KNO-1204-5002

پیاده‌سازی یک ماینر سبک‌وزن مبتنی بر ESP32 با استفاده از پلتفرم آموزشی Duino-Coin برای کاربردهای اینترنت اشیا

محمدحسین دانش‌پژوه[1]*، طالب صافی‌نیا[2]

[1]کارشناسی مهندسی کامپیوتر، دانشکده مهندسی جم، دانشگاه خلیج فارس ، استان بوشهر شهرستان جم
[2]استاد، دانشکده مهندسی جم، دانشگاه خلیج فارس ، استان بوشهر شهرستان جم
mohammadhdp@mehr.pgu.ac.ir

safiniataleb@gmail.com

چکیده

گسترش اینترنت اشیا(IoT) و نیاز به ارتقاء سطح دانش فنی کاربران در زمینه بلاک‌چین، وابستگی به دستگاه‌های کم‌مصرف با قابلیت پردازش لبه‌ای (Edge Computing) را افزایش داده است. از این رو این پژوهش به بررسی امکان پیاده‌سازی و ارزیابی عملکرد یک ماینر سبک‌وزن رمزارز بر پایه میکروکنترلر ESP32 در بستر شبکه Duino-Coin (DUCO) می‌پردازد. مسئله اصلی، سنجش توان عملی سخت‌افزارهای کم‌مصرف و دارای منابع محدود در اجرای وظایف محاسباتی مبتنی بر هش و تحلیل میزان کارایی آن‌ها در یک محیط آموزشی کنترل‌شده است. هدف مطالعه، نمایش عملی قابلیت‌های پردازشی ESP32، ارزیابی بهره‌گیری از پردازش موازی، و بررسی پایداری ارتباط شبکه‌ای در فرایند استخراج رمزارز است. در این راستا، یک ماینر مبتنی بر الگوریتم هش سبک DSHA1 با استفاده از سیستم‌عامل بلادرنگ FreeRTOS طراحی و پیاده‌سازی شد که از هر دو هسته پردازشی ESP32 برای افزایش نرخ هش استفاده می‌کند. همچنین، امکان پایش محلی وضعیت ماینینگ از طریق یک رابط وب تعبیه‌شده فراهم شد تا شاخص‌های عملکردی به‌صورت بلادرنگ قابل مشاهده باشند. نتایج حاصل از آزمایش‌های انجام‌شده نشان می‌دهد که ESP32 قادر است با پایداری مناسب در شبکه Duino-Coin مشارکت کرده و علی‌رغم محدودیت‌های سخت‌افزاری، به نرخ هش قابل قبولی دست یابد. استفاده از پردازش چندنخی منجر به بهبود محسوس عملکرد نسبت به اجرای تک‌هسته‌ای شده و توازن مناسبی میان توان مصرفی و بازده محاسباتی برقرار گردیده است. یافته‌های این مطالعه بیانگر آن است که اجرای وظایف مبتنی بر محاسبات رمزنگاری سبک بر روی سخت‌افزارهای اینترنت اشیا امکان‌پذیر بوده و می‌تواند به‌عنوان بستری مؤثر برای آموزش مفاهیم مرتبط با بلاک‌چین، پردازش لبه‌ای، مدیریت منابع و محدودیت‌های سیستم‌های توکار مورد استفاده قرار گیرد. این نتایج، ظرفیت ESP32 و پلتفرم‌های مشابه را برای کاربردهای پژوهشی و آموزشی در حوزه فناوری‌های نوین به‌طور عملی تأیید می‌کند.

کلمات کلیدی: اینترنت اشیا (IoT)، ESP32، Duino-Coin، ماینینگ سبک‌وزن، محاسبات لبه‌ای (Edge Computing).

۱ مقدمه

پایداری و توسعه شبکه‌های IOT وابسته به به‌کارگیری میکروکنترلرهایی نظیر ESP32 است که به دلیل هزینه پایین، مصرف انرژی کم و پشتیبانی داخلی از Wi-Fi، در کاربردهای مختلفی از خانه‌های هوشمند تا اتوماسیون صنعتی جایگاه ویژه‌ای دارند. در این راستا، هدف این پروژه نمایش ظرفیت محاسباتی ESP32 در انجام یک وظیفه سنگین، یعنی استخراج ارز دیجیتال، به شکلی سبک و آموزشی است.

