The Impact Of Raspberry Pi Pico On Iot-Based Automation Systems: A Review

Amruta Sanap
Electronics and telecommunication
Vishwakarma institute of information technology
Pune,India

Dr. Pravin Gawande
Electronics and telecommunication
Vishwakarma institute of information technology
Pune,India

***Abstract*—The Internet of Things (IoT) revolution has changed automation in various sectors, from home equipment to factory processes, agriculture, and health. The backbone of this revolution is the need for low-cost, reliable, and programmable microcontrollers. The Raspberry Pi Pico, which is powered by the RP2040 microcontroller, offers a compact, low-power, and affordable board with multi-purpose I/O and dual-core capability, which is of keen interest to IoT-based automation systems. This review paper deals with the technical aspects, architectural features, and programming environments of the Raspberry Pi Pico from the IoT perspective. It focuses on its compatibility with sensors, actuators, and communication modules and evaluates its applications in real-time data processing, wireless control, and edge computing. Through case study comparisons as well as other microcontroller platforms to Pico, this paper provides a detailed insight into its contribution, advantage, and drawback in automating smart IoT systems. The paper concludes with research potential and future directions for Raspberry Pi Pico in the forthcoming generation of smart automation.**

***Index Terms*—keywords-Raspberry Pi Pico,RP2040 Microcontroller,Internet of Things (IoT),IoT Automation**

I. Introduction

The Internet of Things (IoT) is transforming automation by linking sensors, devices, and systems for real-time monitoring and control. To apply these systems, miniaturized, energy-saving, and programmable microcontrollers are necessary. The Raspberry Pi Pico, developed by the Raspberry Pi Foundation, is a low-cost, high-performance microcontroller powered by the RP2040 chip. With characteristics such as dual-core processing, adaptable I/O, and MicroPython and C/C++ support, it has gained a favorable position among IoT-based automation projects. This article delves into the application of Raspberry Pi Pico in IoT systems and discusses its architecture, development tools, applications, benefits, drawbacks, and potential in creating intelligent, automated solutions.The Raspberry Pi Pico is an affordable, high-performance microcontroller board intended for embedded and real-time control systems. In contrast to earlier Raspberry Pi boards, which are single- board Linux-based computers (SBCs), the Pico is centered on a specially designed microcontroller chip called the RP2040 and is dedicatedly designed for microcontroller and bare-metal programming. Its architecture focuses on flexibility, cost-effectiveness, and simplicity of integration into diverse automation and IoT-based applications.

II. RELATED WORK

Recent technological advancements in the field of microcontrollers have greatly sped up the evolution of intelligent automation systems, especially in the field of Internet of Things (IoT). Researchers and developers are more and more utilizing platforms such as Raspberry Pi Pico, Arduino, and ESP32 to develop low-cost, efficient solutions for industrial and home automation. This section provides a review of the literature with special emphasis on the application of Raspberry Pi and related platforms in voice-controlled and IoT-based automation systems. A research by M. Ramakrishna et al. (2025) suggested voice-operated home automation based on Raspberry Pi Pico and HC-05 Bluetooth module to offer easy control of house- hold appliances for the elderly and physically challenged. The system translates voice commands sent from a smartphone and executes them through the Rasp- berry Pi Pico to drive relays for controlling electrical loads. Programming was done with Python and MicroPython to show the board's flexibility and simplicity for use in real- life situations. Their prototype, which they developed based on a breadboard system, verified the viability of developing affordable and customizable home automation systems with Pico .[1].

Numerous studies in the IoT and embedded systems domain have explored the integration of microcontrollers such as the Raspberry Pi Pico for automation and surveillance applica- tions. This section summarizes previous research efforts that support the development of IoT-based automation systems using the Raspberry Pi family, highlighting especially the flexibility and cost-effectiveness of the Pico platform. Real- time monitoring of the environment is now a critical feature in intelligent disaster management systems, especially in flood- hit areas. An IoT-based flood alert and monitoring system with Raspberry Pi Pico has been proposed by Bramhadev B. Rupanar, Vrushal S. Shilawant, Akansha R. Pawar, and Prof. M. M. Zade (2025) in a latest study. They employ several sensors—e.g., ultrasonic water level sensors, rain sensors, temperature and humidity sensors using DHT11, and soil sensors—to read holistic environmental data. The Raspberry Pi Pico in combination with an ESP8266 Wi-Fi board handles sensor input and sends out notifications through mobile alerts and a local buzzer as thresholds are reached. The system also stores real-time data on the cloud to access remotely and analyze historically. This effort showcases the Pico's ability in multi-sensor data processing, low-power computing, and IoT-based decision support, substantiating its application in smart automation and natural disaster early warning systems.[2].

