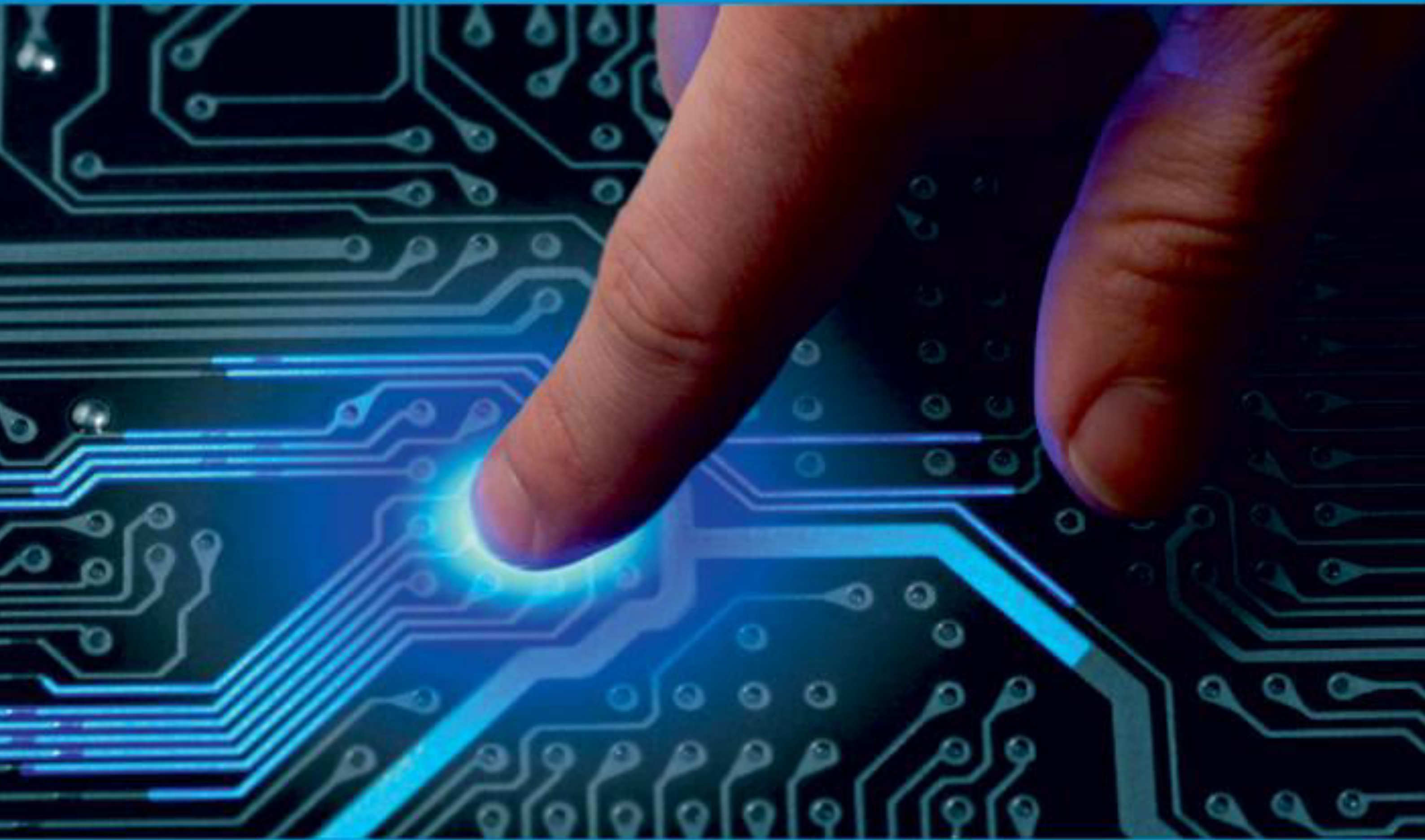
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 12, Issue 3, March 2024

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.379

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 3, March 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1203179 |

Comparative Analysis of HART and MODBUS Communication Protocols

Kaustubh Chavan[1], Soham Methul[2]

Department of Electronics and Telecommunication Engineering, BRACT's Vishwakarma Institute of Technology, Pune, India[1]

Department of Electronics and Telecommunication Engineering, COEP Technological University, Pune, India[2]

ABSTRACT: Communication protocols are crucial in the field of industrial automation and control systems for providing reliable and efficient data flow between equipment. HART and MODBUS are two of the most commonly used communication protocols that have become renowned for their unique functions and uses. An extensive comparison of the HART and MODBUS communication protocols is presented in this research article. The present research seeks to support engineers, researchers, and decision-makers in making educated decisions when choosing the suitable communication protocol for certain industrial applications by in-depth analysis of various protocols. This paper deals with a detailed study of these two following protocols.