برخلاف بلاک‌چین‌های بزرگ که نیازمند سخت‌افزارهای تخصصی (ASIC/GPU) هستند، استفاده از Duino-Coin این امکان را فراهم می‌آورد تا مفاهیم پیچیده مانند الگوریتم‌های هش، ارتباط کلاینت-سرور، و مدیریت منابع در یک پلتفرم کم‌مصرف شبیه‌سازی شوند. این رویکرد، Coin-Duinoرا به ابزاری کارآمد برای آموزش مفاهیم پیشرفته IoT و Edge Computing تبدیل می‌کند، جایی که دستگاه‌های نهایی مسئولیت بخشی از پردازش شبکه را بر عهده می‌گیرند.

شبکه Duino-Coin با هدف ایجاد یک زیرساخت بلاک‌چین در دسترس برای دستگاه‌های با منابع محدود (مانند آردوینو وESP) شکل گرفته است. این پروژه بر روی پلتفرم ESP32 متمرکز است که به دلیل داشتن Wi-Fi داخلی، قابلیت‌های محاسباتی دو هسته‌ای و پشتیبانی از سیستم‌عامل بلادرنگ(RTOS) ، بستر ایده‌آلی برای این منظور فراهم می‌کند. هدف اصلی، توسعه یک ابزار آموزشی و کاربردی برای درک فرآیند ماینینگ و تعامل با پروتکل‌های بلاک‌چینی بر روی سخت‌افزارهای IoT است[6-1].

۲ شرح کار و نتایج

در این بخش، فرایند انجام پژوهش از مرحله پیاده‌سازی تا ارزیابی عملکرد سیستم مورد بررسی قرار می‌گیرد. بدین منظور، ابتدا تجهیزات و بستر سخت‌افزاری و نرم‌افزاری مورد استفاده معرفی شده و سپس روش پیاده‌سازی ماینر سبک‌وزن بر روی میکروکنترلر ESP32 تشریح می‌شود. در ادامه، ساختار نرم‌افزاری سیستم، نحوه مدیریت پردازش‌ها و برقراری ارتباط با شبکه Duino-Coin بیان شده و مکانیزم پایش محلی عملکرد از طریق رابط کاربری وب توضیح داده می‌شود. در نهایت، نتایج حاصل از اجرای آزمایش‌ها ارائه و عملکرد سیستم از نظر پایداری، نرخ هش و کارایی پردازشی تحلیل می‌گردد.

۲-۱ معرفی Duino-Coin (DUCO)

Duino-Coin یک پروژه متن‌باز است که با هدف آموزش و سرگرمی برای دستگاه‌های کم‌مصرف طراحی شده است. این شبکه، برخلاف شبکه‌های مبتنی بر الگوریتم‌های سنگین مانند SHA-256 در بیت‌کوین، از الگوریتم‌های بسیار سبک‌تری استفاده می‌کند که امکان ماینینگ را بر روی پلتفرم‌هایی چون Arduino، ESP8266 و ESP32 میسر می‌سازد[7].

الگوریتم‌های مورد استفاده در DUCO (نظیر Duino-Hash که معمولاً ترکیبی از توابع ساده است) طوری طراحی شده‌اند که نیازمند توان محاسباتی بسیار پایین‌تری نسبت به الگوریتم‌های اثبات کار (PoW) استاندارد باشند. این امر باعث می‌شود که متوسط نرخ هش (Hash Rate) قابل دستیابی توسط ESP32 در مقیاس پایین‌تری قرار گیرد، اما هدف اصلی در DUCO کسب درآمد نیست، بلکه یادگیری عملی مفاهیمی چون الگوریتم‌های هشینگ، ارتباط Client-Server، و مدیریت منابع و بار کاری (Workload Synchronization)است. میزان سختی (Difficulty) در DUCO به گونه‌ای تنظیم می‌شود که حتی دستگاه‌های کم‌توان نیز بتوانند به صورت نظری شانس یافتن یک بلاک و کسب پاداش را داشته باشند[7,8,9].

