WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Internet of Things (IoT): Study of Arduino and Raspberry Pie and their Applications in Various Domains

Princy Sharma

Assistant Professor, Computer Science, Department of Computer Engineering, GCET, University of Jammu

DOI: https://doi.org/10.55248/gengpi.4.923.92507

ABSTRACT

The Internet of Things (IoT) is the most emerging area of research nowadays. IoT refers to the development of the interconnectivity of devices. It is increasingly used in various industries including medicine, engineering, safety, and transportation for wireless communication. Wireless technologies include Bluetooth, ZigBee, and Long Range Wide Area Networks (LoRa WAN). There is a rapid development of new standards, technologies, and platforms for the IoT ecosystem. There are various platforms for analytics of data like thinkspeak and Blynk Cloud. This paper represents an overview of different protocols and analytics platforms for building IoT prototypes. This research paper presents a comparative study of Arduino and Raspberry Pi platforms in the context of Internet of Things (IoT) applications. The aim is to analyze the features, capabilities, and suitability of each platform for different IoT domains. The study explores their hardware specifications, programming capabilities, ease of use, and community support.

Keywords: Bynkapp, ThingSpeak, Raspberry pie, Aurdino

Introduction

The Internet of Things is one of the fastest-growing industries for bringing social and economic advantages to emerging and developing economies. The Internet of Things (IoT) is a collection of interconnected computing devices, physical devices, embedded electronics, sensors, actuators, and network connectivity with unique identifiers and the ability to transport data across a network without human or computer intervention [5], [12]. IoT enables remote control of systems over the internet. IoT is gaining traction in many fields, such as medicine, transportation, and agriculture [3]... The Internet of Things (IoT) consists of two key elements: "Internet" and "things". Communication allows things to coordinate their actions and reach decisions together it allows them to hear, see, think, compute, and act. Technology gives things to make authoritative decisions that benefit various applications using intelligence and consensus. From the standpoint of passive observers, they transform objects or sensors into active members of a computing system, communicating, working collaboratively, and making critical decisions. As a result, they present challenges that require specialized communication standards [3]. IoT is predicted to grow swiftly because of advancements in communication technology, device availability, and computational systems [7]. There are a variety of IoT devices on the market, and the IoT device selection is based on the application and task to be completed. Arduino and Raspberry Pi, on the other hand, are the most widely utilized. IoT devices because of their ease of use and operation. Furthermore, compared to other IoT devices, Arduino and Raspberry Pi are less expensive [13]–[4].

WIRELESS TECHNOLOGIES

Bluetooth:

Bluetooth is a short-range wireless technology primarily designed for personal area networks (PANs). It operates in the 2.4 GHz frequency range and supports data transmission over short distances, typically up to 10 meters. Bluetooth is commonly used for connecting devices such as smartphones, tablets, laptops, and peripherals like keyboards, mice, and headphones. It offers low power consumption, easy pairing, and a secure communication link. Bluetooth is widely utilized for applications such as audio streaming, wireless data transfer, and Internet of Things (IoT) devices

ZigBee:

ZigBee is a wireless communication protocol designed for low-power, low-data-rate applications. It operates in the 2.4 GHz frequency band and offers relatively longer-range coverage compared to Bluetooth. ZigBee networks are typically formed by creating a mesh network topology, where multiple devices (nodes) communicate with each other to relay messages. It is commonly used in home automation systems, smart lighting, industrial monitoring, and control applications. ZigBee provides excellent energy efficiency, scalability, and support for a large number of devices in a network.