KEYWORDS: Highway Addressable Remote Transducer (HART), MODBUS, communication protocols, industrial automation.

I. Introduction

Strong communication solutions are required as a result of the widespread use of automation technologies in industrial processes in order to enable flawless data transmission between various devices, sensors, and control systems. Due to their interoperability, dependability, and extensive acceptance, HART and MODBUS have become contenders among the wide range of communication protocols available. These two protocols are renowned for their use in automation industry and are frequently used in this sector. In order to shed light on their relative merits and comparative strengths and shortcomings, this research explores the technical details, practical applications, and performance of these protocols in the setting of a chemical processing facility.

II. Literature Review

Aishwarya Gunjal, et.al addresses the design of a distributed measurement system implementing HART communication that can measure and track variables in industrial processes including temperature, pressure, and voltage. This system makes use of sensors, multiplexers, analog-to-digital converters, microcontrollers, and an LCD display with graphics. The HART interface for communication and the possibilities for wireless loop current transfer are also mentioned in the article. For industrial control applications, the system may be coupled with controllers like PLCs or DCS. [1]

Cristian Patrascioiu, et.al proposes experimental design and development of a pH monitoring system that makes use of HART communication using LabVIEW software. Also discusses the pH measurement theory, the pH transducer's construction, and the programming and testing of PACTware software. Also describes the modules used in the pH monitoring programme and the LabVIEW programming environment. In addition to outlining the value of pH adjustment and measurement in industrial processes, the document also discusses the advantages of utilising LabVIEW software for control and monitoring. [2]

LiuChang, et.al designs an intelligent temperature transmitter based on the HART protocol in this study. Author describes the transmitter's hardware and software architecture, as well as its sensor, MCU, HART communication, HART modem, and power modules. The author also covers communication programme and measurement programme of the transmitter. According to this, HART protocol serves as a dependable means of communication for process control devices. The automated temperature emitter uses the HART protocol to connect to the control room and transform physical signals into quantifiable electrical data. [3]

Wenzhu You, et.al uses Modbus protocol to develop and construct an intelligent building system. The system is designed to gather and keep track of a variety of information, including temperature, humidity, gas concentrations, and infrared detection of human body heat in a structure. An integrated Modbus gateway is part of the design, and it uses

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 3, March 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1203179 |

the Modbus TCP protocol to link the Modbus field bus and Ethernet. The system also has a user-friendly online console and a three-layer approach for applying the Modbus TCP and Modbus RTU protocols effectively. The document includes information on the system analysis, hardware, and software components. [4]

Su Ying, et.al explains use of the Modbus protocol for data transmission and communication between a master and slave system. Author explains the process of generating a checksum code using the CRC-16 standard and attaching it to the binary code sequence. Also covers the design of the slave system software, including the main program and interruption service procedures. The experimental platform and results are analyzed, highlighting the successful implementation of the protocol control system based on a single-chip microcomputer. [5]

Kelong Wang, et.al explains the Modbus communication protocol's implementation using an ARM Cortex-MO CPU. Focuses on enhancing the real-time and dependability of a power plant's information gathering system. Modbus protocol's structure, communication settings, and the hardware architecture of the data collecting system are all explained in the document. The Modbus RTU protocol is robust and quick, making it important in the power industry, and offers an illustration of system communication. The ARM Cortex-MO data collecting system effectively interfaces with the master station (PC) and extracts parameters from the connected device. [6]

Dr. Ing. Karel Dudáček addresses design and implementation of a communication module using the HART protocol for industrialised process control and automation. A 4-20 mA current loop powers the module, and an insulated I2C interface connects it to the instrumentation equipment. The FreeRTOS real-time OS is utilised in the software development, and the program is divided into different jobs for sharing information. Details on the hardware parts, the HART protocol, power management strategies, and programming strategy utilised in the implementation are provided. [7]

Cristian Patrascioiu, et.al addresses experimental research on pH measurement and monitoring system design. The paper is divided into three sections. The transducer construction, the adaptor that is being utilized, and the principle of measuring the pH of a solution substance are all covered in the first section. The second section focuses on the experimental study that was conducted with the goal of configuring and verifying the pH transducer's operation utilizing the HART protocol through the usage of the PACTware software environment. [8]

Geng Yue covers the use of the Modbus protocol and how to combine the two types of Modbus protocols to create a double-layer monitoring network based on the Modbus/RTU and Modbus/TCP protocols for a super multiport-saving equipment system. After examining the features of Modbus/RTU and Modbus/TCP as well as their potential applications, the article builds and deploys a single-layer network based on Modbus/RTU that enables the control and monitoring of a limited number of equipment groups. Then came the introduction of the second layer network, which was based on Modbus/TCP. All node groups were controlled thanks to this network's high transmission rate. [9]

III. HART PROTOCOL

Analog signalling and digital communication are both used in this communication approach. It is frequently employed to add digital communication capabilities to analogue equipment after the fact. The master-slave design of HART allows for the simultaneous transfer of digital data and analogue process variables.