۲-۲ برد ESP32

از میکروکنترلر ESP32 به عنوان هسته محاسباتی استفاده شده است. برد ESP32 به یک پردازنده دوهسته‌ای Tensilica LX6 با معماری ۳۲ بیتی و فرکانس کاری حداکثر ۲۴۰ مگاهرتز مجهز است و به‌صورت داخلی از قابلیت‌های ارتباطی Wi-Fi و Bluetooth پشتیبانی می‌کند. این برد از استانداردهای ارتباطی Wi-Fi 802.11 b/g/n بهره می‌برد و دارای حافظه SRAM محدود، معمولاً در حدود ۵۲۰ کیلوبایت در مدل‌های متداول، است. ترکیب قدرت پردازشی نسبتاً بالا، مصرف انرژی بهینه و دسترسی گسترده، ESP32 را در مقایسه با بردهایی نظیر Arduino Uno به گزینه‌ای مناسب برای پیاده‌سازی عملی مفاهیم شبکه‌ای و محاسباتی در پروژه‌های اینترنت اشیا تبدیل کرده است. با این حال، محدودیت در میزان حافظه و توان پردازشی همچنان به‌عنوان چالش اصلی این برد در اجرای طولانی‌مدت حلقه‌های محاسباتی و الگوریتم‌های رمزنگاری نسبتاً پیچیده مطرح می‌شود. مدیریت وظایف (Tasks) با استفاده از قابلیت‌های FreeRTOS در ESP32 صورت گرفته و هسته دوم (Core 1) به طور خاص برای اجرای عملیات‌های سنگین ماینینگ اختصاص داده شده است، در حالی که هسته اول (Core 0) مدیریت ارتباطات شبکه و وظایف سیستمی را بر عهده دارد. این تفکیک وظایف برای جلوگیری از تأخیر در عملیات‌های I/O و شبکه حیاتی است[7-11].

۲-۳ الگوریتم هش DSHA1

محاسبات هش از یک پیاده‌سازی اختصاصی به نام DSHA1 استفاده می‌کنند که عملیات اصلی آن بر اساس ساختار SHA-1 تعریف شده است. اگرچه این الگوریتم از لحاظ ساختاری شبیه به SHA-1 است، اما تفاوت‌های جزئی در نحوه مدیریت داده‌ها یا توابع کمکی ممکن است آن را برای منابع محدود بهینه‌تر سازد. این پیاده‌سازی شامل توابع transform و Round برای اجرای چرخه‌های محاسباتی است و از بهینه‌سازی‌های کامپایلری مانند __builtin_bswap32 برای تسریع تبدیل بایت‌ها استفاده می‌کند.

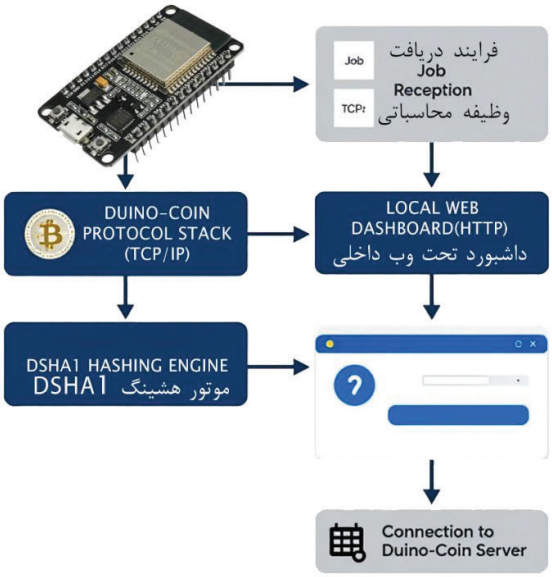
معادله کلی اثبات کار در این محیط به صورت زیر است:

$$H(BlockHeader \mid Counter) \leq Target \quad (۱)$$

در رابطه(1) ، H تابع هش مورد استفاده در فرایند ماینینگ است، $BlockHeader$ شامل اطلاعات بلاک از جمله داده‌های تراکنش‌ها و هش بلاک قبلی می‌باشد، $Counter$ مقدار نانس است که به‌صورت تکراری تغییر داده می‌شود و $Target$ بیانگر سطح سختی تعیین‌شده توسط شبکه است. در صورتی که مقدار هش محاسبه‌شده کوچکتر یا مساوی مقدار هدف باشد، بلاک تولیدشده معتبر تلقی می‌شود[12].

۲-۴ روش‌شناسی پیاده‌سازی

در فرایند پیاده‌سازی، ابتدا برای هر برد ESP32 یک شناسه منحصربه‌فرد تولید می‌شود که این شناسه بر اساس شناسه داخلی تراشه یا آدرس MAC دستگاه استخراج شده و به‌عنوان RIG_IDENTIFIER به شبکه Duino-Coin معرفی می‌گردد. این شناسه نقش کلیدی در شناسایی دستگاه، ردیابی سهم‌های ارسالی و انجام فرایند احراز هویت در استخر ماینینگ ایفا می‌کند.
مدیریت ارتباطات شبکه‌ای بر عهده کلاس MiningJob است که وظیفه برقراری ارتباط TCP با نودهای شبکه Duino-Coin را، معمولاً از طریق پورت استاندارد، بر عهده دارد. پس از برقراری اتصال، دستگاه با ارسال یک درخواست JOB که شامل اطلاعات کاربر و در صورت وجود، داده‌های حسگرهای محیطی نظیر دما و رطوبت است، تقاضای دریافت یک وظیفه جدید ماینینگ می‌کند. پروتکل ارتباطی مورد استفاده ساده و مبتنی بر تبادل داده‌های متنی است که سربار ارتباطی را کاهش داده و با محدودیت‌های سخت‌افزاری ESP32 سازگار است[7,8].



