IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Impact Factor: 7.301

Volume 3, Issue 8, April 2023

An Efficient Intelligent Oil Well Monitoring System for Niger Delta Oil Fields

Julius B. Wosowei[1] and Chandrasekar Shastry[2]

Department of Computer Science and Engineering[1]

Department of Electrical and Communications Engineering[2]

Jain University, Bangalore, Karnataka, India

Julius.wosowei@gmail.com

Abstract: *Oil well monitoring has metamorphosed over the years from Dump Wells completion and the era of permanent gauges through to the hydraulic control wells until this era of intelligent Well completion. These efforts are geared towards an era where Well data can be collected and interpreted with no human intervention. The aim is to improve recovery (optimization), minimized OPEX and CAPEX, and general efficiency. However, intelligent monitoring by virtue of intelligent Well completion is still an expensive venture. The paper presents an efficient IoT-based monitoring system whereby an ESP32 microcontroller and sensors are used to monitor the Well pressure, temperature, level, and flow rate on a real-time basis. The data from the oil well is available to the user at any remote location because the sensor data is sent to a cloud service on the internet. The cloud service used is the MQTT protocol and the MIT APP Inventor. The sensor data is also viewed in an android Mobile App.*

Keywords: Oilfields, Intelligent Oil-well, IoT, Monitoring System, ESP32, Sensors.

I. INTRODUCTION

By design, a well is smart, intelligent or dump. In other words, its completion properties determine whether it is a smart well. 'Smart well completion' is described as "the design, selection and installation of equipment and the specification of treatment and procedures necessary to bring the well into production and subsequently to produce in a way which satisfies the operator's objectives for the field development [1]. In petroleum production, completion is the process of making a well stable for production (or injection). Essentially, this entails prepping the bottom of the hole to fulfill the specified specifications. Several design variables are being evaluated for oilfield performance/optimization, including the number of wells, line pressure, wellbore diameter, peak day requirements, and gathering system capacity. Additional aspects include geology and deposit environment formation features.

Despite the fact that research on smart wells began in the early 1990s, wells were completed using hydraulically moving sleeves and tubing. Yet, technological advancements such as downhole gauges, sliding sleeves, and surface-controlled subsurface safety valves have resulted in the development of smart wells. The smart well technology has progressed from dump wells to smart wells and finally intelligent wells [2].

Dump wells are done without gauges or remote valves inserted in the wellbore and rely solely on the construction of mechanically sliding sleeves, whereas permanent gauge wells are completed with no remote control from the surface, only downhole gauges, and data can be collected in a timely manner using data loggers. Hydraulic Remotely Controlled Wells, on the other hand, have no sensors installed in the bottom hole and only remote valves that can be operated hydraulically from the surface.

Smart wells, on the other hand, feature sensors and remote valves in the wellbore, and data may be processed by operating forces on the surface in real-time. Real-time temperature, pressure, and flow rate data can be acquired from the wellbore's bottom via connected or wireless equipment. These are also referred to as "Low IQ Intelligent Wells," whereas Intelligent Wells is the most advanced level of smart wells. It is a word used to describe wells where data is processed automatically using proper computer software and there is no artificial intervention. The illustration below depicts a well with Smart finishing.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 8, April 2023

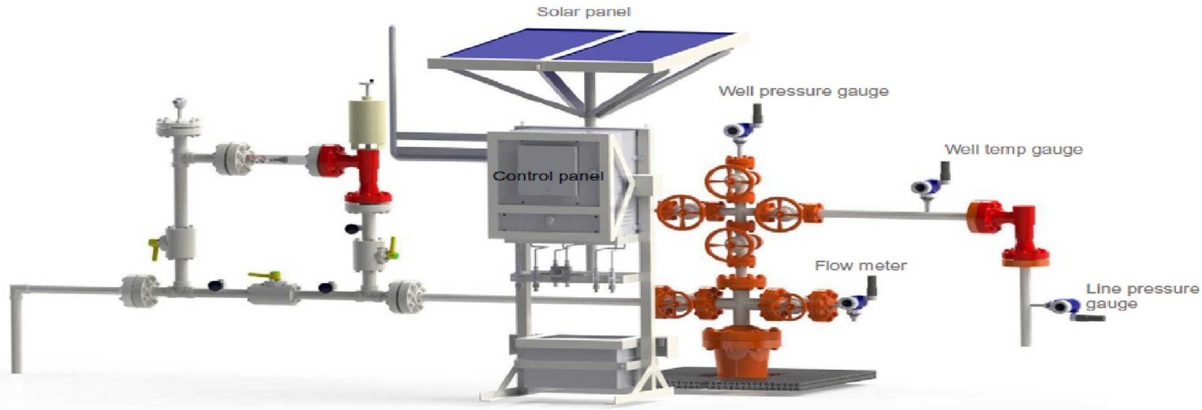

Figure 1: Automated Well (Courtesy of Rockwell Automation)