This protocol was created in the late 1980s to allow digital communication with field equipment that was traditionally analogue. By overlaying digital interaction on the current analogue stream, this hybrid protocol enables bi-directional communication. Process industries, where sensors and actuators are dispersed across long distances, are the main focus of HART.

Approaches used for HART Protocol:

1. **Bell 202 Frequency Shift Keying (FSK): -** This technique represents binary data using two distinct frequencies. In the HART protocol, a frequency of 1200 Hz corresponds to a "0" and a frequency of 2,200 Hz to a "1". It offers strong noise immunity and works with a variety of communication mediums, such twisted pair, coaxial, and fiber optic cables, the Bell 202 FSK technique is frequently employed in HART devices.

2. **Manchester Encoding: -** Through the use of a signal transition in the midst of each bit time, binary data is encoded using this technique. A "0" bit is represented by a high-to-low transition, while a "1" bit is represented by a low-to-high transition. Some HART devices employ Manchester encoding because it offers good synchronisation and is less prone to phase distortion. Digital signals are encoded using any of these two techniques at the physical layer before being transmitted through the communication media. In conclusion, the physical layer of HART is in control of transmitting digital data.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 3, March 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1203179 |

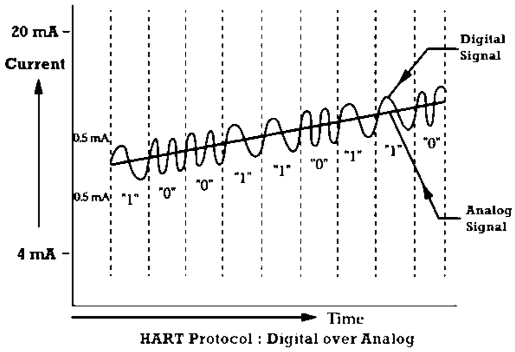


Fig. 1. FSK Signal riding on 4-20mA signal [10]

Network Topology supported by HART:

1) Point-to-point topology: - The master device and only one field device are linked together via a direct, dedicated link within a point-to-point HART architecture. This setup mimics a one-to-one communication connection where the master speaks solely to one particular field device. When a single field device, such a sensor or actuator, has the ability to communicate with a control system and a monitoring station directly, it is often used in those situations. This setup is appropriate for instances in which managing the process using a single device is feasible.
2) Multidrop topology: - A single HART master device is connected to a number of field devices in a multi-point HART topology. The master can communicate with and acquire information from a number of field devices using this setup across a shared network. In industrial applications where several sensors, transmitters, or additional field devices must be controlled and managed by a centralised master device, the multi-point HART architecture is often utilised. When it's essential to gather and analyses data from various devices in order to make informed choices about the industrial procedure, this setup is ideal.
3) Wireless topology: - The HART protocol may be extended to provide wireless communication between field devices and a receiver or central gateway. It is intended to deliver the advantages of HART communication without the requirement for actual wires, which can be especially helpful in circumstances where running wires is difficult or expensive. When wiring is either too expensive or impossible, wireless HART is a good solution. This covers situations when remote sensors may need to be installed in difficult or remote places, such as in the oil and gas industry.
4) Hybrid topology: - In a hybrid HART architecture, both wireless and wired communication are integrated in one network. This enables clients to take use of both the advantages of wired and wireless connection depending on the particular needs of their industrial operations.

In cases when a combination of both wireless and wired communication is needed, this design is especially advantageous. In a big industrial facility, the majority of devices might interact via wired HART, but to broaden the network's coverage for distant or mobile equipment, wireless HART can be used.

