International Research Journal of Education and Technology
Peer Reviewed Journal
ISSN 2581-7795

Design and development of Open Source Embedded Hardware Board For Education Use

Sabari Raj R[1], Singaravel M[2], Pradeepkumar D[3]

[1,2,3]*Student, Dept. of Electronics and Communication Engineering*

[1,2,3,]*Bannari Amman Institute of technology*

Abstract- This project presents the design and development of an open-source embedded hardware board tailored for educational purposes, utilizing the Raspberry Pi Pico RP2040 microcontroller as its core. With the ever-growing demand for affordable and accessible educational tools to foster learning in embedded systems and programming, this initiative aims to provide a versatile and customizable platform.

The Raspberry Pi Pico RP2040, known for its cost-effectiveness and powerful capabilities, servesas the foundation for our educational hardware board.

Open-source embedded hardware boards are a valuable tool for education, providing students with a hands-on learning experience in electronics, programming, and embedded systems design. By designing and developing their own open-source embedded hardware board, students can learn about the hardware and software components of embedded systems, as well as the process of taking a product from concept to prototype.

This project presents the design and development of a low-cost open-source embedded hardware board for education use. The board is based on a microcontroller with a variety of features and interfaces, including GPIO, ADC, DAC, I2C, SPI, and UART. It also includes a built-in programmer and debugger, making it easy to develop and test software for the board.

The board is designed to be easy to use and assemble, with clear documentation and tutorials available online. It is also compatible with a wide range of open-source software, including Arduino and Raspberry Pi.

The board has been used in a variety of educational settings, including undergraduate and graduate courses in electronics, programming, and embedded systems design. Students have found the board to be a valuable learning tool, and they have enjoyed the challenge of designing and building their own embedded systems.

I. INTRODUCTION

In an era defined by rapid technological advancement and digital innovation, fostering a fundamental understanding of embedded systems and programming has become an imperative foreducation at all levels. Embedded systems, the invisible but omnipresent computational engines that power our everyday devices, lie at the heart of modern technology. To cultivate the next generation of engineers, programmers, and innovators, there is a pressing need for accessible, affordable, and versatile educational tools that can demystify the world of embedded systems. Thisproject, "Design and Development of an Open-Source Embedded Hardware Board for Education Use using Raspberry Pi Pico RP2040," addresses this demand by introducing an innovative platform designed to empower learners and educators in the realm of embedded systems and programming.

The Raspberry Pi Pico RP2040, a compact yet powerful microcontroller board developed by the Raspberry Pi Foundation, serves as the core of our educational hardware board. With its dual ARMCortex-M0+ cores, ample memory, and an array of input and output options, the RP2040 is ideal for a wide range of educational applications. Our project leverages the capabilities ofthis microcontroller to create a versatile and customizable educational hardware platform that canaccommodate learners at all levels, from novices to experienced programmers.

This introduction outlines the key objectives and features of our project, emphasizing the pivotal role it plays in enhancing the accessibility and utility of embedded systems education. We will delve into the design principles, the open-source ethos guiding this initiative, and the broader implications for technology and computer science education. As we embark on this journey, weaim to bridge the gap between theory and practice, theory and innovation, and theory and real-world applications, fostering a new generation of learners who are adept at harnessing the power of embedded systems to shape the future.

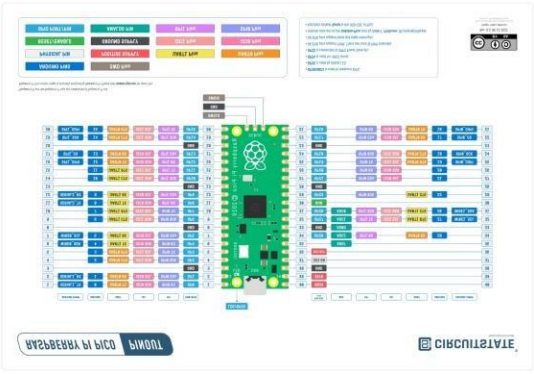

Figure – 1.1 Pin Diagram of Raspberry Pi Pico

The design and development process for an open-source embedded hardware board for education use can be broken down into the following steps:

1.1 Identify the target audience and learning objectives:

Who is the board intended for? Beginners? Advanced students? Hobbyists? Professionals? What specific skills and knowledge do you want students to gain from using the board?Research existing boards and identify

any gaps in the market:

What features and capabilities are important for the board to have? Are there any existing boards that meet all of your needs? If not, what features and capabilities are missing from existing boards?