Nigeria's Petroleum and gas industry

Nigeria is more of a mono-economy, with oil and gas as its mainstay. Nigeria has a gas reserve of 202 trillion cubic feet and a production rate of 8.5 billion standard cubic feet per day (bscf/d) and an oil reserve of 36.972 billion barrels, which is equivalent to 207.6 billion standard cubic feet of natural gas, indicating that Nigeria's gas reserve is 900 times the country's proven oil reserve, making it the largest producer of natural gas in Africa and ninth in the world. Nigeria, with a geographic area of 923,768km2(356,669mile2) and a population of 206,139,589(2020), is located at 10°N, 8°E of the Sub-Saharan African region and shares borders with Benin Republic, Cameroon, Chad, Niger, and the Gulf of Guinea [3]. In 2019, the Nigerian oil sector accounts for 10% of GDP, 65% of government revenue, and 88% of foreign revenues, making it critical to maximise oil output in Nigeria.

II. REVIEW OF LITERATURE

Safaa, N.S et al [4] introduced an efficient IoT-based oil well monitoring system including a temperature sensor, pressure sensor, and gas sensor for data sensing in oilfields. It used an Arduino Uno microcontroller, Wi-Fi, ESP8266 transceiver, and GSM Module. The software applications used in the system were Arduino IDE, 3D builder, and proteus. The prototype made use of the Blynk App for data analysis, system notification messages, and the Android mobile monitoring system.

Neeraj and Dahud [5] proposed and implemented an IoT-based intelligent portable Data Acquisition system that utilised an AVR ATMEL microcontroller as the processing unit and ECLIPSE Software to develop the Android mobile application. The Android app sends data to the server and client applications. A temperature sensor, a light sensor, a gas sensor, a humidity sensor, a moisture sensor, and a level sensor are among the sensors used. Nevertheless, NETBEANS was utilised for the Java IDE and runs in JVM for module development.

Mohammed et al. [6] suggested an intelligent IoT-based monitoring system involving Smart objects for sensing essential field characteristics such as pressure, temperature, vibration, and others, as well as efficiently transferring sensed data to a Control Centre. Anomaly occurrences are also reported by the system for predictive maintenance.

Ali Ibrahim [7] suggested an efficient WSN based on ZigBee technology and a microchip (PIC Controller) that collects data from remote oil wells via wireless communication and sends it to a collection station. The simulator used includes an XBee radio module, and feedback from the wellhead safety values position is gathered by the microcontroller and transmitted to a remote location via the XBee module.

M. Al-Fadhli and A. Zaher[8] proposed a comprehensive automated method for monitoring various refinery processes. The activities of the concept include data collection from tanks, alert notification, pipeline data gathering, and the implementation of fault solutions and other smart features. Lab-view was used to implement it, while Lab-jack was utilised to collect wireless data. In terms of providing real-time and precise data about field equipment, the integration was a success. The system is a functional SCADA version that monitors pressure, temperature, level, and gas leaks.

Ayuba John [9] created and developed a wireless sensor network (WSN) for a real-time remote monitoring system for flowrate metering infrastructure. The system maintained accurate flow rate measurements from the wellheads to the export terminal. A flow sensor, pumping machine, microcontroller, Wi-Fi or wireless connectivity, and a database server comprise the WSN architecture. The data is saved in the database server, and the flow rate is delayed in real-time at the website, which acts as a human-machine interface.

Barani R [10] described a sensor-based intelligent control system for energy conservation and efficient oil health monitoring. Sensors such as a level sensor, a temperature sensor, and a gas sensor are used to monitor the status of the storage tank. The sensors are installed inside the oil storage tanks, and the sensor output is sent to an Atmega2560 microprocessor positioned in each well. The oil pump motor is controlled based on the condition of the oil storage tank, and data about each well is wirelessly communicated to the administrator in remote places, allowing the wireless sensor network to operate and monitor many wells.

III. METHODOLOGY OF PROPOSED WORK

The development follows the waterfall [SDLC] model with each component duly explained in the subsections below.