Types of HART Commands:

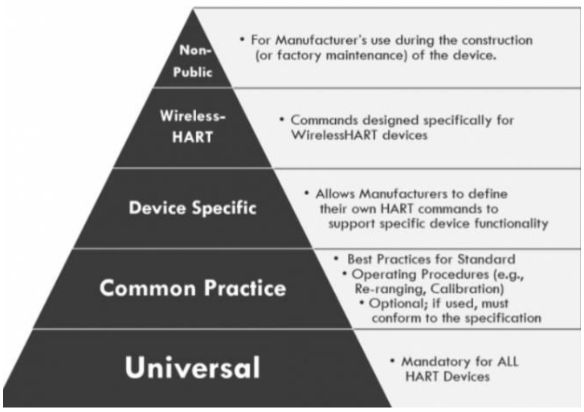


Fig. 2. HART communication protocol flowchart [12]

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 3, March 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1203179 |

1. Universal commands:
   - Check the device type and manufacturer.
   - Examine the units and principal variable (PV).
   - View the production percentage and current output.
   - Up to 4 pre-defined dynamic parameters can be checked.
   - Read or write the date, a 16-character descriptor, and an eight-character tag.
   - Read or type a message with 32 characters.
   - Examine the damping time constant, units, and device range values.
   - Final assembly number should be read or written.
   - Configure a polling address.

2. Common Practice commands:
   - Select up to four dynamic variables to read.
   - Put the damping time constant in writing.
   - Write the values of the device range.
   - Adjust (set span, set zero).
   - Establish a constant output current.
   - Take a self-assessment.
   - Carry out the master reset.
   - Reduce PV to zero.
   - Put PV unit in writing.
   - Reduce DAC zero and increase.
   - Write a linear or square root transfer function.
   - Put the serial number of the sensor here.
   - Write or read assignments with dynamic variables.

3. Device specific commands:
   - Read or write low-flow cut-off.
   - Start, stop, or clear totalizer.
   - Read or write density calibration factor.
   - Choose PV (mass, flow, or density).
   - Read or write materials or construction information.
   - Trim sensor calibration.
   - PID enable.
   - Write PID set point.
   - Valve characterization.
   - Valve set point.
   - Travel limits.
   - User units.
   - Local display information.

Features of HART protocol:

1. Open Standard: - It's a publicly accessible protocol, which entails that the public can view its specifications and standards. Such openness motivates developers to create HART-compatible products while following accepted protocols, guaranteeing a certain amount of uniformity in how devices work together.
2. Universal Commands: - These commands include Write Configuration, Read Device Identification, and Read Current Value, among others.
3. Device Description Language (DDL): - The characteristics and capabilities of HART systems are described using a standardized DDL. Device-specific files that detail a device's characteristics and capabilities are produced by device makers using DDL.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 3, March 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1203179 |

4. Registered Devices: - The certification and registration of HART devices is administered by the HART Communication Organization. For the purpose of making sure their products adhere to HART protocol standards, manufacturers submit their products for testing. Within HART networks, devices that have been registered and verified are guaranteed to function properly.
5. Device Type codes: - Different types of field equipment is given distinctive device type identification numbers by HART. These identifiers ensure that identical devices, even those made by different manufacturers, are recognised and work together.
6. Common Practice commands: - There are a number of instructions in HART that are generally supported by HART devices. These instructions make crucial operations like calibration, diagnostics and configuration possible, etc.
7. Testing: - Manufacturers of HART devices frequently test their products' compatibility with those made by other manufacturers in order to assure interoperability. Any potential compatibility problems are found and fixed as a result of this testing.

Integration with control system: - Distributed control systems, programmable logic controllers, or SCADA systems incorporate HART devices. Because of HART compatibility, data from these devices may be smoothly merged into the larger control system, increasing the reliability and efficiency of the entire process.