New advances in microcontroller technology, particularly among the RP2040 series employed in Raspberry Pi Pico and Pico 2, have brought new opportunities for scientific and industrial automation precision control. Such platforms are being increasingly utilized with MicroPython-based programming by researchers to design low-cost and high-performance solutions for coordinated actuation and control of experiments. In an experiment by S. H. M. Mehr (2025), a specialized language called "CtrlAer" was created to execute on Raspberry Pi Pico/Pico 2 for deterministic, real-time execution of parallel actuation signals. The system exploits the RP2040/RP2350's programmable I/O (PIO) to generate tightly synchronized, gap-free digital outputs on multiple channels. By merging MicroPython's flexibility with PIO's hardware-level accuracy, CtrlAer was able to show the ability to drive devices like stepper motors, piezoelectric actuators, and solenoid valves in laboratory automation applications. The research highlights how such low-cost, reprogrammable solutions can greatly decrease complexity over FPGA- or SBC-based systems, making them desirable for applications that need high-level precision timing. The research also points out possible enhancements, including the incorporation of real-time triggering and more adaptive duty-cycle generation, to enhance the system's functionality further in IoT-based automation scenarios [3].

Apart from application-oriented automation, scientists have also developed the potential of Raspberry Pi Pico in engineering education and computer system design. Davidson, Z. C. M., Dang, S., and Vasilakos, X. (2024) presented a hybrid laboratory approach with Raspberry Pi Pico to deliver digital circuits and systems courses that responded to the increasing demand for low-cost and remote-access learning tools within the field of electronics education. Their article in IEEE Transactions on Learning Technologies highlights the Pico's ability to model and execute digital logic in real time, allowing students to cross from theory to experimentation in physical or virtual labs. Authors showed how Pico's GPIO, PWM, and interrupt handling capabilities could facilitate basic concepts such as flip-flops, counters, and serial comms. The research highlights the worth of Pico as both a hardware control platform and a cost-effective and scalable learning platform in the field of digital electronics and embedded systems, playing an indirect role in the training of capable professionals in the field of IoT automation [4].

New developments in open-source hardware and photodetector technology have made it possible to create low-cost, compact gamma-ray spectrometers for academic, research, and field use. Bonifacio et al. (2024) introduced the OpenGamma spectrometer, a small and economical device constructed on the Raspberry Pi Pico microcontroller, using a GAGG scintillator crystal in combination with a Silicon Photomultiplier (SiPM) as a gamma-ray detector. The system exhibited an energy resolution of 7.6% at 662 keV, on par with other low-cost spectrometers, and a dead time of 20.5 ± 1.5 μs, with the stability observed at lower pulse frequencies. The hardware used a low-noise analog front-end, peak detector, and real-time optimized firmware. An open-source web application was also made available for spectrum analysis and data management. Case studies demonstrated the spectrometer's utility in quantification of standard radioactive sources for calibrating and monitoring the environment. Although efficient, the study found some issues like the temperature sensitivity of the SiPM and limited high-precision performance, with future prospects for improving resolution and environmental hardness for wider deployment[5].

Kasiselvanathan et al. (2023) developed a wireless surveillance robot using Raspberry Pi Pico to address the limitations of traditional CCTV systems, such as poor image quality and high storage requirements . Their system integrates motion detection algorithms with a live video streaming capability, offering a compact and mobile surveillance solution controllable via Bluetooth. The design demonstrates how Pico, combined with mobile technology, can serve as a central controller in real-time image processing and motion tracking applications. The use of MicroPython and integration with a web camera showcases Pico's flexibility for surveillance and smart monitoring tasks.[5].

The combination of industrial automation with IoT has emerged as a revolutionary force in real-time danger detection systems. In research work by Kondeti Chirunadh, G. Sai Varaprasad Goud, G. Tejaswini, and B. Srinivasa Rao (2023), an exhaustive industrial monitoring and warning system was developed through the application of Raspberry Pi Pico, NodeMCU, and GSM technology. The system involves the use of a sensors' network—flame, gas, fire, LM35, and PIR sensors—to monitor unusual environmental conditions in an industrial environment. The data from the sensors is processed by the NodeMCU and Pico and communicated through GSM for SMS notifications and sent to the ThingSpeak IoT platform for cloud-based real-time monitoring. The system exemplifies a multi-sensor, scalable method to detecting hazards in the industrial environment, with a focus on modularity, real-time response, and IoT-based data analysis. This book indicates Raspberry Pi Pico's functionality in supporting smart surveillance and predictive maintenance within Industry 4.0 settings.[6]. Broadening the educational horizon of IoT applications, David R. Loker (2023) investigated using the Raspberry Pi Pico as an underlying platform for embedded systems and IoT projects in engineering and technology education. The article explained how the RP2040 microcontroller on the Pico board provides flexible I/O capabilities like ADCs, UART, SPI, I2C, and PWM that can be used for both digital and analog interfacing. Loker illustrated the application of MicroPython and the Thonny IDE for ease of embedded programming, and its demonstration using real-world examples such as LCD interfacing, DAC interfacing, accelerometer interfacing, and ESP8266 WiFi module interfacing for TCP-based IoT applications. The project further highlighted how these projects help achieve ABET student learning outcomes, with hands-on assignments like plant monitoring system and speed monitoring of vehicles. This research is in favor of the argument that Pico not only fulfills industrial IoT requirements but is also an affordable educational platform for creating intelligent automation systems.[7].