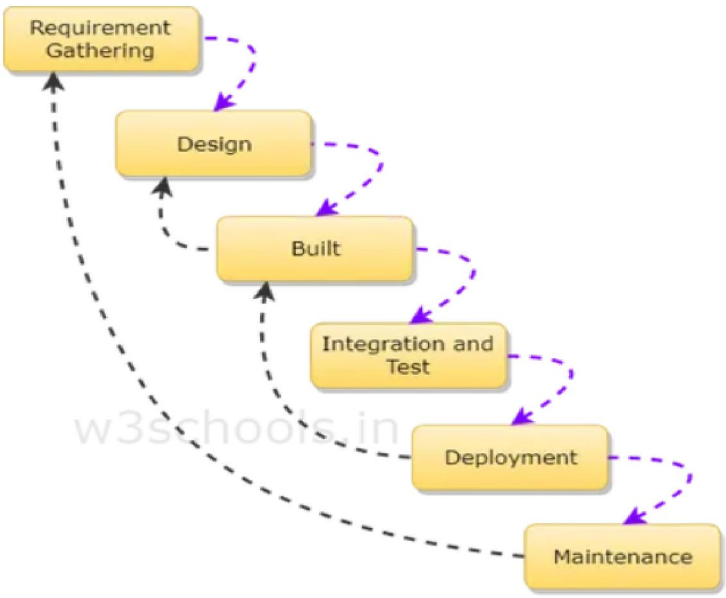

Figure 2: Waterfall Software Development Life Cycle

A. Collecting and Analyzing Requirements.

This design phase focuses on eliciting the requirements for an efficient intelligent oil well monitoring system using an Esp32 microcontroller. It also examines the project's financial and technological feasibility, tactics, and constraints. A requirement specification document captures and documents the aforementioned requirements.

B. Design Phase: The first phase's required specifications were analysed, and a system design or conceptual design was created. The hardware and software that comprise the architecture were identified and specified, including a 3D design using 3D builder software and circuit diagram modeling using Capital ElectraTM software modeling tools.

C. Build and Implementation Phase: This is the build and coding step. The system is first built in discrete programs called units and then incorporated into the next phase based on inputs from the system design. Unit Testing was used to test the functionality of the developed units. The idea is translated into source code and UI plus UX design utilizing programming language and tools at this point in the coding process.

D. Testing and integration.

When the units have been tested, all of the developed units from the implementation phase are combined into a single system in this step. The tests include 1. testing the software by the development team; 2. testing the software by friendly customers; and 3. testing the product for acceptability.

E. Deployment

After the functional and non-functional tests for release, the program is shown or installed on the user end during the deployment phase, as the name implies.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 8, April 2023

F. Maintenance

To make the program fully and flawlessly functional, the developer must update the product, patch problems, and errors, and develop additional key components based on user feedback. The various types of maintenance are as follows:

1. Corrective maintenance: repairing errors.
2. Perfective maintenance: To improve efficiency and effectiveness.
3. Adaptive maintenance: The process of transferring applications from one operating system (OS) to another.

IV. FLOWCHART AND 3D DESIGN

The figure below depicts the flowchart of the intelligent oil well monitoring system.

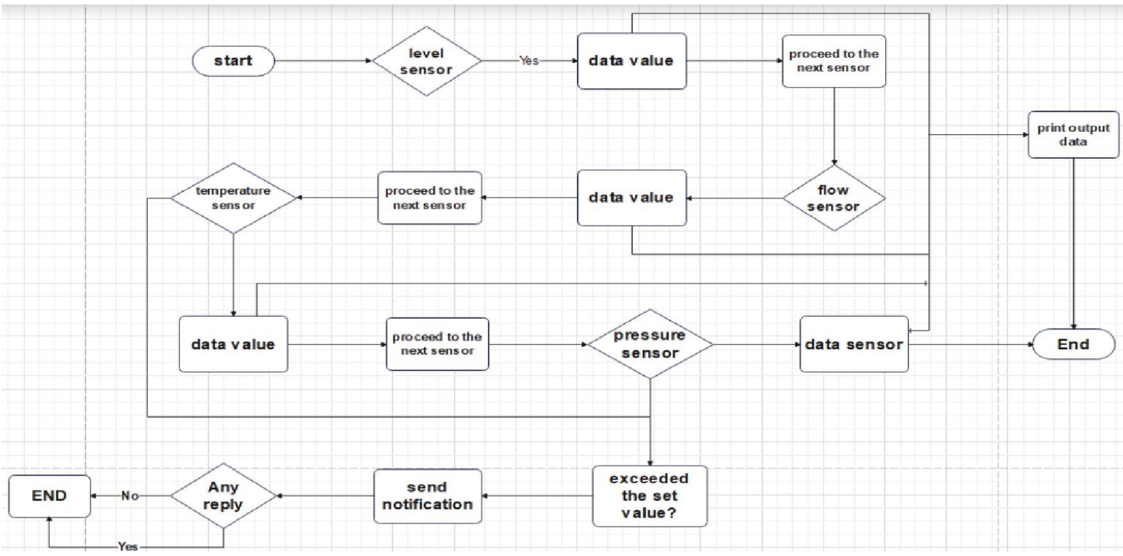

Figure 3: Flow Chart

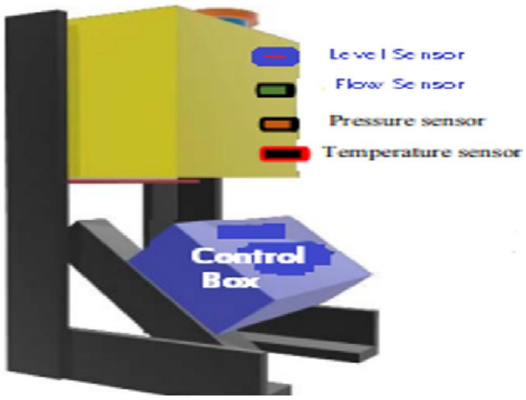

Figure 4: 3D Design model of the system

The 3D builder software was used to design the prototype which has tremendously improved the look and feel of the design and removed unnecessary steps and items in developing the prototype.