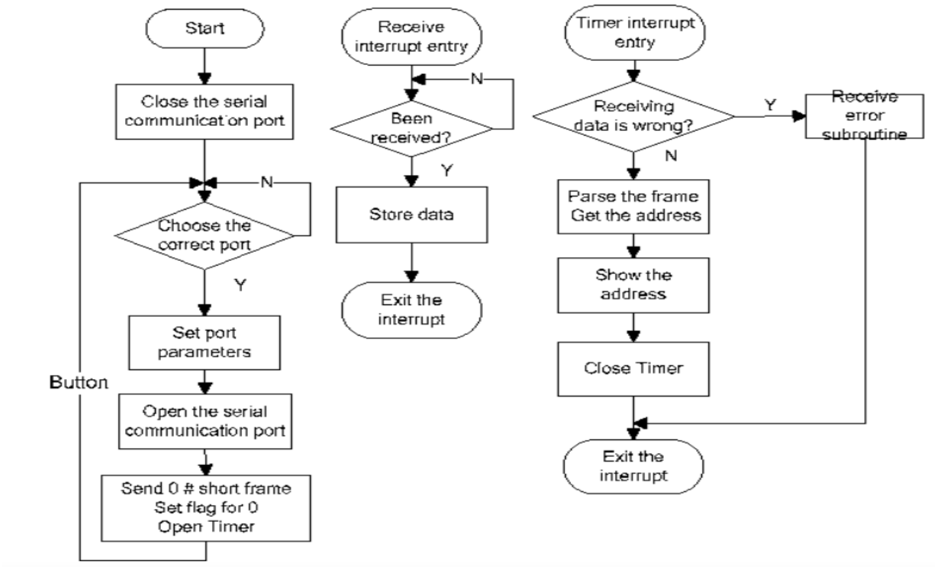


Fig. 3. HART communication protocol flowchart

In HART communication protocol updates are required for several things, such the operation monitoring screen's SP, PV, and MV numbers display. Some device information, such the device descriptor in the device information page, the device station number, and basic device information, do not need to be updated. The HART update frame is the program flow of the HART frame that doesn't require updating. The user inputs the screen update, activates Timer 1, and sets the flag, which is the command number for the initial frame. Timer 1 enters the Timer 1 interrupt after reaching the point. The flag number provides the command, which is subsequently packaged as the appropriate effective HART frame. Subroutine Timer 1 interrupt transmits a HART frame module. The HART communication startup and the receive interrupt subroutine share the same design. The receive interrupt must end before Timer 2 reaches the interval, or else the receiver is invalid and a new message must be sent. Extract data and convert numerical to display if checking is accurate. Next, shut Timer 2, open Timer 1, put the command number of the subsequent HART frame to the flag, and end the interrupt. Send the following HART frame after Timer 1 reaches its point. Changing several Timers allows achieve looping by sending multiple HART messages and presenting updated information.

Advantages of HART:

- Compatibility
- Bi-Directional Communication
- Enhanced Diagnostics
- Reduced Wiring

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 3, March 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1203179 |

- Increased Accuracy
- Easy Configuration
- Safety

Disadvantages of HART:

- Limited Bandwidth
- Limited Range
- Compatibility Issues

HART-IP: - In 2012, HART-IP was incorporated into the HART specification. In an installation with multiple devices and a host communicating with them all, the host must be aware of the device's name or address in order to send a command to it. The HART protocol standard defines this addressing method. It is no longer necessary to use the addressing system specified by HART when the internet protocol, or IP, is integrated into the device. Instead, each device is given an IP address through the use of IP-addressing.

IV. MODBUS PROTOCOL

MODBUS is a popular protocol in industrial automation that was created in 1979 by Modicon. It was initially intended to facilitate communication between field equipment and programmable logic controllers.

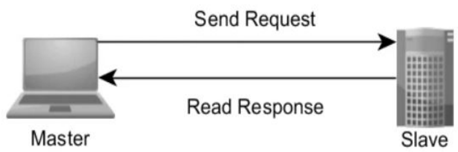


Fig. 4. MODBUS communication protocol [11]

Figure 4 shows MODBUS communication which is a straightforward and widely utilized communication protocol that is frequently used to link devices in systems for industrial automation. It is available in variations including MODBUS RTU and MODBUS ASCII, each designed for certain kinds of communication channel.

MODBUS PDU (Protocol Data Unit):

The Modbus Application Protocol Specification is centered on the PDU and the programming that manages it. The definition of the Modbus PDU format is a function code followed by a collection of related data. The function code determines the size and content of this data, and the total PDU (data + function code) cannot be larger than 253 bytes. Slaves can implement each function code's individual behavior in a flexible way according to the application behavior they want. Fundamental ideas for information access and manipulation are described in the PDU standard; however, a slave can handle data in a fashion not specified in the specification.

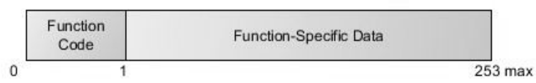


Fig. 5. PDU [13]

MODBUS ADU (Application Data Unit):

You can employ several network protocols in addition to the Modbus protocol's PDU core capabilities. Modbus comes with an array of ADU variations that are specific to each network protocol in order to transfer the data required for Modbus across these layers.