Smart irrigation and precision agriculture have gained significant attention with the growing demands for optimal water management. Dr. G. Dhanalakshmi, Bharathwaj Dasoju, Dr. G. Vijayabhasker, and Dr. G. Srinivas (2023) created an IoT-based agriculture monitoring system based on Raspberry Pi Pico and wireless sensor networks. The system employs DHT-11 and soil moisture sensors for tracking environmental conditions such as temperature, humidity, and soil wetness. It controls pumps (AC/DC) autonomously based on sensor inputs to water the field and also sends real-time reports over GSM to the user's mobile phone. The system also has an ESP32-CAM module for live video monitoring remotely. Raspberry Pi Pico was used over Pi 3/4 due to its lower cost and adequate GPIO functionality needed for agriculture automation. This research supports Pico's capability of providing affordable, sensor-based precision agriculture with real-time data capture, control, and IoT integration.[8].

The use of Raspberry Pi Pico within embedded systems and IoT learning has been investigated as an approach to support hands-on learning in university courses. Dimitra Doropoulou et al. (2023) introduced a systematic course for undergraduate Computer Science students on MicroPython programming for Raspberry Pi Pico and Pico W boards. The course integrated physical computing, project-based learning, and IoT application programming, spanning ARM Cortex-M architecture and analog interfacing to RP2040 Programmable I/O (PIO) and cloud data aggregation through ThingSpeak. Hands-on lab sessions covered activities such as the construction of an IoT-based weather station, analog measurement with PWM control implementation, and a semester-long project in the shape of a low-cost line follower robot. The research indicated that students with little prior knowledge of embedded systems in the beginning, developed their skills and confidence considerably in MicroPython, IoT topics, and hardware interfacing by the end of the course. The results indicate that systemic, interactive lab sessions involving Raspberry Pi Pico can successfully close theoretical learning and practical embedded system implementation gaps in tertiary education [9].

Home automation technologies have advanced significantly with the adoption of IoT-enabled microcontrollers, enabling seamless control of household appliances via smartphones and wireless modules. A study by M. G. Sumithra, B. Banupriya, and R. Mathankumar (2023) proposed a Raspberry Pi Pico–based smart home automation system integrating the Blynk IoT platform for remote operation. The system utilized Pico's GPIOs to interface with relays for controlling lights and fans, while the Blynk application provided a user-friendly mobile interface. Communication between the Pico and the cloud platform was established through an ESP8266 Wi-Fi module, enabling real-time device control and status monitoring from anywhere with internet access. The design emphasized affordability, modularity, and ease of configuration, allowing users to expand the system with additional sensors and actuators. Prototype testing demonstrated reliable switching performance with minimal latency, highlighting the potential of Pico-based solutions in creating scalable, cost-effective IoT smart home systems [10].

The Raspberry Pi Pico has emerged as an accessible and versatile microcontroller platform for both educational and practical automation applications. In a study by Ngoc Doan Duong (2023), the RP2040-based Raspberry Pi Pico was introduced to university students as a tool for learning electronics, automation, and IoT. The work outlined the structure and capabilities of the Pico and Pico W boards, including GPIO interfacing, built-in ADC, temperature sensing, and Wi-Fi connectivity in the Pico W variant. A series of practical demonstrations showcased its versatility: automatic light control using a photoresistor and ADC, real-time clock and temperature display on an LCD, text and image display on a TFT screen, and internet-based device control through the Anvil platform. These examples illustrated the Pico's ability to integrate with various communication protocols (I2C, SPI, UART) and support both MicroPython and C/C++ programming. The findings suggest that Raspberry Pi Pico offers a low-cost, engaging platform for students in electronics, mechatronics, and automation to gain hands-on skills applicable in academic projects, graduation theses, and professional applications [11].