Block Diagram

The efficient intelligent oil well monitoring system using ESP32 is divided into the following parts as shown in figure 5 These include the oil well, sensors, and power supply which serves as the input source and others are ESP32 Controller, output source, and the MQTT protocol which enables the output value (serial monitor) to the mobile phone.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 8, April 2023

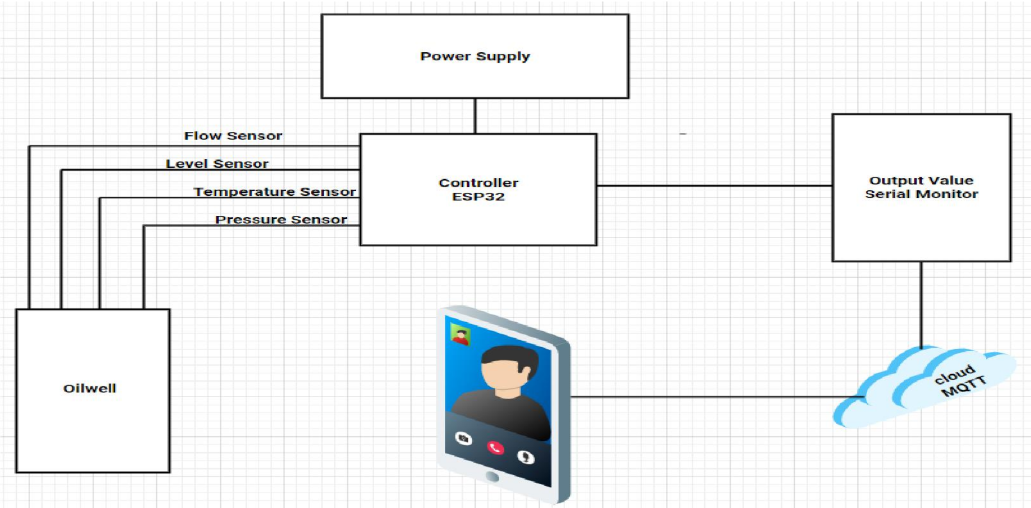

Figure 5: Block diagram of the efficient monitoring system.

A. Temperature Sensor (DS18B20) - This is a waterproof, sealed, and pre-wired digital temperature sensor probe-based DS18B20 sensor that is 1 meter long. The sensor's signal does not degrade over extended distances. It has the following specifications: temperature range (-55 to 125°C (-67°F to +257°F), 64bit ID burned into the chip, accuracy (0.5°C accuracy from -10°C to +85°C), input voltage (3v to 5.5v), query time of 750ms, and just one digital pin.

B. Pressure Sensor - The range of pneumatic pressure sensors is 0 to 1mPa. Additional specifications include a 5v input voltage, 15mA operating current, and a pressure resistance range of 60k-110k ohms.

C. Ultrasonic Level Sensor (HC-SR04). The ultrasonic-HC-SR04 level sensor is a low-cost sensor that provides non-contact measurement functionality spanning from 2cm to 400cm (4m) with a ranging accuracy of 3mm, a working voltage of 5VDV, a working/operating current of 15mA, and a frequency of operation of 40Hz. It consists of a transmitter, a receiver, and a control circuit module. The HC-SR04 includes four PINS: VCC (power), trig (trigger), echo (receiving), and GND (ground).

D. Water Flow Sensor (HZ21WA) -The SEN-HZ21WA sensor is a small-size water flow sensor that is also easy to install, composed of high-quality materials, long-lasting, and requires no electricity. It has three lines, a test range of 130L/m3|h, a maximum working current of 15mA, a voltage of 15V, and a working pressure of 1.75mPa.