Certain characteristics must be present for Modbus to offer dependable communication. In every ADU format, the Unit ID or Address is utilized to supply the application layer with routing information. Every ADU is accompanied by a complete PDU that contains its function code along with associated information for a specific request. Every message has error-checking information for dependability. In conclusion, every ADU offers a method for identifying the start and finish of a request frame, but with varying implementations. There are three common ADU formats: ASCII, RTU, and TCP.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 3, March 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1203179 |

1. TCP/IP ADU

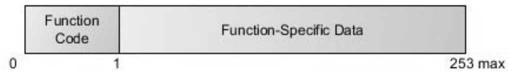


Fig. 6. TCP/IP ADU [13]

The Modbus Application Protocol (MBAP) Header and the Modbus PDU are concatenated to form the TCP ADUs. The general-purpose header known as MBAP is dependent on a dependable networking layer. It contains a transaction identifier first. This is useful on a network wherein multiple requests may be waiting at once. That is, requests 1, 2, and 3 can be sent from a master. Further on, a slave will be able to reply in the sequence 2, 1, 3, and the master will be able to accurately analyze data and match requests with responses. The protocol id usually starts with zero, though it can be used to increase the protocol's functionality. The protocol itself uses the length field to specify how long the remaining portion of the packet is. This element's placement also reveals how dependent this header format is on a dependable networking layer. Packet length can be found anywhere in the header since these packets contain built-in error checking to guarantee data coherency and delivery. TCP offers a fair level of defense against this circumstance. Usually not used for TCP/IP devices, the Unit ID is used to identify the location of the slave device for which the PDU is truly designed. There is a PDU in the ADU. For the standard protocol, this PDU can still be limited to 253 bytes wide.

2. RTU ADU

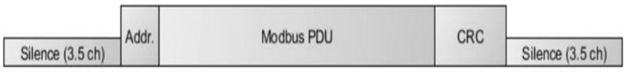


Fig. 7. RTU ADU [13]

Apart from the primary PDU, this ADU only contains two additional pieces of data. Initially, a PDU's address determines which slave it is meant for. The "broadcast" address on most networks is defined as an address of 0. In other words, all slaves should process a request sent by a master to address 0, but no slave should reply. In addition to this address, a CRC is employed to ensure the data's authenticity. Two silent periods, or instances when there is no interaction on the bus, are what separate the packet. The pace is around 4 milliseconds for the baud rate of 9,600. Regardless of baud rate, the standard specifies a minimum silence length of little less than 2 milliseconds. This comes with an efficiency cost as the device can't process the packet before the idle time is over. Taking off the layer of abstraction between the Modbus PDU and the networking layer is a popular solution to these problems. In other words, the function code is obtained by the serial code querying the Modbus PDU packet. If this is not used, code may get more "corrupt" packets than expected.

3. ASCII ADU

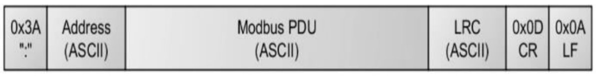


Fig. 8. ASCII ADU [13]

While the ASCII ADU is a bit more sophisticated than RTU, it also stays away from a lot of the problems with RTU packets. The ASCII ADU eliminates the problem of figuring out packet size by giving each packet a distinct and well-defined start and end. In other words, every packet starts with ":" and finishes with a line feed (LF) and carriage return (CR). Furthermore, data in a buffer can be read with ease by serial APIs such as NI-VISA and the.NET Framework SerialPort Class, up until a certain character like CR/LF becomes available. These characteristics facilitate the effective processing of the serial data stream in contemporary application code.

The ASCII ADU's drawback is that all data is transmitted as ASCII-encoded hexadecimal bits. In other words, for function code 3, 0x03, instead of sending a single byte, it sends the ASCII characters "0" and "3," or 0x30/0x33. This improves the protocol's readable character but requires sending and receiving programs to be able to decode ASCII values and transmitting twice as much data through a serial network.

Network topology supported by MODBUS:

Since the MODBUS protocol for communication is fundamentally straightforward and adaptable, it may be used with a variety of network topologies. For MODBUS, the following network topologies are supported:

1) Serial Line Configuration:

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 3, March 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1203179 |

1.1) Point to point: - When two devices are connected directly to one another in a point-to-point arrangement, generally a master and a single slave device, MODBUS can be used. This straightforward topology is frequently used to establish local connections.

1.2) Multi point: - A single serial communication channel, such as RS-485, is linked to several slave devices in a multi-drop setup. The master addresses each slave specifically while communicating with them. This architecture is employed in situations where it is necessary for several devices to share a single communication channel.

2) Ethernet Networks:

2.1) Star Topology: - A central switch or hub connects every device in a star arrangement in an Ethernet-based MODBUS network. Data is routed through the switch to reach its destination, and each device connects directly with the main switch. Modern industrial networks frequently use this design, which works well for bigger deployments.

2.2) Bus Topology: - In certain Ethernet-based MODBUS networks, equipment's are chained together in a bus-like arrangement, with each unit linked to the one before it. Until they arrive at the destination device, data packets go down the chain.