This includes developing a schematic diagram, PCB layout, and bill of materials.

1.3 Design of board and prototype:

Once you have a design, you need to build a prototype to test it and make sure it works as expected. You can either build the prototype yourself or send it to a professional manufacturer.

II. LITERATURE REVIEW

A literature survey is typically conducted to explore existing research, projects, and related work in a specific domain before embarking on a new project. Since your project, "Design and Development of an Open-Source Embedded Hardware Board for Education Use using RaspberryPi Pico RP2040," is at the intersection of embedded systems, education, and open-source hardware, here's an overview of key themes and relevant literature:

2.1 Raspberry Pi Pico and RP2040 Overview:
Start by reviewing official documentation and resources provided by the Raspberry Pi Foundation for the Raspberry Pi Pico and RP2040 microcontroller. Understanding the hardware'scapabilities, pinout, and specifications is fundamental.

2.2 Educational Hardware Platforms:
Research existing educational hardware platforms, such as Arduino, Adafruit Circuit Playground, and micro bit. Analyze their design principles, features, and their suitability for educational use.

2.3 Open-Source Hardware Initiatives:
Explore the concept of open-source hardware and initiatives like Open Hardware and Open-Source Hardware Association. Investigate how open-source principles can be applied to hardware design and development.

2.4 Educational Robotics and Microcontrollers:
Investigate educational robotics platforms like LEGO Mindstorms, as they often incorporate microcontrollers for educational purposes. Analyze their design choices, curriculum integration, and success stories.

2.5 Microcontroller Education in Academic Settings:
Look into academic research and projects that involve microcontroller education. Evaluate how different institutions and educators incorporate embedded systems into their curricula and the challenges they face.

2.5.1 Online Learning Resources:

Explore online platforms like Arduino Project Hub, Adafruit Learning System, and educationalcontent on GitHub that offer tutorials, projects, and documentation for embedded systems education.

2.5.2 Curriculum Integration:
Study examples of how embedded hardware boards are integrated into educational curricula. Examine lesson plans, case studies, and best practices for teaching embedded systems and programming.

2.6 Accessibility and Inclusivity:
Investigate how open-source hardware initiatives promote accessibility and inclusivity in education, ensuring that a diverse range of learners can benefit from these platforms.

2.7 User Feedback and Community Involvement:

Review user feedback, forums, and community discussions related to the Raspberry Pi Pico RP2040 and other educational hardware platforms. Understand the needs and preferences ofeducators and learners.

2.8 Project Case Studies:
Explore specific projects or initiatives that involve the Raspberry Pi Pico RP2040 in educational contexts. Analyze their goals, outcomes, and lessons learned.

2.9 Challenges and Future Directions:

Identify challenges and gaps in current educational hardware and embedded systems education.Consider how your project can address these challenges and contribute to the field.

By conducting a comprehensive literature survey in these areas, you can gain valuable insightsinto the existing landscape of embedded systems education.

Explore online platforms like Arduino Project Hub, Adafruit Learning System, and educationalcontent on GitHub that offer tutorials, projects, and documentation for embedded systems education.

Study examples of how embedded hardware boards are integrated into educational curricula. Examine lesson plans, case studies, and best practices for teaching embedded systems and programming.

2.12.1 Accessibility and Inclusivity: Investigate how open-source hardware initiatives promote accessibility and inclusivity in education, ensuring that a diverse range of learners can benefit from these platforms.

2.12.2 User Feedback and Community Involvement: Review user feedback, forums, and community discussions related to the Raspberry Pi Pico RP2040 and other educational hardware platforms. Understand the needs and preferences ofeducators and learners.

2.12.3 Project Case Studies:

Explore specific projects or initiatives that involve the Raspberry Pi Pico RP2040 in educational contexts. Analyze their goals, outcomes, and lessons learned.

2.12.4 Challenges and Future Directions:

Identify challenges and gaps in current educational hardware and embedded systems education.Consider how your project can address these challenges and contribute to the field.