Innovative uses of Raspberry Pi Pico in interactive art and embedded systems have proven the platform viable for sensor-based, responsive designs. John Crabtree (2023) created an interactive picture frame based on the University of Georgia Bulldogs football stadium tradition where cell phone flashlights illuminate the stands at the fourth quarter. The system utilized a Raspberry Pi Pico H microcontroller that had been coded in MicroPython using the Thonny IDE, taking input from photoresistors and providing output in the form of red and white LEDs in circular arrangement around a stadium photo. The LEDs' brightness changed based on light intensity as sensed by the photoresistors, thus enabling user interaction through a phone flashlight. Clock-based scheduling governed the "wave" illumination pattern, with each LED linked to its own GPIO pin for seamless sequencing. Using one photoresistor–LED pair per circuit ensured greater reliability and beauty. Although wireless connectivity was not included in the current design, physical protection helped counter security threats. The project emphasizes the Pico's ability in low-power real-time interactive installations and suggests improvements for future projects like buzzer support, light strips, and object-oriented programming for code optimization [12].

Current breakthroughs in embedded computing have made it easier to implement low-cost microcontrollers such as the Raspberry Pi Pico in Internet of Things (IoT) use cases. Kumar and Kumar (2023) showcased the Pico as a capable platform for building scalable IoT applications for home automation, environmental monitoring, and industrial control. The research highlighted Pico's native MicroPython support, low power usage, and dual-core ARM Cortex-M0+ processor, allowing for ease of integration with wireless communication modules and rapid prototyping. Various case studies such as sensor automation and data logging systems demonstrated how the Pico can be used to interface with cloud platforms for control and real-time monitoring. The authors further emphasized the use of IoT-enabled Pico projects in educational environments to offer experiential learning opportunities for students to connect theory with practice in the area of embedded systems. Some of the issues like network security, interoperability, and long-term reliability were noted as major points of consideration for future deployments [13].

In an extensive review of Raspberry Pi Pico–based experimental control platforms, Di Lorenzo et al. (2022) demonstrated a hardware and software platform to produce accurate, reproducible timing signals for lab experiments. Taking advantage of the RP2040 microcontroller's programmable I/O (PIO) subsystem, the devised solution supported sub-microsecond accuracy in multi-channel digital signal generation, appropriate for controlling devices like actuators, sensors, and data acquisition units. The authors used a modular MicroPython library to program and run synchronized output sequences, allowing researchers to substitute expensive FPGA or function generator configurations with a low-profile, low-cost option. Demonstration experiments in neuroscience and optics confirmed the performance of the system, demonstrating the ability of the Pico to drive hardware with deterministic timing demands reliably. The authors concluded that such cost-effective, open-source control solutions can democratize high-precision experimental configurations in low-resource research settings [14].

As lifestyle diseases become increasingly of concern, low-cost microcontroller-based health monitoring systems have become popular solutions for ongoing fitness tracking. A paper by Jagan Balaji Thiyagarajan and Madhavan Thothadri (2021) suggested a Raspberry Pi Pico-based and MAX30100 pulse-oximeter module-based fitness monitoring system to calculate and store eight vital health parameters: Beats Per Minute (BPM), Oxygen Saturation (SpO_2), Relative Fat Mass (RFM), Maximum Heart Rate (MHR), Heart Rate Reserve (HRR), Target Heart Rate (THR), Resting Metabolic Rate (RMR), and Daily Energy Expenditure (DEE). The system takes simple user inputs like age, gender, height, weight, waist circumference, and activity level, and processes sensor values in real time using MicroPython programming. One of the remarkable aspects of the design is its on-board data logging feature, thereby obviating the requirement for external SD cards or cloud storage via internet connectivity. Data is saved in CSV format and can be offline-analyzed through software like GNU Octave. The device is a non-wearable, contact-based measurement system that provides up to 98.84% accuracy for SpO_2. Prototype testing on multiple users verified its strength, ease of accessing data, and appropriateness as a low-cost, standalone fitness companion independent of network resources [15].

A complete summary of the history, technical details, and usage areas of the Raspberry Pi was given by Hirak Dipak Ghael, Dr. L. Solanki, and Gaurav Sahu (2021). Their review highlighted the evolution of Raspberry Pi as a mere educational microcomputer into an all-purpose embedded system utility with global application in automation, artificial intelligence, and IoT-based solutions. The authors documented the hardware evolution of the Raspberry Pi, particularly focusing on the upgrade from the initial BCM2835-powered early models to the more sophisticated Raspberry Pi 4 with its quad- core processors and improved connectivity. They investigated the use cases in the areas of home automation, motion-based security systems, and do-it-yourself AI assistants, highlighting the feature of the platform in supporting multiple programming languages and peripherals. While the paper generally addresses the entire Raspberry Pi family as opposed to the Pico model, it solidifies the Pi ecosystem's image as a cheap, versatile option for prototyping as well as deployment of intelligent automation systems.[16].