3) Wireless Network:

Utilizing Wi-Fi, Zigbee, or other wireless protocols, MODBUS may also be used in wireless networks. This is especially helpful when physically running wires would be difficult.

Features of MODBUS protocol:

1. Open Standard: - An open documentation communication protocol. The availability of its specifications and standards encourages developers to create MODBUS-compatible products while following established protocols. This openness guarantees that devices interact in a consistent manner.
2. Communication Modes: - Serial and Ethernet are only two of the many communication methods that MODBUS supports. This adaptability makes it possible to integrate into different network infrastructures, improving compatibility.
3. Register mapping: - The locations and functionalities of various data points are specified in a common register map, which is generally used by devices that adhere to the MODBUS standard. Devices can transmit and understand data reliably because to this mapping.
4. Function Codes: - Devices employ a defined set of function codes called MODBUS to ask other devices for certain actions to be taken or data to be provided.
5. Testing: - Manufacturers frequently assess the compliance of their MODBUS products to guarantee compatibility. These tests confirm that the devices meet the requirements of the MODBUS protocol and can efficiently communicate with other compatible devices.
6. Gateway devices: - Gateway devices that interpret among several MODBUS generations or connect MODBUS with additional industrial protocols for communication can help to improve MODBUS harmony. These gateway electronics facilitate the fusion of old and new technology.
7. Standard Data types: - Standard data types, such as 16-bit integers or 32-bit floats, are frequently used by MODBUS devices to represent data.
8. Integration with PLC- A lot of SCADA platforms, PLCs, and other control systems incorporate MODBUS devices. Within these systems, data interchange and control operations are made possible via MODBUS compatibility.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 3, March 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1203179 |

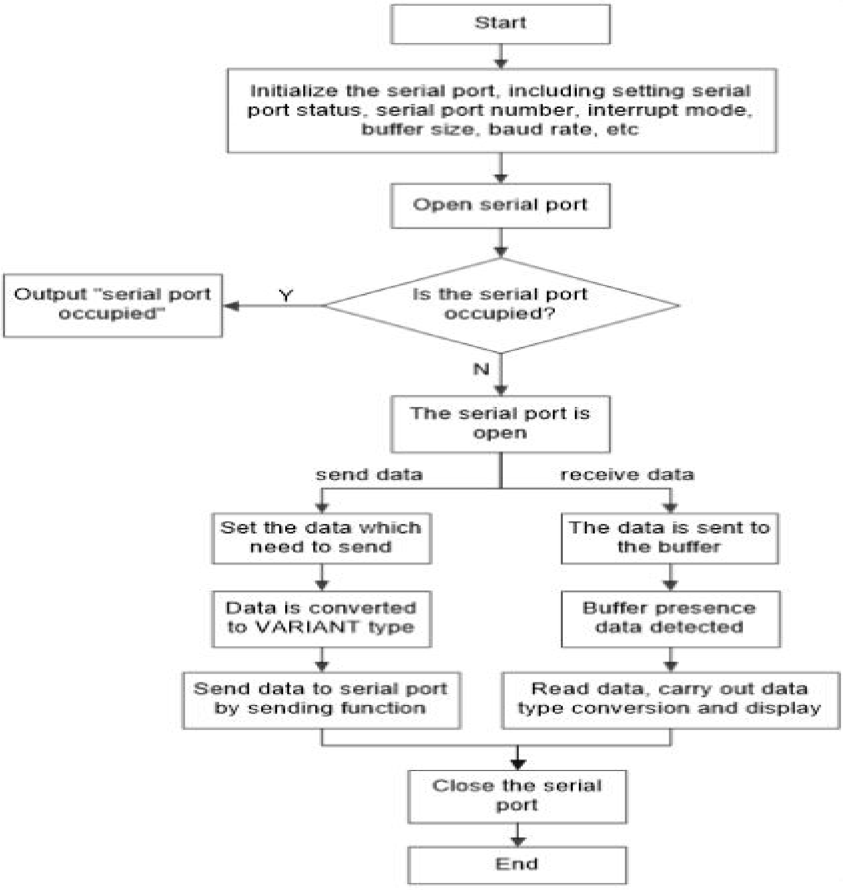


Fig. 9. MODBUS communication protocol flowchart

First, initialize the serial port. Employ the functions offered by the program control to configure the serial port's state and parameters in accordance with the apparatus utilized in the experiment. These parameters include the serial port's state, its number, its receiving mode, its ability to receive and send buffer sizes, it's baud rate and check method among other settings. If the serial port is found to be closed, it can be opened by sending an open serial port operation. If found to be open, an error will be reported. When transferring data, it is first input, then transformed into the VARIANT data type needed for serial communication, and finally delivered via the control function. When the device reads its status, it uses the 485 bus to send the status back to the set. When the program notices that there are characters in the buffer, it converts the received characters into a format.