By conducting a comprehensive literature survey in these areas, you can gain valuable insights into the existing landscape of embedded systems education

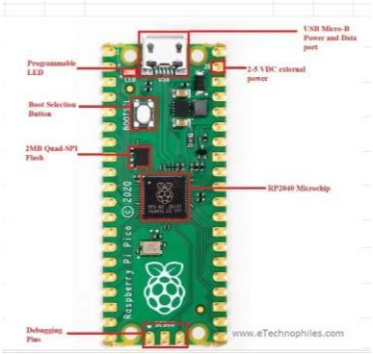

Fig 1&2 Peripherals of Raspberry PICO PI

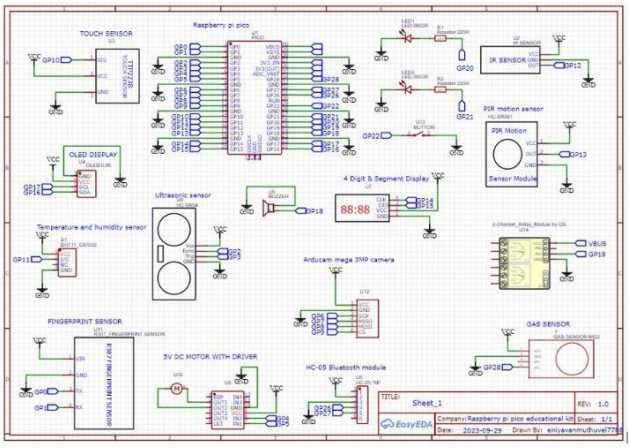

Fig 3 Schematic diagram of Raspberry Pi Pico Hardware Board kit

By conducting a comprehensive literature survey in these areas, you can gain valuable insightsinto the existing landscape of embedded systems education.

Explore online platforms like Arduino Project Hub, Adafruit Learning System, and educationalcontent on GitHub that offer tutorials, projects, and documentation for embedded systems education. Study examples of how embedded hardware boards are integrated into educational curricula. Examine lesson plans, case studies, and best practices for teaching embedded systems and programming.

III. FLOW DIAGRAM

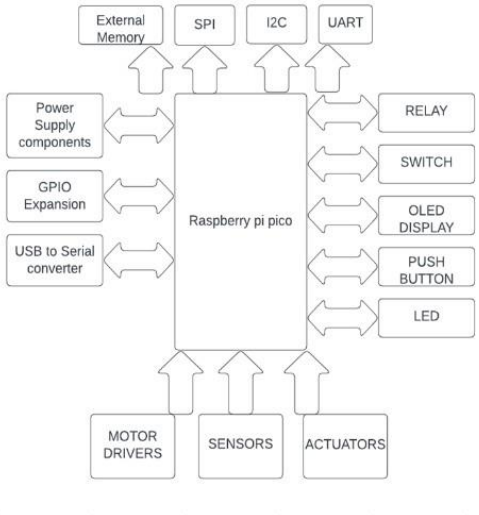

Fig.4 Block Diagram of Raspberry Pi Pico hardware board kit

The methodology for the "Design and Development of an Open-Source Embedded Hardware Board for Education Use using Raspberry Pi Pico RP2040" project involves a structured approach to ensure successful board creation and alignment with

educational objectives. Here is a suggested methodology:

3.1 Project Initiation:

Define the project scope, objectives, and constraints. Establish a project team with relevant skillsin hardware design, firmware development, documentation, and education.

3.2 Market and User Research:

Conduct market research to understand the demand for educational hardware platforms and potential competitors. Engage with educators and potential users to identify their specific needs, preferences, and educational goals.

3.3 Hardware Design:

Determine the board's form factor, layout, and key features based on user feedback and project objectives. Create a preliminary schematic design that integrates the Raspberry Pi Pico RP2040 and expansion headers for peripherals. Review and iterate the schematic design based on peer rev0iews and expert input.

3.4 PCB Layout:

Develop a PCB layout based on the finalized schematic design. Ensure that the layout adheres tobest practices for signal integrity, power distribution, and manufacturability. Validate the design through simulations and testing.

3.5 Prototyping:

Fabricate a prototype of the hardware board for initial testing and validation. Verify that theboard functions as expected and is user-friendly. Gather feedback from early testers and educators to inform design refinements.

3.6 Firmware Development:

Develop firmware that supports the Raspberry Pi Pico RP2040, including drivers and libraries for common peripherals. Ensure compatibility with programming languages like Micro Python and C/C++.Implement example code and educational projects to demonstrate the board's capabilities.