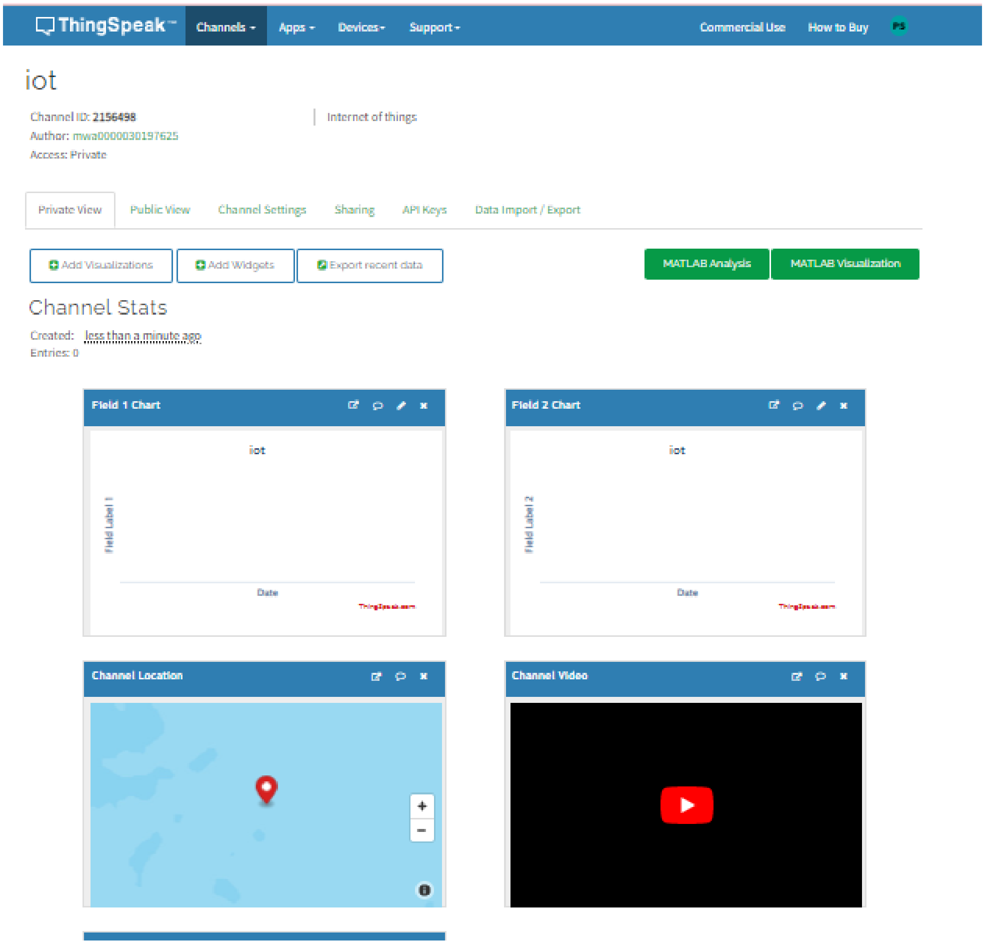
Long Range Wide Area Networks (LoRaWAN):

LoRaWAN is a wireless technology specifically designed for long-range communication and low-power applications. It operates in the sub-GHz frequency bands (such as 868 MHz in Europe and 915 MHz in North America). LoRaWAN uses a spread spectrum modulation technique called chirp spread spectrum (CSS), which allows for long-range communication with low power consumption. It is well-suited for applications that require wide-area coverage, such as smart cities, agriculture, asset tracking, and environmental monitoring. LoRaWAN networks typically consist of gateways that receive data from remote sensors or devices and forward it to a central server or cloud platform.

Different platforms for analytics of Data: In the context of IoT

ThingSpeak

ThingSpeak is an IoT analytics platform developed by MathWorks. It provides a cloud-based environment for collecting, analyzing, and visualizing data from IoT devices. ThingSpeak allows users to create IoT projects and connect their devices to the platform for data acquisition. It supports various communication protocols such as MQTT, HTTP, and HTTPS for data transmission. ThingSpeak provides built-in capabilities for data visualization through customizable charts and graphs. It also offers features like data logging, real-time data processing, and triggering of alerts based on predefined conditions. Additionally, ThingSpeak allows integration with other platforms and services, making it a versatile tool for IoT data analytics.



iot

Channel ID: 2156498
Author: mwa0000030197625
Access: Private

Internet of things

Channel Stats

Created: less than a minute ago
Entries: 0

Fig 1: (ThingSpeak Account with public channel)

Blynk Cloud:

Blynk Cloud is an IoT platform that focuses on providing a user-friendly interface for developing and controlling IoT projects. It offers a mobile app (available for both iOS and Android) and a cloud-based backend infrastructure. Blynk allows users to create custom IoT applications by drag-and-drop interface elements called "widgets" on the mobile app. These widgets can be linked to various IoT devices or sensors. Blynk provides a simple way to establish communication between the mobile app and the IoT hardware using supported protocols like Wi-Fi, Ethernet, or Bluetooth. The platform also includes built-in features for data visualization, notifications, and control interfaces. Blynk Cloud offers integration with popular IoT development boards and supports a wide range of hardware platforms.