E. ESP32 - The ESP32 is a low-cost, low-power microcontroller with built-in Wi-Fi and Bluetooth that replaces the ESP8266, another low-cost Wi-Fi Microchip. The ESP32 has an advantage over the ESP8266 in that it does not require a microcontroller and an add-on, Wi-Fi, or Bluetooth module to develop linked objects. As seen in the block diagram below, this is the only chip you may require:

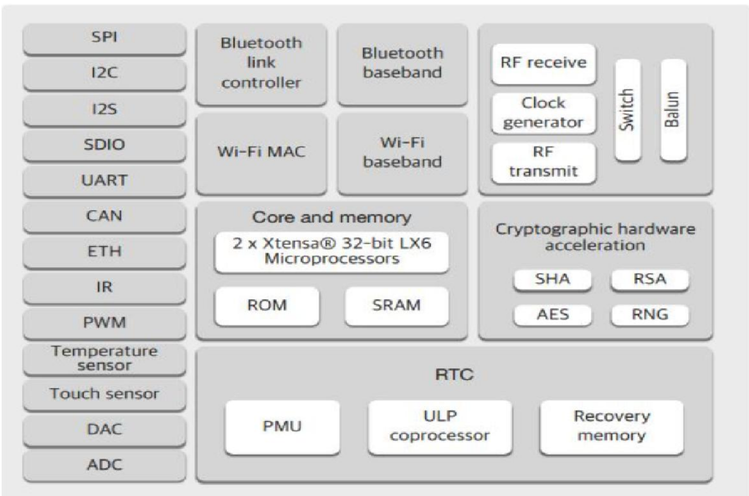

Fig 6. ESP32 block diagram (/wiki/file: featuresESP32.jPG)

The ESP32 has an Xtensa®Dual-Core 32-bit LX6 microprocessor that can run at up to 600DMIPS and can run on breakout boards and modules at speeds ranging from 160Mhz to 240MHz, which is an incredible speed for anything that requires a microcontroller with connectivity choices.

Protocol CPU (PRO CPU) and Application CPU (APP CPU) are the two (2).
The PRO CPU Processor handles Wi-Fi, Bluetooth, and other interval peripherals such as SPI, I2C, and others, whereas the APP CPU Processor handles application code. The ESP32 processor map is shown below:

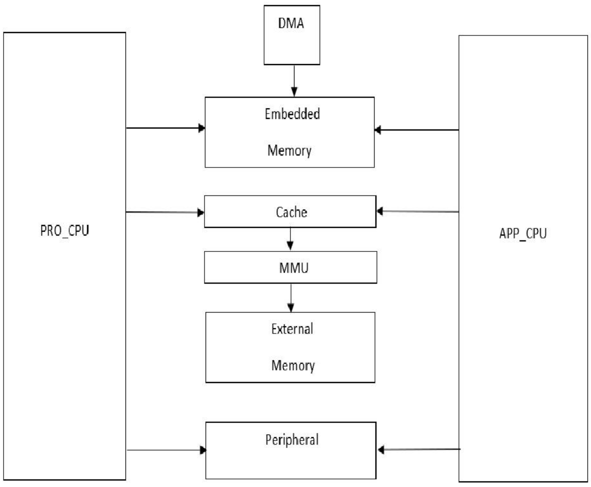

Figure 7: ESP32 Processor map (/wiki/file: ESP32_processor_map.JPG)

F. Wi-Fi

The Wi-Fi is part of the ESP32 chip and it is implementable with TCP|IP, full 802.11b/g/n/e/i WLAN MAC Protocol, and Wi-Fi Direct specification. Another specification of the ESP32 is the Wi-Fi direct support which is a good option for peer-to-peer connection without the need for an access point.

G. Bluetooth

Wi-Fi and Bluetooth connectivity are important aspects of the ESP32. The ESP32 not only supports Bluetooth 4.2, which is Bluetooth Low Energy (BLE), but also Bluetooth classic, which means it can communicate with both old and modern Bluetooth phones/tablets. The ESP32 Bluetooth Radio supports all class-1, class-2, and class-3 transmit output levels, as well as a dynamic control range of more than 30 dB.

H. MQTT-MQ Telemetry Transport

MQTT is a low-bandwidth publish/subscribe communications protocol designed for M2M (Machine to Machine) telemetry. MQTT is an abbreviation for MQ Telemetry Transport, which was previously known as Message Queuing Telemetry Transport. The MQTT protocol is low-energy and simple to implement for a wide range of devices. MQTT in IoT employs QoS levels to assure message delivery to recipients even when connections between them are unstable.
The diagram below depicts the basic architecture of a MQTT and its components.

