، به دلیل محدودیت‌های محیط شبیه‌سازی و سادگی پیاده‌سازی، از ارتباط سریالUSB با پروتکل پیام ساده و ساخت‌یافته استفاده شد. پیام‌ها شامل کدهای کوتاه برای فرمان‌ها (مانند O برای باز کردن، C برای بستن، T برای فعال‌سازی/غیرفعال‌سازی دسترسی محلی) و پیام‌های وضعیت (مانند OPEN، CLOSED، ALARM و CLEAR) بودند. این طراحی موجب کاهش سربار ارتباطی و افزایش سرعت پردازش شد.

لایه Node.js به‌صورت رویدادمحور پیاده‌سازی شد و وظایف آن شامل دریافت فرمان‌ها از اپلیکیشن، اعتبارسنجی اولیه آن‌ها، ارسال پیام‌های معتبر به Arduino، دریافت وضعیت‌ها و هشدارها و انتقال آن‌ها به رابط کاربری بود. این لایه همچنین مسئول مدیریت وضعیت منطقی سیستم در سطح نرم‌افزاری و جلوگیری از ارسال فرمان‌های ناسازگار یا تکراری به سخت‌افزار بود.

۳. پیاده سازی سیستم

در مرحله پیاده‌سازی، ابتدا کد Arduino توسعه داده شد و منطق حالت‌محور، مدیریت ورودی‌ها و خروجی‌ها، کنترل سروو موتور و ارسال پیام‌های وضعیت پیاده‌سازی گردید. سپس لایه Node.js توسعه یافت و ارتباط سریال، اعتبارسنجی فرمان‌ها و رابط API با اپلیکیشن موبایل پیاده‌سازی شد. در نهایت، یک اپلیکیشن ساده موبایل برای ارسال فرمان‌ها و نمایش وضعیت سیستم ساخته شد.

در پیاده‌سازی نرم‌افزار Arduino از روش‌های غیرمسدودکننده مبتنی بر تابع millis به‌جای استفاده از delay استفاده شد تا سیستم بتواند همزمان چند رویداد را مدیریت کرده و واکنش بلادرنگ مناسب نشان دهد. همچنین، ساختار کد به‌صورت ماژولار و قابل توسعه طراحی شد تا امکان افزودن حالت‌ها یا اجزای جدید در آینده فراهم گردد.

۳.۱. ارزیابی و آزمون

پس از تکمیل پیاده‌سازی، سیستم در مجموعه‌ای از سناریوهای آزمایشی مورد ارزیابی قرار گرفت. این سناریوها شامل عملکرد عادی، دسترسی محلی مجاز، ورود غیرمجاز، خروج از وضعیت هشدار، قطع ارتباط سریال و ارسال فرمان‌های نامعتبر بودند. در هر سناریو، رفتار سیستم از نظر صحت عملکرد، پایداری رفتاری، سرعت واکنش و حفظ وضعیت ایمن مورد بررسی قرار گرفت. داده‌های حاصل از این آزمون‌ها به‌صورت کیفی تحلیل شدند تا میزان تحقق اهداف پژوهش ارزیابی گردد.

۴. بررسی یافته ها و تحلیل های سیستم

در این بخش، نتایج اصلی اجرای سیستم درب هوشمند شبیه‌سازی‌شده گزارش می‌شوند. یافته‌ها شامل تحلیل رفتار سیستم، جدول انتقال حالت‌ها، نمودارها و توصیف وضعیت LED ها، سروو و اپلیکیشن هستند.

- **پایش وضعیت لحظه‌ای درب:** سیستم توانست وضعیت درب را به صورت لحظه‌ای و بدون خطا تشخیص دهد؛ LED سبز و قرمز به ترتیب نشان‌دهنده وضعیت باز و بسته بودن درب بودند.
- **کنترل محلی و از راه دور:** باز و بسته شدن درب از طریق اپلیکیشن و دکمه محلی، مطابق منطق کنترلی تعریف‌شده و تنها در صورت فعال بودن دسترسی محلی، با موفقیت انجام شد.
- **تشخیص ورود غیرمجاز:** در سناریوهای غیرفعال بودن دسترسی محلی و فشردن دکمه، LED زرد چشمک‌زن و پیام هشدار در LCD فعال شد و پیام "ALARM" به Node.js ارسال گردید.
- **رفتار سیستم در شرایط بحرانی (Fail-Safe):** تمامی ورودی‌ها به جز فرمان بستن نادیده گرفته شدند و سروو در وضعیت ایمن باقی ماند.
- **تطبیق رفتار با منطق طراحی:** نتایج شبیه‌سازی دقیقاً مطابق جدول انتقال حالت‌ها و منطق مبتنی بر حالت (State-Based Design) بود.
- **کارایی لایه Node.js:** تمام پیام‌های سریال با صحت بررسی شدند و هیچ فرمان نامعتبر یا تکراری اجرا نشد، نشان‌دهنده‌ی قابلیت اطمینان و امنیت بالای سیستم بود.