شکل ۱ – نمودار معماری سیستمی

پس از دریافت Job شامل هش بلاک قبلی و مقدار هدف، فرایند ماینینگ آغاز می‌شود. در این مرحله، یک شمارنده به‌صورت متوالی با داده‌های بلاک ترکیب شده و خروجی حاصل توسط الگوریتم هش DSHA1 محاسبه می‌گردد(شکل ۱). این عملیات تا زمانی ادامه می‌یابد که مقدار هش تولیدشده با شرایط تعیین‌شده توسط شبکه مطابقت داشته باشد. در صورتی که یک سهم معتبر شناسایی شود، به این معنا که مقدار هش محاسبه‌شده کوچکتر یا مساوی مقدار هدف باشد، سهم تأییدشده به همراه فراداده‌های مربوط به دستگاه، شامل نسخه نرم‌افزار و شناسه ریگ، به نود ارسال می‌شود. در نهایت، پاسخ سرور به‌صورت پیام‌های GOOD یا BAD دریافت شده و وضعیت پذیرش یا رد سهم توسط استخر ماینینگ مشخص می‌گردد[13-11].

۲-۵ داشبورد نظارتی تحت وب

به‌منظور ارائه بازخورد لحظه‌ای از وضعیت عملکرد سیستم و تسهیل فرایند عیب‌یابی در حین اجرای ماینینگ، یک صفحه وب ساده مبتنی بر HTML در حافظه فلش داخلی دستگاه ذخیره شده است. این صفحه با استفاده از ماکروهای PROGMEM و در قالب فایل Dashboard.h پیاده‌سازی شده تا ضمن کاهش مصرف حافظه SRAM، امکان بارگذاری سریع رابط کاربری فراهم شود. دسترسی به این داشبورد از طریق سرور وب داخلی ESP32 و معمولاً با بهره‌گیری از کتابخانه ESPAsyncWebServer انجام می‌گیرد که امکان پاسخ‌گویی غیرهمزمان و پایدار به درخواست‌های مرورگر را فراهم می‌کند.

داشبورد طراحی‌شده، مجموعه‌ای از شاخص‌های آماری حیاتی مرتبط با فرایند ماینینگ را به‌صورت متمرکز و قابل فهم نمایش می‌دهد. از جمله این شاخص‌ها می‌توان به نرخ هش اشاره کرد که به‌صورت لحظه‌ای و بر حسب کیلوهش بر ثانیه (kH/s) محاسبه و نمایش داده می‌شود و بیانگر توان پردازشی فعلی دستگاه است. همچنین، مقدار سختی مربوط به Job دریافتی از شبکه نمایش داده می‌شود که سطح پیچیدگی محاسباتی مورد انتظار را مشخص می‌کند. علاوه بر این، تعداد کل سهم‌های ارسال‌شده، شامل سهم‌های پذیرفته‌شده و ناموفق، به‌عنوان معیاری برای ارزیابی کیفیت ارتباط و کارایی ماینر در داشبورد ارائه می‌گردد.

در بخش اطلاعات سخت‌افزاری، مشخصاتی نظیر نسخه نرم‌افزار ماینر و میزان حافظه آزاد سیستم (Free Heap) نمایش داده می‌شود که این اطلاعات نقش مهمی در پایش سلامت سیستم و تشخیص مشکلات احتمالی ناشی از کمبود منابع دارند. داده‌های نمایش‌داده‌شده در داشبورد به‌صورت پویا از متغیرهای سراسری واکشی می‌شوند که توسط وظایف ماینینگ در سیستم‌عامل FreeRTOS به‌روزرسانی شده‌اند. این رویکرد امکان مشاهده وضعیت لحظه‌ای سیستم بدون ایجاد سربار محاسباتی قابل توجه را فراهم کرده و ابزار مناسبی برای تحلیل عملکرد و اشکال‌زدایی در زمان اجرا محسوب می‌شود[9,13].