III. INFORMATION

This review paper utilises a qualitative and comparative methodology to examine the contribution and influence of the Raspberry Pi Pico in IoT-based automation systems. The research is organized around a systematic review of scholarly literature, technical documentation, and practical project deployments, with the emphasis on the integration of the Pico platform into smart automation applications.

A. Selection and Process of Reviewing Literature

The initial stage of the methodology was the gathering of applicable research papers, journal articles, and techni- cal reports from established databases like IEEE Xplore, ScienceDirect, ASEE, IJCRT, IRJMETS, and IJRPR. Main-keywords used in the search were ”Raspberry Pi Pico,” ”IoT automation,” ”smart systems,” ”sensor integration,” and ”embedded microcontroller projects.” There were 9 main papers published between 2021 and 2025 chosen on relevance, technical intensity, and practical application of Raspberry Pi Pico in IoT settings.

B. Thematic Categorization

After reading the papers shortlisted, the content was thematically classified into the following application areas:

Home Automation
Industrial Monitoring and Safety
Agricultural IoT Systems Surveillance
and Security
Educational and Laboratory Platforms

This classification aided in comparing the usage of Raspberry Pi Pico across various IoT environments.

C. Comparative Analysis

Every implementation was assessed on the basis of the following parameters:

1) *System Architecture:* Hardware utilized, sensors, and peripheral integration

2) *Programming Environment:* Utilization of MicroPython, C/C++, IDEs such as Thonny or Arduino IDE

3) *Communication Modules:* Wi-Fi (ESP8266/ESP32), GSM, Bluetooth

4) *Automation Features:* Real-time control, data logging, alert systems, cloud integration

5) *Power Efficiency and Cost:* Applicability to low-power, budget-limited IoT applications

IV. Technical Specifications

The Raspberry Pi Pico, released in January 2021 by the Raspberry Pi Foundation, is a low-cost, small, and power-efficient microcontroller board for embedded systems and IoT. At the heart is the RP2040 microcontroller, which is specially developed to provide accurate control of digital and analog devices in real-time systems. The technical features and specifications of the Pico are as follows:

A. Microcontroller: RP2040

1) *Processor:* Dual-core ARM Cortex-M0+
2) *Clock Speed:* Up to 133 MHz
3) *On-chip SRAM:* 264 KB (distributed over six banks)
4) *External Flash Memory:* 2 MB QSPI Flash (onboard)
5) *DMA Controller:* For optimized memory access without CPU load
6) *Built-in ROM:* Includes USB bootloader and library functions

B. Input/Output and Interfaces

1) *GPIO Pins:* 26 multi-function GPIO pins (3.3V logic)
2) *Analog Inputs:* 3× 12-bit ADC channels
3) *PWM Channels:* 16 channels (from 8 PWM blocks)
4) *Timers:* 4 programmable timers with 2 alarms each
5) *PIO (Programmable I/O):* 2 PIO blocks (each contain- ing 4 state machines) for implementing custom I/O protocols in hardware

C. Power Supply and Consumption

1) *Operating Voltage:* 1.8V to 5.5V (VSYS input)
2) *Logic Level:* 3.3V (all GPIO)
3) *USB Interface:* Micro-USB port for power and data
4) *Low-Power Modes:* Sleep and dormant modes available via firmware
5) *Typical Power Consumption:* 1.3 mA (active), ¡1 mA (sleep)

D. Physical and Development Specifications

1) *Form Factor:* 21 mm × 51 mm
2) *Weight:* Around 3 grams
3) *Mounting Holes:* 3 through-hole mounting holes
4) *Programming Languages:* C/C++ SDK, MicroPython, CircuitPython
5) *Development Tools:* Thonny IDE, Arduino IDE (through unofficial core), VS Code with Pico SDK

E. Special Features

1) *Built-in Temperature Sensor:*
2) *Boot Selection Button:* To boot in USB mode
3) *Debug Port:* 3-pin SWD (Serial Wire Debug) interface
4) *No Built-in Wireless:* Needs external modules (e.g., ESP8266/ESP32) for Wi-Fi or Bluetooth connectivity

V. Port Details

The Raspberry Pi Pico is a low-performance, affordable mi-crocontroller board designed by the Raspberry Pi Foundation. The Raspberry Pi Pico is built on the RP2040 microcontroller chip and has several input/output ports that enable it to communicate with sensors, actuators, and peripherals. The ports offer basic communication and interfacing functions for embedded systems and IoT implementation.