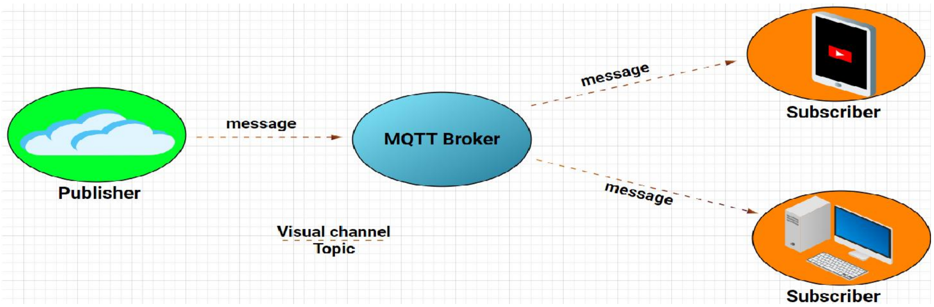

Figure 8: MQTT Architecture.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 8, April 2023

1. Message

The message is the data transmitted or received across the network for the application, and it has the following parameters:

Payload

Quality of Service (QoS)

Collection of properties and services

Topic Name Client

The subscriber and publisher roles are the two (2) fundamental functions of an MQTT communication protocol. The client subscribes to the subject in order to receive and publish messages. That is, an MQTT-enabled device is a client. If a device makes a network connection to the server and publishes messages for clients, subscribes, receives, and unsubscribes to messages, and shuts network connections to the server, it is considered a client. As a result, the client in MQTT executes two processes:

Publish: This process occurs when the client transmits data to the server.

Subscribe: When the client receives data from the server, this is referred to as a subscription action.

2. Server (Broker)

The Broker or Server is a device or program that allows the client to publish and subscribe to information.

3. Communication. The server accepts the network connection, and client messages handle subscribed and unsubscribed requests, forwards application messages to the client, and terminates the client's network connection.

4. The topic is the label assigned to the message and is checked against the server's subscription.

5. Mobile App: The mobile phone displays the server output in the serial monitor linked by a USB cord. The MIT App Inventor is used to construct the mobile App. MIT App Inventor is a cloud-based, visual programming environment that allows users to create fully working apps for Android phones, iPhones, and Android/IOS tablets. It is a block-based framework that allows you to create complicated, high-impact apps in much less time than standard programming environments.

I. Software Requirements

The following software was used to successfully construct this effective oil well monitoring system: ESP32/Arduino IDE: Coding, 3D Builder: Prototype design, and Circuit Design: Capital ElectraTM / Radical software.com/app/edit/demo

MIT APP Inventor: Mobile App Development

J. Circuit Diagram.The circuit diagram and detailed system connection are shown in the image below. It is created using "Capital ElectraTM / Radical software.com/app/edit/demo." Which is an online circuit design software.

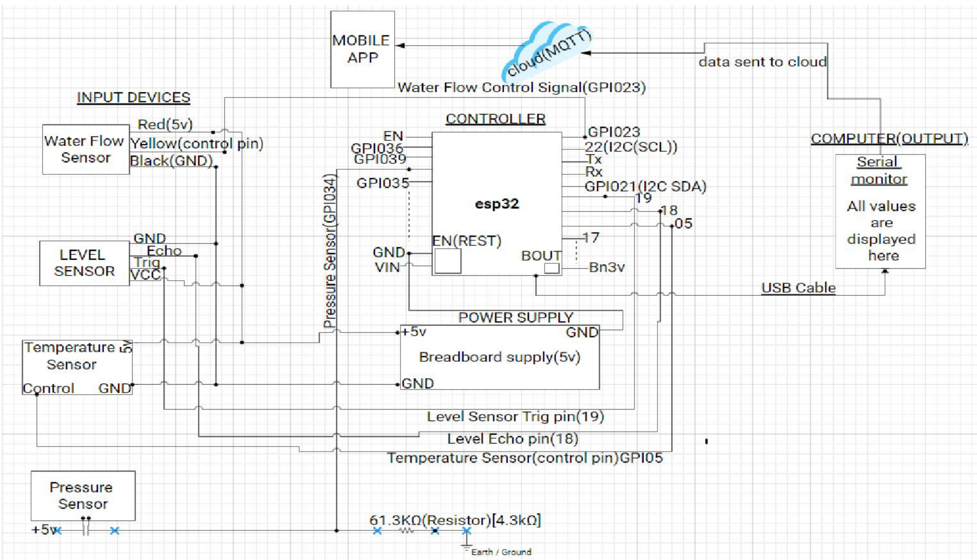

Figure 9: Circuit implementation diagram of the system

VI. RESULTS AND DISCUSSION

Assembling of the prototype

The system box with its components is described in the architecture as shown below in figure 10 below.