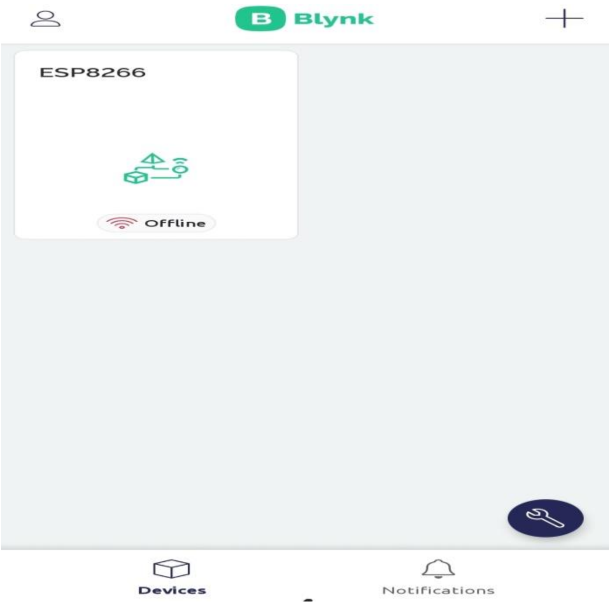


Fig2: (Blynkapp Account)

Both ThingSpeak and Blynk Cloud provide powerful tools and services for IoT data analytics, but they have slightly different focuses. ThingSpeak emphasizes data collection, analytics, and visualization, making it suitable for projects that require more complex data processing and analysis. On the other hand, Blynk Cloud prioritizes ease of use and quick prototyping, enabling users to create IoT applications with a visual interface without much programming knowledge.

Provide an overview of Arduino and Raspberry Pi.

Aurdino is an open-source electronics platform designed for building interactive projects. It consists of a microcontroller board and a development environment, making it accessible for beginners and professionals alike. Arduino boards are available in various models, but they typically include digital and analog input/output pins, which allow them to interact with sensors, actuators, and other electronic components. Arduino uses a simplified version of C/C++ programming language and provides an integrated development environment (IDE) for writing and uploading code to the board. It is known for its simplicity, affordability, and wide range of applications in areas such as robotics, home automation, and prototyping.

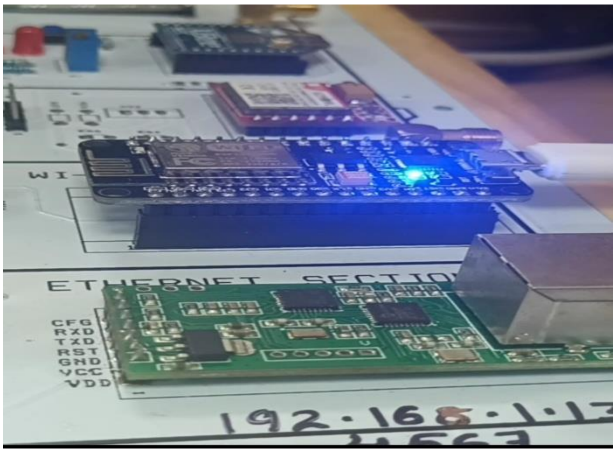


Fig3: Arduino placed on Iot and embedded development kit

Raspberry Pi:

Raspberry Pi is a single-board computer developed to promote computer science education and DIY projects. It is a credit-card-sized board that incorporates a processor, memory, and input/output interface. Raspberry Pi runs on Linux-based operating systems, such as Raspbian, and supports various programming languages, including Python, C/C++, and Java. It offers HDMI output, USB ports, Ethernet connectivity, and GPIO (General-Purpose Input/Output) pins, allowing it to connect to displays, and peripherals, and interact with external devices. Raspberry Pi is known for its versatility, higher computational power compared to Arduino, and its ability to handle more complex tasks. It is commonly used for projects like media centers, home automation, IoT applications, and even as a small-scale server