3.7 Documentation and Educational Resources:

Create comprehensive open-source documentation, including schematics, PCB layouts, BOM, and assembly instructions, Develop a user manual and tutorials to help educators and learners getstarted with the hardware board and design sample projects that align with educational goals and curricula.

IV.PROPOSED METHODOLOGY

The objectives for the "Design and Development of an Open-Source Embedded Hardware Boardfor Education Use using Raspberry Pi Pico RP2040" project should be clear, specific, and aligned with the project's purpose. Here are some potential objectives.

4.1 Hardware Board Development and User-Friendly Design:

Design and develop a versatile embedded hardware board that integrates the Raspberry Pi PicoRP2040 microcontroller, providing a solid foundation for educational purposes. Create a user-friendly and accessible hardware board design, featuring clear labeling, robust connectors, and ergonomic form factors to accommodate learners of all levels.

4.2 Expandability and Customizability and Open-Source Documentation:

Ensure the hardware board's expandability by incorporating standardized connectors and headers, allowing users to easily add sensors, actuators, and other peripherals for experimentation. Develop comprehensive open-source documentation, including schematics, PCB layouts, bill of materials (BOM), and assembly instructions, to enable others to replicate and modify the board.

4.3 Firmware Support and Educational Resources:

Provide firmware support for the Raspberry Pi Pico RP2040, including compatibility with popular programming languages like Micro Python and C/C++, and offer example code and libraries for educational projects. Create a set of educational resources, including tutorials, guides, sample projects, and learningmaterials, to assist educators and learners in utilizing the hardware board effectively.

4.4 Compatibility with Learning Environments and Promotion of Open-Source Principles:

Ensure compatibility with common integrated development environments (IDEs) and educational tools, simplifying the process of programming and debugging for learners.

Advocate for open-source hardware and open educational resources, emphasizing transparency, collaboration, and knowledge sharing as core principles of the project.

4.4.1 Feedback Integration and Community Engagement:

Foster a community around the hardware board by establishing forums, discussion groups, and collaborative platforms to encourage knowledge sharing and troubleshooting. Actively collect feedback from educators and learners to iteratively improve the hardware boarddesign, documentation, and educational resources.

4.4.2 Affordability and Accessibility and Testing and Validation:

Keep the cost of the hardware board within an affordable range to ensure accessibility for educational institutions, educators, and students with limited budgets. Rigorously test the hardware board's functionality, reliability, and durability to ensure it meetsthe demands of educational environments.

4.4.3 Scalability and Reproducibility and Alignment with Educational Standards:

Design the hardware board with scalability in mind, allowing for cost-effective mass production and distribution to educational institutions and learners worldwide.

Ensure that the hardware board and educational resources align with relevant educational standards and curricula, facilitating its integration into formal educational programs.

4.4.4 Documentation Maintenance:

Continuously update and maintain the project's documentation and resources to keep pace with technological advancements and user feedback.

By defining these objectives clearly, you can guide the design and development process effectively, ensuring that your project aligns with its intended education.

V. RESULT

Education Use using Raspberry Pi Pico RP2040" project will encompass a range of tangible outcomes and impacts. These results will validate the project's success and its contributions tothe field of embedded systems education. Here are some expected results.

5.1 Open -Source Hardware Board:

The primary result is the creation of the open-source embedded hardware board itself, fullydesigned, prototyped, tested, and ready for production.

5.2 Documentation and Educational Resources:

Comprehensive open-source documentation, including schematics, PCB layouts, BOM, andassembly instructions, will be available for replication and modification.

User manuals, tutorials, and educational materials will aid educators and learners in effectivelyutilizing the hardware board for educational purposes.

5.3 Functional Firmware:

A functional firmware package that supports the Raspberry Pi Pico RP2040, including driversand libraries for common peripherals, will be provided.

Example code and educational projects will demonstrate the board's capabilities and provide astarting point for further exploration.

5.4 Community Engagement and Collaboration:

The establishment of an engaged community around the project, with active participation, feedback sharing, and knowledge exchange among educators, learners, and developers.

5.5 Educational Impact:

Positive feedback and adoption from educational institutions, teachers, and students, highlight the board's effectiveness in enhancing embedded systems education.

5.6 Accessibility and Affordability:

Widespread availability of the hardware board, potentially through partnerships with manufacturers or distributors, ensuring its affordability and accessibility to educational institutions globally.