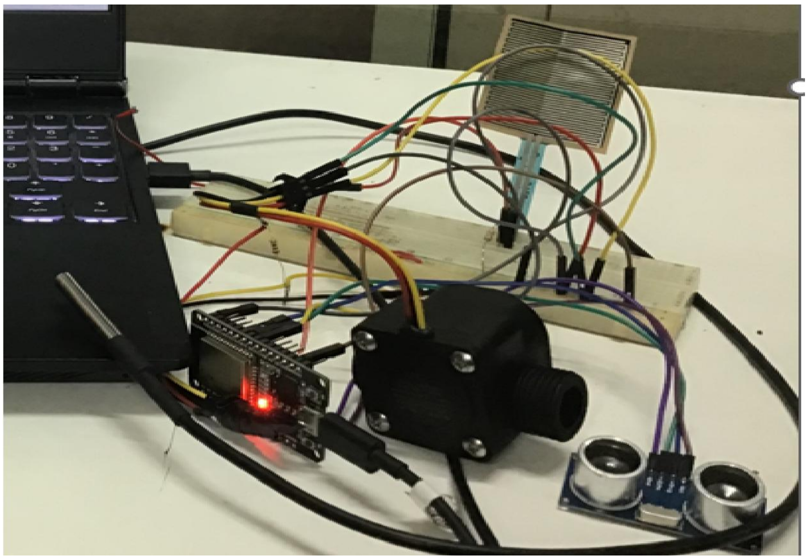

Figure 10. System box with components

System Test

To receive the test results, the subjects are added to the test.

(a) Pressure data was obtained by pressurizing the pneumatic pressure sensor with air. The magnitude of the pressure readout is determined by the amount of force used. The well pressure, on the other hand, is generated naturally by the reservoir pressure, allowing crude oil to flow from the reservoir to the surface. Crude oil is made of oil, water, and gas.

(b) To imitate the temperature, a hot soldering gun was held close to the non-contact sensor. The readings received are affected by the proximity of the soldering gun to the sensor. The temperature of the well is natural and relies on the well characteristics and field location.

(c) The level sensor detects the levels of field equipment, particularly separators. This is simulated by pouring fluid into a graduated pipe and taking a level reading, the value of which is referenced to the total capacity.

(d) The flow sensor simulates the flow rate of an oil well. This is determined by the amount of crude oil in relation to time. As a result, in the laboratory, the flow rate is measured in litres per minute, but crude oil is measured in barrels per day (bbl/day). Additional units used in comparison to free values are Pascal (Pa) for laboratories and Pound Per Square Inch (PSI) or Bar for the pitch. Temperature is measured in degrees Celsius (°C) in the laboratory and in degrees Fahrenheit (°F) in the field. The unit for the level sensor is %, and depending on the mode of graduation, numerous different units are employed.

The readings are as indicated in the figures below:

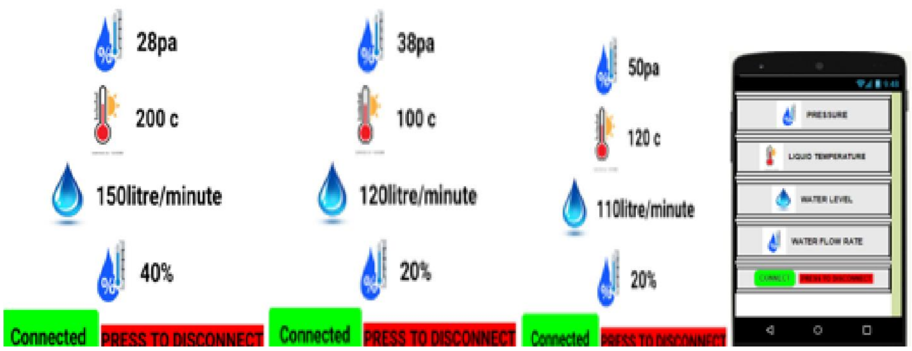

Figure 11: Readings

DOI: 10.48175/IJARSCT-9563

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 8, April 2023

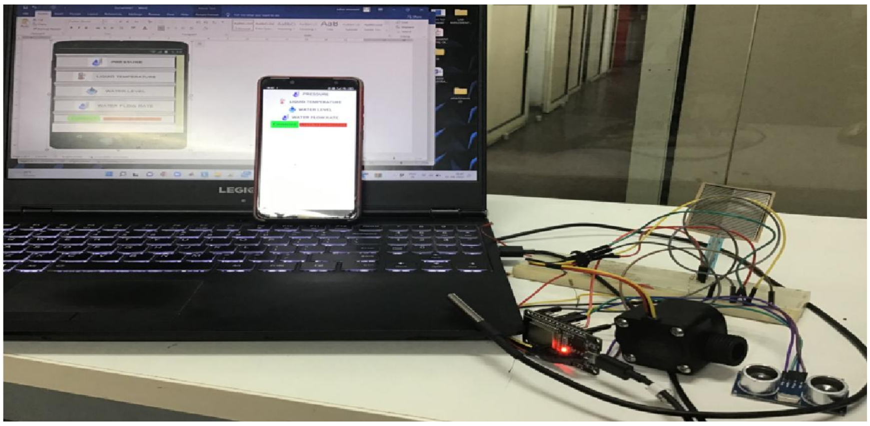

Fig 12:System box with Experimental result displayed on MIT APP & Laptop

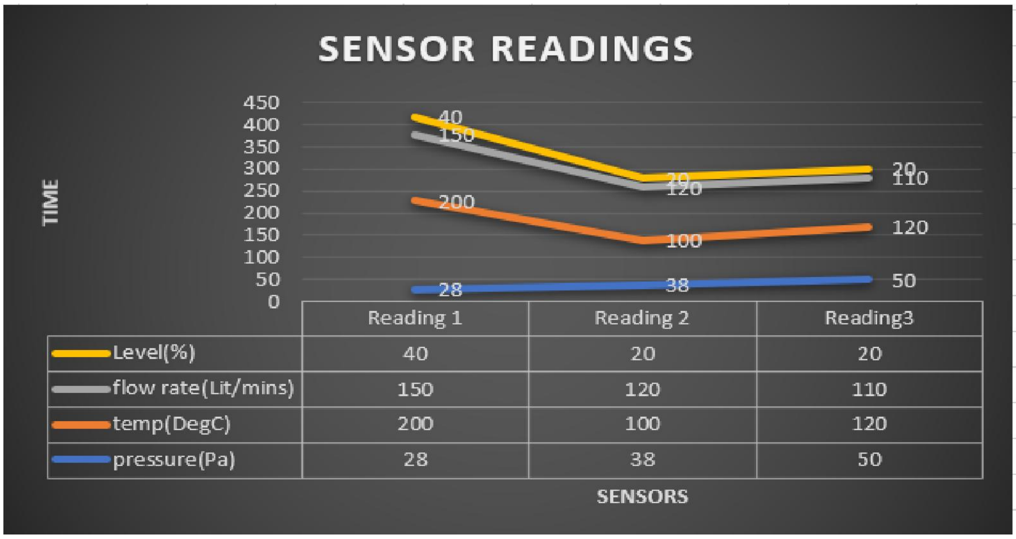

Figure 13: Plots of Results.

VII. CONCLUSION

Around ninety-five percent of Nigeria's petroleum reserve is guarded in the region of Nigeria known as the Niger Delta. Insecurity, on the other hand, has emerged as a significant obstacle to the efficient operation of those multinational firms; as a result, the use of remote operations has become unavoidable. Therefore, the application of intelligent design and deployment will reduce the incessant shutdown and kidnapping of company staff because the majority of the operational personnel will not be needed on site, and ultimately the reduction of the OPEX and CAPEX, which is the key business objective, which is to make a profit. In addition, the application of intelligent design and deployment will reduce the incessant shutdown and kidnapping of company staff.

REFERENCES

[1]. A. Yashin, A. Konev, and M. Khakimyanov, "'Smart Well' Concept in Oil Production," 2022, pp. 403–414.

[2]. M. R. Konopczynski and A. A. Ajayi, "Design of Intelligent Well Downhole Valves for Adjustable Flow Control," 2004.

[3]. J. Wosowei, C. Shastry "Domain-Specific Modeling for Monitoring Operations in a Smart field," Int. J. Innov. Res. Technol., 2022.

[4]. S. N. S. Al-Humairi, D. R. Tarshen Ravindran, M. I. Abdullah, B. Hamzah, and M. H. Alkawaz, "Intelligent Monitoring System for Oil Well Efficiency," in 2020 16th IEEE International Colloquium on Signal Processing & Its Applications (CSPA), 2020, pp. 13–17, doi: 10.1109/CSPA48992.2020.9068668

[5]. D. A. Neeraj, S.O Kulkarni, "Intelligent Portable Data Acquisition System with Predictive Features," Int. J. Innov. Res. Comput. Commun. Eng., vol. vol.3 I, no. ssue 5, 2015, doi: 10.15680/ijircce.2015.0305100.

[6]. M. Y. Aalsalem, W. Z. Khan, W. Gharibi, and N. Armi, "An intelligent oil and gas well monitoring system based on Internet of Things," in 2017 International Conference on Radar, Antenna, Microwave, Electronics, and Telecommunications (ICRAMET), 2017, pp. 124–127, doi: 10.1109/ICRAMET.2017.8253159.

[7]. A. Ibrahim and A. Ibrahim, "Using ZigBee for Wireless Remote Monitoring and Control," 2015.

[8]. M. Al-Fadhli and A. Zaher, "A Smart SCADA System for Oil Refineries," in 2018 International Conference on Computing Sciences and Engineering (ICCSE), 2018, pp. 1–6, doi: 10.1109/ICCSE1.2018.8373996.

[9]. Z. Y. Zhong, X. Li Zhi, and W. J. Yi, "Oil Well Real-time Monitoring With Downhole Permanent FBG Sensor Network," in 2007 IEEE International Conference on Control and Automation, 2007, pp. 2591–2594, doi: 10.1109/ICCA.2007.4376830.

[10]. R. Barani and M. V. J. lakshmi, "Oil Well Monitoring and Control Based on Wireless Sensor Networks using Atmega 2560 Controller," 2014