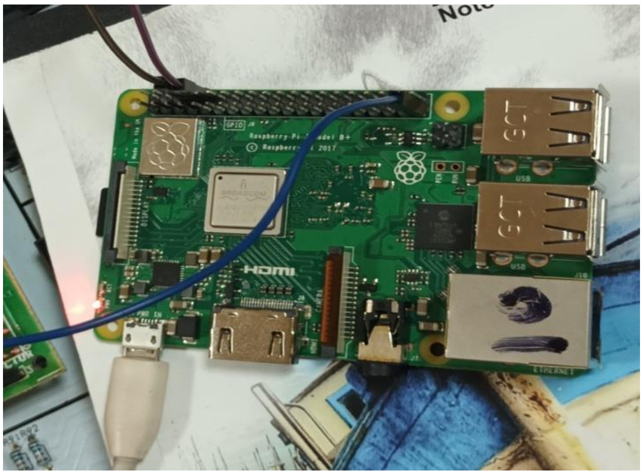


Fig 4: Raspberry Pi

Experimental setup of Audino and Rasberry pi

Audino

The experimental setup for Arduino can vary depending on the specific project or task you are working on. However, I can provide you with a general outline of the components and steps involved in setting up an Arduino experiment:

- **Components:**

Arduino board: Choose the appropriate Arduino board based on your project requirements (e.g., Arduino Uno, Arduino Mega, etc.).

- **Sensors/Actuators:**

Select the sensors or actuatorsyou need for your experiment (e.g., temperature sensor, ultrasonic sensor, LED, motor, etc.).

Breadboard/Jumper wires: Use a breadboard and jumper wires to create the circuit connections between the Arduino board and the components.

- **Power supply:**

Provide power to the Arduino board and the connected components using a USB cable or an external power source (e.g., battery pack or power adapter).

- **Circuit Connection**:

Identify the required input/output pins on the Arduino board for connecting the components.

Use jumper wires to establish the connections between the Arduino board and the sensors/actuators. Follow the component datasheets or tutorials for proper wiring instructions.

Place the components on the breadboard and connect them to the appropriate pins on the Arduino board.

Download and install the Arduino Integrated Development Environment (IDE) from the official Arduino website.Connect the Arduino board to your computer using a USB cable.Launch the Arduino IDE and select the correct Arduino board and port under the Tools menu.

- **Programming:**

Write the code for your experiment in the Arduino IDE. Define the pin configurations, read sensor values, control actuators, and implement the desired functionality.

Upload the code to the Arduino board by clicking on the "Upload" button in the Arduino IDE.

Below is an code that uses the ESP8266 WiFi module, the ArduinoJson library for parsing JSON responses, the ThingSpeak library for sending data to ThingSpeak, and the Adafruit Motor Shield library for controlling the motor based on sensor input:

```
#include <ESP8266WiFi.h>          // ESP8266 WiFi module library

#include <ArduinoJson.h>          // ArduinoJson library

#include <Adafruit_MotorShield.h>  // Adafruit Motor Shield library

#include <Wire.h>

#include <ThingSpeak.h>           // ThingSpeak library
```

```
const char* ssid = "Princy";        // WiFi credentials
const char* password = "123456789";
// ThingSpeak information
const char* thingSpeakAddress = "api.thingspeak.com";
const String thingSpeakWriteAPIKey = "VMU3S9PERGA01PXL";
const unsigned int thingSpeakChannelID =  2156498;
Adafruit_MotorShield AFMS = Adafruit_MotorShield();  // Create an instance of Adafruit Motor Shield
Adafruit_DCMotor* motor = AFMS.getMotor(1);         // Create a motor object
const int sensorPin = A0;       // Sensor input pin
const int threshold = 500;     // Threshold value for motor activation
void setup()
{
AFMS.begin();                 // Initialize the Adafruit Motor Shield
 motor->setSpeed(255);         // Set the motor speed (0-255)
pinMode(sensorPin, INPUT);    // Set the sensor pin as input
Serial.begin(9600);           // Initialize serial communication
connectToWiFi();              // Connect to WiFi network
ThingSpeak.begin(client);     // Initialize ThingSpeak
}
void loop() {
 int sensorValue = analogRead(sensorPin);  // Read the sensor value
 if (sensorValue > threshold) {
    motor->run(FORWARD);       // Run the motor in the forward direction
    Serial.println("Motor ON");
    sendToThingSpeak(sensorValue); // Send sensor value to ThingSpeak
  }
Else
 {
    motor->run(RELEASE);        // Stop the motor
    Serial.println("Motor OFF");
  }
  delay(100);                  // Delay for stability
}
void connectToWiFi()
 {
  Serial.println();
  Serial.print("Connecting to WiFi");

  WiFi.begin(ssid, password);    // Connect to WiFi network
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println();
  Serial.print("Connected to WiFi, IP address: ");
  Serial.println(WiFi.localIP());
}
void sendToThingSpeak(int value)
{
  ThingSpeak.setField(1, value);  // Set the value to be sent to ThingSpeak
int response = ThingSpeak.writeFields(thingSpeakChannelID, thingSpeakWriteAPIKey);
    if (response == 200)
{
    Serial.println("Data sent to ThingSpeak");
  }
else
{
    Serial.print("Error sending data to ThingSpeak. Response code: ");
    Serial.println(response);
  }
}
```