A. GPIO Ports (General Purpose Input/Output)

1) The Pico has 26 multi-function GPIO pins.:
2) They can be configured as digital input or output to read switches or drive LEDs.:
3) GPIOs are programmable and can be used for any number of different functions like UART, SPI, I2C, PWM, etc.
4) They are 3.3V logic level.

B. Analog Input Ports (ADC)

1) The Pico has 3 analog input channels : GPIO26, GPIO27, and GPIO28.
2) These are linked to a 12-bit ADC (Analog-to-Digital Converter), which enables it to capture analog voltages from sensors (e.g., temperature or light sensors).
3) Voltage input range: 0V to 3.3V.

C. PWM Ports (Pulse Width Modulation)

1) Every GPIO can be made a PWM output.:
2) The RP2040 has 16 PWM channels, which are helpful for driving motors, LEDs (intensity), and producing analog- like signals.

D. UART Ports (Universal Asynchronous Re-ceiver/Transmitter)

1) Two UART ports: UART0 and UART1 for serial com-munication.
Used for communication with GPS modules, Bluetooth modules, etc.
2) Default pins: UART0: GPIO0 (TX), GPIO1 (RX)
UART1: GPIO4 (TX), GPIO5 (RX)

E. SPI Ports (Serial Peripheral Interface)

1) Two SPI interfaces: SPI0 and SPI1.
2) Used for high-speed communication with peripherals such as displays, SD cards, and sensors.:
3) Default SPI0 pins: GPIO16 (RX), GPIO17 (CSn), GPIO18 (SCLK), GPIO19 (TX).

F. I2C Ports (Inter-Integrated Circuit)

1) Two I2C interfaces: I2C0 and I2C1.
2) Appropriate for interfacing many low-speed devices such as sensors, EEPROMs, etc.:
3) Default I2C0 pins: GPIO4 (SDA), GPIO5 (SCL).

G. USB Port

1) A Micro-USB port is utilized for: Power supply
Programming the Pico (drag-and-drop)
USB communication (can be used as USB host or device)

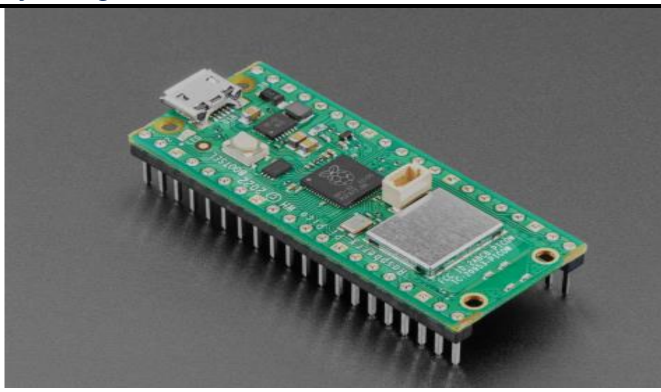

Fig.1. Raspberry Pi Pico

H. Power and Ground Pins

1) VBUS: 5V from USB
2) VSYS: System power (input: 1.8V to 5.5V)
3) 3V3: Regulated 3.3V output
4) GND: Ground connections (several available)

I. Debug Port (SWD – Serial Wire Debug)

Two pins for debugging using the SWD interface
GPIO24 (SWCLK)
GPIO25 (SWDIO)
Allows low-level access for development and troubleshooting.

VI. WORKING

The Raspberry Pi Pico functions on the basis of the RP2040 microcontroller, a dual-core ARM Cortex-M0+ processor de- signed by Raspberry Pi. Upon power supply through USB or an external source (using the VSYS pin), the onboard voltage regulator provides the input to a stable 3.3V needed by the RP2040 and its peripherals. During bootup, the micro- controller looks for executable firmware in its external QSPI flash memory (usually 2MB). Holding the BOOTSEL button at connection puts the Pico into USB boot mode, where users can drag-and-drop firmware files onto it.

After loading firmware, the processor will start running user-provided code. This can be MicroPython, C, or C++ code and can control the GPIO pins for digital or analog signal management. The GPIOs are employed to talk with a variety of sensors, actuators, and displays. There are digital communication protocols like I2C, SPI, and UART to enable the Pico to exchange data with external peripherals. The onboard 12-bit analog-to-digital converter (ADC) is used for reading analog sensors, and PWM signals can be created to drive motors and LEDs.