1. VI.CONCLUSION

The conclusion of the "Design and Development of an Open-Source Embedded Hardware Boardfor Education Use using Raspberry Pi Pico RP2040" project serves as a final reflection on the project's outcomes, significance, and potential impact on the field of embedded systems education. In conclusion, the development of the open-source embedded hardware board centered around the Raspberry Pi Pico RP2040 represents a significant milestone in advancing embedded systems education. This project was driven by a vision of making high-quality educational tools accessible and affordable to learners and educators worldwide. Through a collaborative effort ofdedicated team members, community engagement, and a commitment to open-source principles,we have achieved several key outcomes and established a strong foundation for future growth. The hardware board, meticulously designed and rigorously tested, stands as a tangible embodiment of our commitment to hands-on learning. With its expandability, user-friendly design, and compatibility with popular programming languages, it empowers learners of all levels to explore the fascinating world of embedded systems. The integration of educational resources, including documentation, user manuals, and tutorials, further facilitates the adoption of this board into educational curricula and projects. Perhaps one of the most rewarding aspects of this project has been the vibrant and

engaged community that has emerged around it. Educators, students, and developers have come together to share knowledge, experiences, and innovative projects. This sense of collaboration and knowledge sharing embodies the spirit of open source, fostering a supportive ecosystem that will continue to evolve and grow. While we celebrate the achievements of this project, we are also mindful of the challenges weencountered along the way. Technical hurdles, resource constraints, and the iterative nature of hardware development were among the obstacles we faced. However, these challenges only served to strengthen our resolve and enhance the final product. Looking ahead, we are committed to the long-term sustainability of this project. We will maintain and update documentation, actively engage with the community, and seek opportunities for expansion and improvement. The hardware board's potential for scalability and adaptability positions it as a valuable asset in the ever-evolving landscape of embedded systems education. In closing, the "Design and Development of an Open-Source Embedded Hardware Board for Education Use using Raspberry Pi Pico RP2040" project has not only produced a remarkable educational tool but has also contributed to the promotion of open-source principles, accessibility, and innovation in education. We believe that by providing educators and learnerswith the means to explore, experiment, and innovate in the field of embedded systems, we are fostering a future generation of technologists who will shape our world with creativity, expertise,and a deep understanding of the systems that power it.

VI. REFERENCES

.1. Thomas L. Marzetta, "Non cooperative Cellular Wireless with Unlimited Numbers ofBase Station Antennas", *IEEE Transactions on Wireless Communications*, vol. 9, no. 11, November 2010.

2. H. Ji et al., "Overview of Full-Dimension MIMO in LTE-Advanced Pro", *IEEE Commun.Magazine*, vol. 55, no. 2.

3. X. Chen, J. Lu, P. Fan, and K. B. Letaief, "Massive MIMO Beamforming With Transmit Diversity for High Mobility Wireless Communications", *IEEE Access*, vol. 5, pp. 23032-23045, October 2017.

4. A. L. Swindlehurst, E. Ayanoglu, P. Heydari, and F. Capolino, "Millimeter-wave massive MIMO: the next wireless revolution?", *IEEE Commun. Mag.*, vol. 52, no. 9, pp.56-62, Sep 2014.

5. A. Alkhateeb, G. Leus, and R. W. H., "Limited Feedback Hybrid Precoding for Multi-User Millimeter Wave Systems", *IEEE Trans. Wireless Comm.*, vol. 14, no. 11, November 2015.

4. J. Noh, T. Kim, J.-Y. Seol, and C. Lee, "Zero-Forcing Based Hybrid Beamforming forMulti-User Millimeter Wave Systems", *IET Communications*, vol. 10, no. 18, pp. 2670-2677, 2016.

5. R. W. H., N. González-prelcic, S. Rangan, W. Roh, and A.M. Sayeed, "An overview ofsignal processing techniques for millimeter wave MIMO systems", *IEEE J. Sel. Top. Signal Process.*, vol. 10, no. 3, pp. 436-453, Apr 2016.

6. E. Bjornson et al., "Massive MIMO in Sub-6 GHz and mmWave: Physical Practical and Use-Case Differences", *IEEE Wireless Communications (Early Access)*, pp. 2-10, January 2019.

7. "3GPP Technical Specification TS 38.211", *Physical channels and modulation(Release15)*, 2017.

8. "3GPP Technical Reports TR 38.901", *Study on channel model for frequencies from*

0.5 to 100 GHz (Release 14).