- **Testing and Observation:**

Power on the Arduino board and observe the behavior of the connected components.Use the serial monitor in the Arduino IDE to view debug information or print sensor readings.Make observations, collect data, and analyze the results based on your experiment objectives. the Arduino community for specific instructions and best practices related to your chosen components and experiment.

Raspberry Pi

The experimental setup for Raspberry Pi can vary depending on the specific project or task you are working on. However, I can provide you with a general outline of the components and steps involved in setting up a Raspberry Pi experiment:

- **Components:**

Raspberry Pi board: Choose the appropriate Raspberry Pi board based on your project requirements (e.g., Raspberry Pi 4, Raspberry Pi Zero, etc.).Power supply: Provide power to the Raspberry Pi board using a suitable power adapter or USB cable.

- **MicroSD card:**

Insert a microSD card into the Raspberry Pi board to install the operating system and store data.Peripherals: Connect peripherals such as a monitor, keyboard, and mouse for initial setup and configuration.

- **Operating System:**

Download the desired operating system for Raspberry Pi, such as Raspbian (official OS), Ubuntu, or other compatible distributions.Flash the downloaded OS image onto the microSD card using an image flashing tool like BalenaEtcher or Raspberry Pi Imager.

- **Initial Setup:**

Insert the microSD card into the Raspberry Pi board.Connect the necessary peripherals (monitor, keyboard, and mouse) to the Raspberry Pi. Power on the Raspberry Pi by plugging in the power supply.

- **Configuration:**

Follow the on-screen prompts to set up the operating system on the Raspberry Pi.Configure network settings (Wi-Fi or Ethernet) to enable internet connectivity.Optionally, configure other system settings, such as language, time zone, and password.

- **Enable VNC on Raspberry Pi:**

To enable VNC on the Raspberry Pi, you can follow these steps:

Open the Raspberry Pi configuration by opening the terminal and running the following command:

arduinoCopy code

```
sudo raspi-config
```

In the configuration menu, navigate to "Interfacing Options" and select "VNC" to enable VNC.

Confirm the change and exit the configuration menu.

Install VNC Viewer on your computer: Download and install VNC Viewer on the computer you want to use to remotely access the Raspberry Pi.

- **Connect Raspberry Pi and computer**:

Ensure that both the Raspberry Pi and the computer running VNC Viewer are connected to the same network.

Launch VNC Viewer: Open VNC Viewer on your computer.

1. Find the IP address of the Raspberry Pi: On the Raspberry Pi, open a terminal and enter the following command:

cssCopy code

```
hostname -Princy
```

2. **Connect to Raspberry Pi using VNC Viewer:** In VNC Viewer on your computer, enter the IP address of the Raspberry Pi in the address bar and click "Connect". You may be prompted to enter a username and password for the Raspberry Pi.
3. **Access Raspberry Pi remotely**: Once connected, you should be able to remotely access and control the Raspberry Pi using VNC Viewer. You will see the Raspberry Pi desktop on your computer/mobile screen
4. **Programming and Experimentation:**

Launch the programming environment suitable for your experiment. For example, Python IDE, Arduino IDE, or other development tools.Write your code or script to implement the desired functionality.Connect sensors, actuators, or other external devices to the Raspberry Pi's GPIO pins or USB ports as required by your experiment.Execute the code and observe the behavior of the connected components or sensors.