There are Programmable I/O (PIO) blocks on the Pico in addition to its primary cores, which provide accurate control of timing-critical data operations or the programming of custom communication protocols. The dual-core architecture of the RP2040 enables parallel tasking, improving performance in multitasking applications like data acquisition and real-time control. Power savings features such as low-power sleep modes are also supported by the Pico, which is thus well- positioned for energy-efficient embedded systems. The inte- gration of solid input/output handling, real-time operation, and simplicity of programming makes the Raspberry Pi Pico an efficient microcontroller for a wide variety of Internet of Things (IoT), automation, and educational uses.

VII. EASE OF USE

Raspberry Pi Pico is user-friendly and very much accessible to beginners, educators, and professionals alike. It allows for both MicroPython and C/C++ programming languages, giv- ing users with different expertise levels flexibility. Firmware upload is also made simple through a drag-and-drop USB bootloader, without the need for special programming tools. Its small, breadboard-compatible size, with well-marked GPIO pins, makes it easy to prototype and integrate into circuits. The Pico comes with beginner-friendly development platforms like Thonny and Visual Studio Code, which provide an improved coding experience. Along with detailed documentation, official SDKs, and an active online community, the learning curve is also smooth. Onboard debugging support through SWD and built-in peripherals such as LEDs and timers make early experimentation easy. These features combined guarantee that the Raspberry Pi Pico continues to be a mighty but simple- to-use platform for efficient rapid development in embedded systems.

VIII. ADVANTAGES AND DISADVANTAGES

A. Advantages

- Low Cost: Very reasonably priced, making it perfect for mass deployment and student projects.
- Low Power Consumption: Perfect for battery-powered and low-power IoT systems.
- Flexible I/O: 26 GPIO pins with ADC, PWM, UART, SPI, and I2C support.
- Dual-Core Processor: Permits parallel processing, en- hancing performance for multitasking.
- Programmable I/O (PIO): Facilitates custom peripheral simulation and tight signal control.
- Easy Programming: Supports MicroPython and C/C++, with friendly tools such as Thonny IDE.
- Compact Size: Tiny shape factor makes it perfect for embedded and space-restricted projects.

B. Disadvantages

- No Built-in Wi-Fi or Bluetooth: Needs external modules for wireless connectivity.
- Limited Flash Memory (2MB): Not suitable for data-intensive applications.
- Does not have an Operating System: Does not support Linux or heavy multitasking.
- No built-in Real-Time Clock (RTC): Needs external RTC modules for time-related operations.
- No onboard debugging through USB: Debugging needs extra setup through SWD interface.

IX. APPICATIONS

The Raspberry Pi Pico is widely used in various IoT-based automation systems due to its low cost, low power consumption, and flexible GPIO interface. Its applications span across several key areas:

1) *Home Automation:* Used for controlling lights, fans, and appliances via voice or mobile commands, often

with Bluetooth or Wi-Fi modules.

2) Industrial Monitoring: Helps detect gas leaks, fire, and motion, and sends alerts using GSM or IoT platforms.

3) Smart Agriculture: Automates irrigation by monitoring soil moisture, temperature, and humidity.

4) Waste Management: Operates smart dustbins that open automatically and track garbage levels using sensors.

5) Security Systems: Powers low-cost surveillance robots and motion-detection systems for homes and small offices.

6) Education: Used in engineering labs to teach microcon- troller programming and build IoT prototypes.

X. CONCLUSION

The Raspberry Pi Pico is a versatile, cost-effective micro-controller for creating IoT-based automation systems. With its flexibility, simplicity of programming using MicroPython, and multi-protocol communication compatibility, it can be applied across home automation, industries, agriculture, education, and smart cities. Without the inbuilt wireless connectivity, it is still an attractive option for low-power real-time control applications when used with external modules such as Wi-Fi or GSM.

The following review has emphasized several successful applications and research projects that reveal the increasing involvement of the Pico in smart systems. With IoT adoption increasing, the simplicity and scalability of the Pico will further make it an important tool for both developers and learners in the automation and embedded space.

REFERENCES

[1] M. Ramakrishna, G. Swapna, P. Naga Harika, and M. Manoj Kumar, "Voice Control Based Home Automation Using Raspberry Pi Pico," International Journal of Creative Research Thoughts (IJCRT), vol. 13, no. 1, pp. 1208–1214, Jan. 2025, ISSN: 2320-2882.

[2] B. B. Rupanar, V. S. Shilawant, A. R. Pawar, and M. M. Zade, "IoT Flood Monitoring and Alerting System Using Raspberry Pi Pico," International Journal of Research Publication and Reviews (IJRPR), vol. 6, no. 3, pp. 4697–4702, Mar. 2025, ISSN: 2582-7421.

[3] S. H. M. Mehr, "CtrlAer: Programmable real-time execution of scientific experiments using a domain specific language for the Raspberry Pi Pico/Pico 2," *SoftwareX*, vol. 30, p. 102175, May 2025.

