IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.047 ⁜ Vol. 10, Issue 5, May 2022

DOI: 10.17148/IJIREEICE.2022.10554

IOT Based Expandable Multi Stepper Motor Control System On CAN Bus

Mr. Yogesh R. Chauhan[1], Prof. Tejas Patel[2]

[1]P.G Scholar, Electrical Department, LDCE, Ahmedabad, Gujarat, India

[2]Assistant Professor, Electrical Department, LDCE, Ahmedabad, Gujarat, India

Abstract: The modern control applications require the simultaneous or individual controlling of multiple stepper motors which can be achieved by using the CAN bus. A CAN bus communication established using ESP-32's Controller Area Network (CAN) controller and TJA1050 CAN transceiver. The entire system suggested here can be expanded by connecting an individual stepper motor controller to an existing system as needed or for future development via a CAN bus link. Using ESP32 IOT capabilities, developed industrial remote monitoring and control system sets motion control parameters of each stepper motor and controls each stepper motor's position. The proposed system is compatible and capable of controlling any actuator. The whole arrangement is presented with the help of software and hardware design. The proposed approaches appear to be promising based on the experimental results.

Keywords: ESP-32, CAN bus, TWAI, TJA1050 CAN transceiver, ESP-IDF, Web Server, IOT

I. INTRODUCTION

For systems such as Multi-Head Weighers, real-time synchronization of multiple stepper motors is essential. The Controlled Area Network (CAN) protocol allows devices to communicate effectively with one another. Thanks to the excellent connectivity, modern IoT devices allow data transmission and remote communication to be reliable. The ESP32 device is a powerful microcontroller with built-in Wi-Fi and Bluetooth, making it ideal for IoT applications. The WiFi connection and CAN port on the ESP32 Wroom board will be used to connect to a real CAN network and send messages to an external device connected to the microcontroller.

This dissertation work originated from the requirement to synchronise several motors and expand the existing CAN network for future development. Taking this need into account, the dissertation work explains the path for the actual implementation of the CAN network.

The thesis intends to create a network between several microcontrollers and control each slave using a web-server that has been developed. The scope of the dessertation work can also be extended to individually control or synchronise any actuator. The ESP32 will act as a network master, directing other microcontrollers to switch on/off or adjust the rotation angles of stepper motors. The ESP32 will function as a station point, enabling it to connect to an existing Wi-Fi network.

The main goal of the project is to demonstrate how an ESP32 can be used to broadcast or receive data from a CAN Bus Network. The CAN Port on the ESP32 Wroom board will be used to connect to a CAN network and transmit messages to a microcontroller-connected external device. The ESP32 hosts a web server that accepts user input.

II. LITERATURE SURVEY

Traditional stepper motor control systems typically operate a single stepper motor, which cannot solve the problem of many stepper motors functioning in the same system, which is frequently required in modern industry [1]. Because stepper motors provide accurate positioning and open-loop control, they have been widely employed in the domains of electronics, metallurgy, aerospace, and military industry [2]. Many researchers have proposed various solutions to this dilemma. WangGaogao [3] proposed multi control system based on CAN bus, which achieves real-time, synchronous, and high precision drive control of numerous stepping motors using the CAN bus architecture. Ma Dandan [4] proposes a smart stepping motor drive control module based on CAN-BUS and STM32. The system can effectively realise the drive control of the distributed motor intelligent module, considerably improving the system's accuracy and dependability. Chai Wenfeng [5] proposes the design of a CAN interaction happens network based on STM32 and MCU, which makes full use of the characteristics and advantages of the CAN bus to enable data communication between forty nodes. CAN BUS is a vital network transmission technique used in the car network communication area due to its real-

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.047 ⁜ Vol. 10, Issue 5, May 2022

DOI: 10.17148/IJIREEICE.2022.10554

time, reliability, and adaptability. The CAN-bus based communication system for vehicle automation is designed in this study. Both the software and hardware systems are easily expandable and upgradeable [6].

The CAN bus connection system was built in this study to maintain rapid and stable communication between a range of control modules. CAN bus connection system was intended as follow: TMS320F2407 DSP chip eCAN module was used as a CAN controller, and the chip SN65HVD230 was used as a CAN transceiver, as well as a variety of control modules. The bus connects them. The chassis connection of a security robot, for example, is used to demonstrate how the CAN bus works. Empirical findings indicate that robot security chassis communication through CAN bus is fast and stable [7]. The CAN bus communication system described in this study can solve the unstable problem in control module communication and can be extended to other control systems. With great qualities, extreme dependability, and a one-of-a-kind design, the CAN protocol is particularly ideal for industrial operations monitoring equipment interconnections, and it has been acknowledged as one of the most promising field bus [8].

III. SOFTWARE ARCHITECTURE

The CANopen protocol was created to allow many devices to interact, access each other’s data, and monitor nodes in embedded environments.