Collect data, analyze results, and make observations based on your experiment objectives.Remember to refer to the documentation, tutorials, and examples provided by the Raspberry Pi community or the specific sensor/actuator manufacturers for detailed instructions and best practices related to your experiment.

```
import RPi.GPIO as GPIO

import time

# GPIO pin assignments

motor_pin = 17          # Motor control pin

sensor_pin = 18         # Sensor input pin

# GPIO setup

GPIO.setmode(GPIO.BCM)

GPIO.setup(motor_pin, GPIO.OUT)

GPIO.setup(sensor_pin, GPIO.IN)

# Motor control function

def motor_control(value):
    if value > threshold:
        GPIO.output(motor_pin, GPIO.HIGH)    # Turn motor on
        print("Motor ON")
    else:
        GPIO.output(motor_pin, GPIO.LOW)     # Turn motor off
        print("Motor OFF")
try:
threshold = 500     # Threshold value for motor activation
while True:
sensor_value = GPIO.input(sensor_pin)   # Read the sensor value
motor_control(sensor_value)
time.sleep(0.1)   # Delay for stability
except KeyboardInterrupt:
GPIO.output(motor_pin, GPIO.LOW)    # Turn off motor
GPIO.cleanup()                    # Clean up GPIO pins
```

Popularity and applications OF Ardino and Raspberry Pi in Various domain

Arduino

- **Accessibility:**

Arduino's user-friendly design, simplified programming language, and extensive documentation make it accessible to beginners with little or no prior electronics or programming experience.

- **Open-Source Nature:**

Arduino is an open-source platform, which means that the hardware specifications, software libraries, and development tools are freely available for modification and sharing. This fosters a collaborative community where users can contribute and benefit from each other's projects and innovations.

- **Versatility:**

Arduino boards can be used for a wide range of applications, from simple blinking LED projects to complex robotics and home automation systems. The flexibility of the platform enables users to adapt it to their specific needs and create customized solutions.

- **Cost-Effective:**

Arduino boards are relatively inexpensive compared to other microcontroller options, making them an affordable choice for hobbyists, students, and professionals working on projects with budget constraints.

- **Extensive Community Support:**

Arduino has a large and active community of users, makers, and developers worldwide. This community provides forums, online tutorials, and resources, making it easy to seek help, learn, and collaborate on projects.

Raspberry Pi:

- **Computing Power**:

Raspberry Pi offers significantly more computational power compared to Arduino. It can run full-fledged operating systems, handle multitasking, and perform complex computations, making it suitable for a broader range of projects that require more processing capabilities.

- **Connectivity and I/O:**

Raspberry Pi has built-in Ethernet, USB ports, HDMI output, and GPIO pins, providing connectivity options to peripherals, networks, and external devices. This expands its usability for applications like media centers, servers, IoT projects, and home automation systems.

- **Community and Ecosystem**:

Similar to Arduino, Raspberry Pi has a large and passionate community that actively contributes to the platform. The community provides support, tutorials, and a wealth of projects and ideas, fostering innovation and collaboration.

- **Education Focus**:

Raspberry Pi was initially developed with a strong emphasis on education, aiming to promote computer science and programming skills. Its affordability and capabilities have made it a popular choice for educational institutions, teachers, and students to learn programming, electronics, and system-level concepts.

CONCLUSION

The integration of Arduino or Raspberry Pi with ThingSpeak provides a scalable solution for IoT applications. ThingSpeak's cloud-based architecture allows for easy scalability and real-time data analysis. It also provides integration with other popular platforms and services, enabling developers to create comprehensive IoT ecosystems.

The experiment demonstrated the successful integration of Raspberry Pi, Arduino, and ThingSpeak for motor sensor monitoring and control. The combination of these platforms provides a powerful solution for real-time data collection, analysis, visualization, and remote management. The insights gained from this experiment lay the foundation for future developments in motor control systems, with potential applications across various industries.