Advantages of MODBUS:

- Simplicity
- Wide Adoption
- Versatility
- Efficiency
- Real-Time Control
- Scalability
- Open Standard

Disadvantages of MODBUS:

- Limited Data Types
- Limited Address Space
- Slow Speed in Serial Mode

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 3, March 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1203179 |

- Complex Configuration

No Acknowledgment of Data Delivery

V. CONCLUSION

Having its hybrid analog-digital design, HART excels in retrofitting older analog devices and is a good choice in situations where remote device management and backward compatibility are essential. Due to its ability to operate in dual modes, it can adjust with current structures, making it a great option for businesses in the chemical, pharmaceutical, and oil and gas sectors.

On alternate side, MODBUS stands apart due to its simplicity of use and adaptability. It allows for effective digital communication and is extensively used in many sectors, notably manufacturing, where it makes it easier to coordinate multiple systems and equipment.

Table I. Comparison of HART vs MODBUS protocol

PARAMETERS	**HART**	**MODBUS**
Communication type	Hybrid	Digital
Configuration	Request response and Burst mode	Master Slave only
Message Format	Digital signal superimposed on analog signal	Binary based function codes
Data Representation	Analog and Digital	Binary
Transmission medium	4-20mA analog signal	Digital lines or Ethernet
Topology	Single drop or Multidrop	Point to point or Star topology
Addressing	Every device has a unique address	Devices are addressed by unique identifiers
Diagnostic support	Built in diagnostics	Requires additional diagnostic support
Flexibility	Limited	Versatile and supports various device types
Applications	Process industries, Chemical plants, smart devices	Industrial automation, PLC systems, HVAC, etc.

From Table I, a clear differentiation of HART and MODBUS communication protocol can be observed. Both protocols are widely used in industrial automation field. The selection of the protocol depends on the user preference and compatibility of device designed/used.

REFERENCES

1. Gunjal Aishwarya, Suruchi Gaware, at.el, "HART Communication Based Distributed Measurement System", *IJRAR- International Journal of Research and Analytical Reviews*, vol 6, pp. 196-200, 2019.
2. Patrascioiu, Cristian, and Grigore Stamatescu. "Monitoring pH with HART communication." In *9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications,* vol. 2, pp. 864-869. IEEE, 2017.
3. Chang, Liu, and Zhang Guoguang. "The Design of Intelligent Temperature Transmitter Based on HART Protocol." IEEE *Second International Conference on Instrumentation, Measurement, Computer, Communication and Control*, pp. 1499-1502. 2012.
4. Yu Wenzhu, and Haibo Ge. "Design and implementation of modbus protocol for intelligent building security." *IEEE 19th International Conference on Communication Technology*, pp. 420-423. IEEE, 2019.
5. Ying Su, at.el "Research on Modbus Bus Protocol Implementation Technology Based on Single Chip Microcomputer." In *2018 3rd International Conference on Information Systems Engineering*, pp. 127-131. IEEE, 2018.
6. Wang Kelong, at.el. "Implementation of Modbus communication protocol based on ARM Coretx-M0." In *2014 IEEE international conference on system science and engineering*, pp. 69-73. IEEE, 2014.
7. Dudáček, Karel, Jiří Ledvina, and Vlastimil Vavřička. "Development of a HART compatible HART line powered communication module." *International Conference on Applied Electronics*, pp. 75-78. IEEE, 2014.
8. Patrascioiu, C., & Stamatescu, G. "Monitoring pH with HART communication", *9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications,* (Vol. 2, pp. 864-869), 2017.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 3, March 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1203179 |

9. Yue Geng. "Design of intelligent monitoring and control system based on modbus",*5th International conference on communication, image and signal processing*, pp. 149-153. IEEE, 2020.
10. https://i.pinimg.com/736x/97/18/12/9718122b5b1b818c4569386d9d2fe1ae.jpg
11. https://ni.scene7.com/is/image/ni/fdatawyg5871526514498411034?scl=1
12. https://www.fieldcommgroup.org/sites/default/files/styles/coh_medium/public/imce_files/technology/images/HART_commands_diagram.png?itok=aMdtYkWr
13. https://www.ni.com/en/shop/seamlessly-connect-to-third-party-devices-and-supervisory-system/the-modbus-protocol-in-depth.html#:~:text=Modbus%20is%20a%20request%2Dresponse,responsible%20for%20initiating%20every%20interaction