۳ نتیجه‌گیری

در بخش نتیجه‌گیری باید نتایج مهم مقاله به طور مختصر ارایه گردند. حداکثر مطالب این بخش بین یک تا ۲ پاراگراف است.

در این پژوهش، امکان پیاده‌سازی و اجرای یک ماینر سبک‌وزن مبتنی بر میکروکنترلر ESP32 در بستر شبکه آموزشی Duino-Coin به‌صورت عملی مورد بررسی قرار گرفت. نتایج حاصل نشان داد که ESP32 با بهره‌گیری از معماری دوهسته‌ای و مدیریت وظایف مبتنی برFreeRTOS ، قادر است عملیات محاسباتی مورد نیاز برای الگوریتم هش DSHA1 را با پایداری مناسب و نرخ هش قابل قبول برای کاربردهای آموزشی انجام دهد. استفاده هدفمند از منابع حافظه، به‌ویژه به‌کارگیری PROGMEM برای ذخیره داده‌های ثابت و رابط کاربری وب، نقش مؤثری در کاهش مصرف RAM و افزایش پایداری سیستم در اجرای طولانی‌مدت داشته است. این یافته‌ها بیانگر آن است که با طراحی نرم‌افزاری مناسب، می‌توان محدودیت‌های ذاتی سخت‌افزارهای توکار را تا حد قابل توجهی مدیریت کرد.

به‌طور کلی، این پروژه نشان می‌دهد که پلتفرم‌های کم‌هزینه و کم‌مصرف اینترنت اشیا می‌توانند به‌عنوان بستری مؤثر برای آموزش عملی مفاهیم مرتبط با بلاک‌چین، محاسبات توزیع‌شده و مدیریت منابع مورد استفاده قرار گیرند. معماری پیشنهادی ضمن دستیابی به توازن مناسب میان کارایی پردازشی و مصرف منابع، قابلیت توسعه‌پذیری برای کاربردهای پژوهشی آینده را نیز داراست. نتایج این مطالعه می‌تواند مبنایی برای پژوهش‌های بعدی در زمینه بهینه‌سازی الگوریتم‌های سبک رمزنگاری، پردازش لبه‌ای و یکپارچه‌سازی سیستم‌های توکار با زیرساخت‌های نظارتی پیشرفته‌تر در حوزه اینترنت اشیا و بلاک‌چین فراهم آورد[14-16].

منابع

[1] S. Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System,” White Paper, 2008.

[2] M. Crosby, P. Pattanayak, S. Verma, and V. Kalyanaraman, “Blockchain Technology: Beyond Bitcoin,” *Applied Innovation Review*, no. 2, pp. 6–19, 2016.

[3] A. Dorri, S. S. Kanhere, and R. Jurdak, “Blockchain in Internet of Things: Challenges and Solutions,” *Computer Communications*, vol. 127, pp. 85–97, 2018.

[4] K. Christidis and M. Devetsikiotis, “Blockchains and Smart Contracts for the Internet of Things,” *IEEE Access*, vol. 4, pp. 2292–2303, 2016.

[5] M. Reyna, C. Martín, J. Chen, E. Soler, and M. Díaz, “On Blockchain and Its Integration with IoT: Challenges and Opportunities,” *Future Generation Computer Systems*, vol. 88, pp. 173–190, 2018.

[6] H. Hassan, M. Afifi, and A. E. M. Taha, “Energy-Efficient Blockchain for IoT Devices,” *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6392–6404, 2020.

[7] Espressif Systems, “ESP32 Technical Reference Manual,” Espressif Systems Inc., 2023.

[8] Espressif Systems, “ESP32-WROOM-32 Datasheet,” Espressif Systems Inc., 2023.

[9] R. Barry, *Mastering the FreeRTOS™ Real Time Kernel*, Real Time Engineers Ltd., 2016.

[10] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, “Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions,” *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[11] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, “Edge Computing: Vision and Challenges,” *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.

[12] A. Baliga, “Understanding Blockchain Consensus Models,” Persistent Systems Ltd., Technical Report, 2017.

[13] Duino-Coin Team, “Duino-Coin: A Lightweight Cryptocurrency for Educational Purposes,” Project Documentation, 2021.

[14] M. Conti, S. Kumar, C. Lal, and S. Ruj, “A Survey on Security and Privacy Issues of Bitcoin,” *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 3416–3452, 2018.

[15] N. Szabo, “Smart Contracts: Building Blocks for Digital Markets,” 1997.

[16] A. Bahga and V. Madisetti, *Internet of Things: A Hands-On Approach*, VPT, 2014.