FUTURE SCOPE

In summary, the combination of Arduino, Raspberry Pi, and ThingSpeak offers a powerful platform for IoT development. It enables developers to collect, analyze, and visualize data from various IoT devices, opening up a wide range of possibilities for creating innovative and intelligent applications. With ongoing advancements and increasing demand for IoT solutions, the future looks bright for this technology stack. Further uses in Environmental Monitoring, Smart Agriculture, Home Automation, Industrial Automation, Healthcare Monitoring, Weather Monitoring, Energy Monitoring.

References

1. Arampatzis, T. Lygeros, J. Manesis, S. (2005) A survey of applications of wireless sensors and Wireless Sensor Networks. *In 2005 IEEE International Symposium on Intelligent Control & 13th Mediterranean Conference on Control and Automation. Limassol, Cyprus,* 1-2, 719- 724.

2. Ali M., V laskamp J.H.A, Eddiny N.N., Falconer B. and Oramc.(2013) ―Technical Development and Socioeconomic Implications of the Raspberry Pi as a Learning Tool in Developing Countries‖.*5th Computer Science and Electronic Engineering Conference* (CEEC),(pp. 103- 108).

3. C. W. Zhao, J. Jegatheesan, and S. C. Loon.(2015) “Exploring IoT application using Raspberry Pi, *International Journal of Computer Networks and Applications*,(vol. 2, no. 1, pp. 27–34).

4. R. I. S. Pereira, I. M. Dupont, P. C. M. Carvalho, and S. C. S. Jucá,(2018) IoT embedded Linux system based on Raspberry Pi applied to real-time cloud monitoring of a decentralized photovoltaic plant. Measurement, vol. 114, (pp. 286–297).

5. M. S. Chavan, V. P. Patil, S. Chavan, S. Sana, and C. Shinde. (2018) Design and implementation of IoT-based real-time monitoring system for aquaculture using Raspberry Pi. *International Journal on Recent and Innovation Trends in Computing and Communication.* (vol. 6, no. 3, pp. 159–161).

6. Nidhi Patel,(2018) IOT system for converting handwritten text to editable format. *Vasileios Sideridis , “GestureKeeper: Gesture Recognition for Controlling Devices in IoT.*

7. M. binti M. Noor and W. H. Hassan. (2018) Current research on internet of things (IoT) security: *A survey," Computer Networks,* (vol. 148, pp. 283–294).

8. F. Abate, M. Carratù, C. Liguori, and V. Paciello(2019). A *low cost smart power meter for IoT*,(vol. 136, pp. 59–66,).

9. M. Sridharan, R. Devi, C. S. Dharshini, and M. Bhavadarani(2019). IoT based performance monitoring and control in counter flow double pipe heat exchanger. *Internet of Things,* (vol. 5, pp. 34–40).

10. El-Hajj, M.; Fadlallah, A.; Chamoun, M.; Serhrouchni, A. (2019).A Survey of Internet of Things (IoT) Authentication Schemes. 19, 1141.

11. G. Vial(2019). Understanding digital transformation: A review and a research agenda.*The Journal of Strategic Information Systems*. (vol. 28(2) pp. 118–144).

12. R. Atiqur and Y. Li (2020). Automated smart car parking system using raspberry Pi 4 and iOS application. *International Journal of Reconfigurable and Embedded Systems (IJRES).* (vol. 9, no. 3, pp. 229–234).

13. P. Pawar, M. T. Kumar, and P. Vittal K (2019).An IoT based intelligent smart energy management system with accurate forecasting and load strategy for renewable generation(vol. 152).

14. Sachin Kumar, Prayag Tiwari & Mikhail Zymbler (2019). Internet of things is a revolutionary approach for future technology enhancement: *A Review*. Springer.

15. Chintan Patel, Nishant Doshi. (2020). A Novel MQTT Security framework In Generic IoT Model. *Third International Conference on Computing and Network Communications* (CoCoNet'19)1877-0509.

16. Sfar AR, Zied C, Challal Y.(2017) A systematic and cognitive vision for IoT security: a case study of military live simulation and security challenges. In: Proc. *2017 International conference on smart, monitored and controlled cities* (SM2C).