[4] Z. C. M. Davidson, S. Dang, and X. Vasilakos, "Blended Laboratory Design Using Raspberry Pi Pico for Digital Circuits and Systems," IEEE Transactions on Learning Technologies, vol. 17, pp. 1170–1183, 2024, Article ID: 10423829, DOI: 10.1109/TLT.2024.3363230.

[5] D. A. B. Bonifacio, U. B. Rodriguez, J. Alcántara-Núñez, A. D. dos Santos, D. V. S. Carvalho, H. M. Murata, M. B. Perotoni, M. Moralles, M. Rosezky, "Open-source hardware and cost-effective gamma-ray spectrometer using Raspberry Pi Pico," *Radiation Physics and Chemistry*, Preprint, Oct. 2024. Available: https://ssrn.com/abstract=4996723.

[6] K. Kasiselvanathan, T. Kanaga, and A. Shanmugapriya, "Wireless Video Surveillance Robot using Raspberry Pi Pico," International Journal for Research in Applied Science and Engineering Technology (IJRASET), vol. 11, no. 5, pp. 4536–4542, May 2023, ISSN: 2321-9653.

[7] K. Chirunadh, G. S. V. Goud, G. Tejaswini, and B. S. Rao, "Rasp- berry Pi Pico Industrial Sensor Monitoring System with GSM Web App/IoT Technology," International Research Journal of Modernization in Engineering Technology and Science (IRJMETS), vol. 5, no. 6, pp. 2844–2849, Jun. 2023, e-ISSN: 2582-5208. DOI: 10.56726/IR- JMETS42308.

[8] D. R. Loker, "Raspberry Pi Pico as an IoT Device," Proceedings of the American Society for Engineering Education (ASEE), 2023, Paper ID 37487. Available: https://www.asee.org/public/conferences/

[9] G. Dhanalakshmi, B. Dasoju, G. Vijayabhasker, and G. Srinivas, "IoT Based Agriculture System Using a Wireless Sensor Network with Raspberry Pi Pico," Dogo Rangsang Research Journal, vol. 13, no. 4, pp. 131–135, Apr. 2023, ISSN: 2347-7180.

[10] D. Doropoulou, V. Angelopoulos, P. Amanatidis, T. Lagkas, and D. Karampatzakis, "Teaching Embedded Systems and IoT at the University using MicroPython and Raspberry Pi Pico," *Proceedings of the 27th Pan-Hellenic Conference on Progress in Computing and Informatics (PCI 2023)*, Lamia, Greece, Nov. 24–26, 2023, ACM, Article 111, 6 pages. doi: 10.1145/3635059.3635079.

[11] M. G. Sumithra, B. Banupriya, and R. Mathankumar, "Smart Home Automation using Raspberry Pi Pico and Blynk IoT," *International Journal for Research in Applied Science and Engineering Technology (IJRASET)*, vol. 11, no. 3, pp. 1077–1081, Mar. 2023, ISSN: 2321-9653.

[12] N. D. Duong, "Utilizing Raspberry Pi Pico microcontroller board in teaching and training students," *Journal of Science and Technology – Binh Duong University*, vol. 6, no. 3, pp. 141–148, 2023. doi: 10.56097/binhduonguniversityjournalofscienceandtechnology.v6i3.179.

[13] J. Crabtree, "Raspberry Pi Pico Interactive Picture Frame," *SWE 6823 Embedded Systems – Final Report*, Kennesaw State University, May 1, 2023.

[14] S. Kumar and R. Kumar, "Application of Raspberry Pi Pico in IoT-based Systems," in *Internet of Things and Its Applications*, 1st ed., CRC Press, 2023, pp. 317–333. doi:10.1201/9781003471059-27.

[15] G. Di Lorenzo, R. H. S. Carp, J. V. Terry, D. Alonso-Álvarez, and A. Papageorgiou, "Raspberry Pi Pico: A Versatile, Low-Cost, High-Precision Control Platform for Laboratory Experiments," *arXiv preprint*, arXiv:2206.08662v3, Jun. 2022.

[16] J. B. Thiyagarajan and M. Thothadri, "Fitness Monitoring System with Raspberry Pi Pico," *International Journal of Science Academic Research (IJSAR)*, vol. 2, no. 7, pp. 1840–1845, Jul. 2021, ISSN: 2582-6425.

[17] H. D. Ghael, L. Solanki, and G. Sahu, "A Review Paper on Raspberry Pi and its Applications," International Journal of Advances in Engineering and Management (IJAEM), vol. 2, no. 12, pp. 225–227, Dec. 2020, ISSN: 2395-5252. DOI: 10.35629/5252-0212225227.