برای تحلیل، داده‌ها به صورت جدول جمع‌آوری شدند:

جدول-۱ تحلیل نشان دادن وضعیت حالات سیستم

وضعیت فعلی سیستم	رویداد	وضعیت دسترسی	حالت بعدی سیستم	توضیح
درب بسته	فرمان باز از اپلیکیشن	مجاز	درب باز	باز شدن از راه دور
درب بسته	فشردن دکمه محلی	فعال	درب باز	باز شدن مجاز محلی
درب بسته	فشردن دکمه محلی	غیرفعال	هشدار	ورود غیر مجاز
درب باز	فرمان بستن از اپلیکیشن	-	درب بسته	بستن عادی درب
درب باز	فشردن دکمه محلی	غیرفعال	هشدار	ورود غیرمجاز درب باز
هشدار	دریافت فرمان بستن (C)	-	درب بسته	خروج از وضعیت بحرانی و پاک شدن آلارم
هشدار	ارسال فرمان باز (O) یا تغییر دسترسی (T)	-	هشدار	فرمان نادیده گرفته می‌شود، سیستم در حالت هشدار باقی می‌ماند

تحلیل داده‌ها نشان می‌دهد که سیستم شبیه‌سازی‌شده، رفتار کاملاً قابل پیش‌بینی، ایمن و منطبق با طراحی سیستم‌های واقعی دارد و می‌تواند پایه‌ای برای توسعه به سامانه‌های صنعتی و IoT باشد.

۵. بحث و نتیجه‌گیری

نتایج نشان می‌دهد سیستم شبیه‌سازی‌شده رفتار کاملاً پیش‌بینی‌شده و ایمن دارد و با استانداردهای سیستم‌های درب هوشمند واقعی هم‌راستا است. پایش وضعیت درب و کنترل محلی و از راه دور بدون خطا انجام شد، LED ها و سروو دقیقاً مطابق منطق طراحی پاسخ دادند و سیستم توانست ورود غیرمجاز را شناسایی و هشدار مناسب صادر کند. مقایسه یافته‌ها با پژوهش‌های مشابه در زمینه سیستم‌های کنترل دسترسی نشان می‌دهد طراحی مبتنی بر حالت باعث جلوگیری از رفتارهای نامعتبر و افزایش قابلیت عیب‌یابی شده است و لایه نرم‌افزاری Node.js نقش حیاتی در اعتبارسنجی داده‌ها و هماهنگی خروجی‌ها ایفا می‌کند.

با توجه به نتایج، طراحی مبتنی بر حالت و مدیریت دقیق ورودی‌ها و خروجی‌ها، پایداری و ایمنی سیستم را تضمین می‌کند و قابلیت تعمیم به پروژه‌های مشابه و صنعتی را دارد. پیشنهاد می‌شود در نسخه‌های صنعتی، لاگ‌ها و پیام‌های هشدار با پروتکل امن منتقل شوند و محدودیت نرخ برای ورودی‌های محلی اعمال گردد. این شبیه‌سازی نشان داد که سیستم می‌تواند به راحتی برای پروژه‌های اینترنت اشیا و کنترل دسترسی توسعه یابد و عملکرد آن با سامانه‌های واقعی قابل مقایسه است.

۶. قدردانی

مراتب قدردانی و سپاس خود را به جناب آقای دکتر الیاس عرب محقی به منظور راهنمایی‌های عالمانه و رهنمودهای سازنده ایشان در فرآیند نگارش این مقاله ابراز می‌دارم. بی‌تردید، نقش برجسته و حمایت‌های ارزشمند ایشان در ارتقای کیفیت این پژوهش، شایسته بالاترین تقدیرهاست

۷. منابع

1. Permana, K. A., Piarsa, I., & Wiranatha, A. A. K. A. (2024). *IoT-Based Smart Door Lock System with Fingerprint and Keypad Access*. Journal of Information Systems and Informatics, 6(3), 2086–2098.
2. Tailor, A., & Pandya, V. (2023). *IoT Based Smart Door Lock System*. IJRASET Journal for Research in Applied Science and Engineering Technology.

Logical Simulation of Smart Door System

Elyas arabmoheghi[1], mohammadreza shahani [2]
1- Department of Computer Engineering, Shahid Mohammad Montazeri University, Mashhad
elyas.arabmoh@gmail.com
2- Associate Student of Software, Shahid Mohammad Montazeri University, Mashhad
mohamadreza.shahani75@gmail.com

Abstract— Smart access control systems are essential for increasing security and remote monitoring. The aim of this research is to design and simulate a secure smart door system with a focus on robust control logic and threat detection. The methodology is based on practical implementation in a simulation environment with a three-layer architecture: the hardware layer (Arduino Uno) responsible for implementing state-based control logic and direct management of inputs/outputs, the middle layer (Node.js) as a secure interface for validation and sending commands, and the user layer (mobile application) for controlling and viewing the status. Key findings indicate the stability of the system resulting from the purposeful design of the pins and the modular architecture. The most important achievement is the successful implementation of the Alarm State, which is activated upon detecting local unauthorized access and places the system in a Lockdown state with Fail-Safe behavior; so that only the door closing command is accepted. The simulation results indicate the integrated and reliable operation of all layers in different scenarios. Finally, this simulated system, with a structure similar to industrial systems and independent security logic, has high potential for development into a real Internet of Things (IoT)-based system.

Keywords: Smart door, Arduino Uno, Internet of Things, Access control, Stateful logic, Embedded systems