A. Heartbeat Protocol

The Heartbeat protocol is used to keep track of the network’s node. A byte in the data component of the packet indicates the node state.The Heartbeat protocol is used to keep track of the network’s nodes and make sure they’re still alive. A byte in the data component of the packet indicates the node state. These signals are read by the heartbeat consumer. If the messages do not come within a specified time limit the master node will notifies the user or removes that node for further communication.

B. Block Diagram

The CAN BUS Analyzer Tool is used to monitor the CAN bus’s traffic. A unique id will be assigned to each node in the network. This one-of-a-kind identifier will be used to refer to each node in the network. Individual nodes or all nodes get directives from the master.

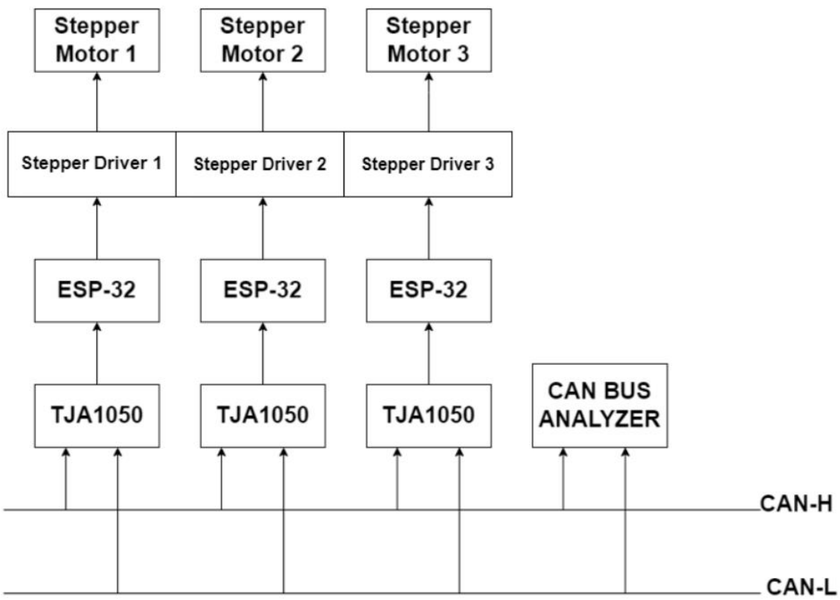

Figure 1: CAN Architecture

The CAN transceiver converts the logic levels on the CAN bus to the logic levels on the microcontroller. After then, each slave in the network applies an acceptance filter to the messages it receives. If the received messages have passed via the acceptance filter, the received message will be checked to see if it is for that node or not using the identifier value. If a message is meant for a certain slave, the slave will conduct the appropriate action and then respond with the state of the action. The master node writes the stepper motor direction and rotation angle in the slave’s memory map, as well as the stepper motor run instruction in a separate message.

ISSN (O) 2321-2004, ISSN (P) 2321-5526
International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
ISO 3297:2007 Certified ⁒ Impact Factor 7.047 ⁒ Vol. 10, Issue 5, May 2022
DOI: 10.17148/IJIREEICE.2022.10554

C. Interpretation of COB-ID

The 11-bit CAN ID is broken into two parts: a 4-bit function code that defines what action should be done and a 7-bit CANopen node ID that identifies each node individually. A CANopen network can only have 127 nodes because of the 7-bit size limitation. The 4 bits of the 11-bit CAN ID determine what action should be taken, while the remaining 7 bits are utilised to identify the node. Because of the 7-bit size, a maximum of 127 nodes can be created.

COB ID		Data Field	
Function Code	Node ID	Address Offset	Value

COB ID		Data Field	
Function Code	Node ID	State	NULL

Figure 2: COB-ID Structure

The first two bytes of the data field represent the CAN memory map's address offset. The other two bytes fields will have value if it was a write command. In the data field of the read command, only the address will be delivered.

D. Reset Mechanism

The master node is in charge of loading default data to each slave node. When the master node boots up, it sends a CAN command across the network to load default values. If a slave node is reset unexpectedly under this manner, the slave node notifies the master node. The slaves' default values were established by the master node. If the master node is reset, the slave nodes must also be reset in order to sync data between the master and slave nodes. As a result, slaves send a reset message to the master node, and the master transmits the slaves default data.

E. Wi-Fi

There are three different Wi-Fi modes available. The ESP32 will link to an established WiFi network in station mode. In SoftAP mode, the ESP32 acts as an access point for other nodes, allowing them to connect to it. In Coexistence Mode, the ESP32 will be able to generate an access point while also connecting to a WiFi network.

F. Web-Server

The online interface for controlling numerous stepper motors in this project was created using the http web server. To control a stepper motor or sync all nodes in the network, the user can choose which node to utilise.

Figure 3: Web Server

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526
International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 7.047 ⁜ Vol. 10, Issue 5, May 2022
DOI: 10.17148/IJIREEICE.2022.10554

The user can adjust the stepper motor's rotation angle using the slider on the web interface.

G. Work Flow

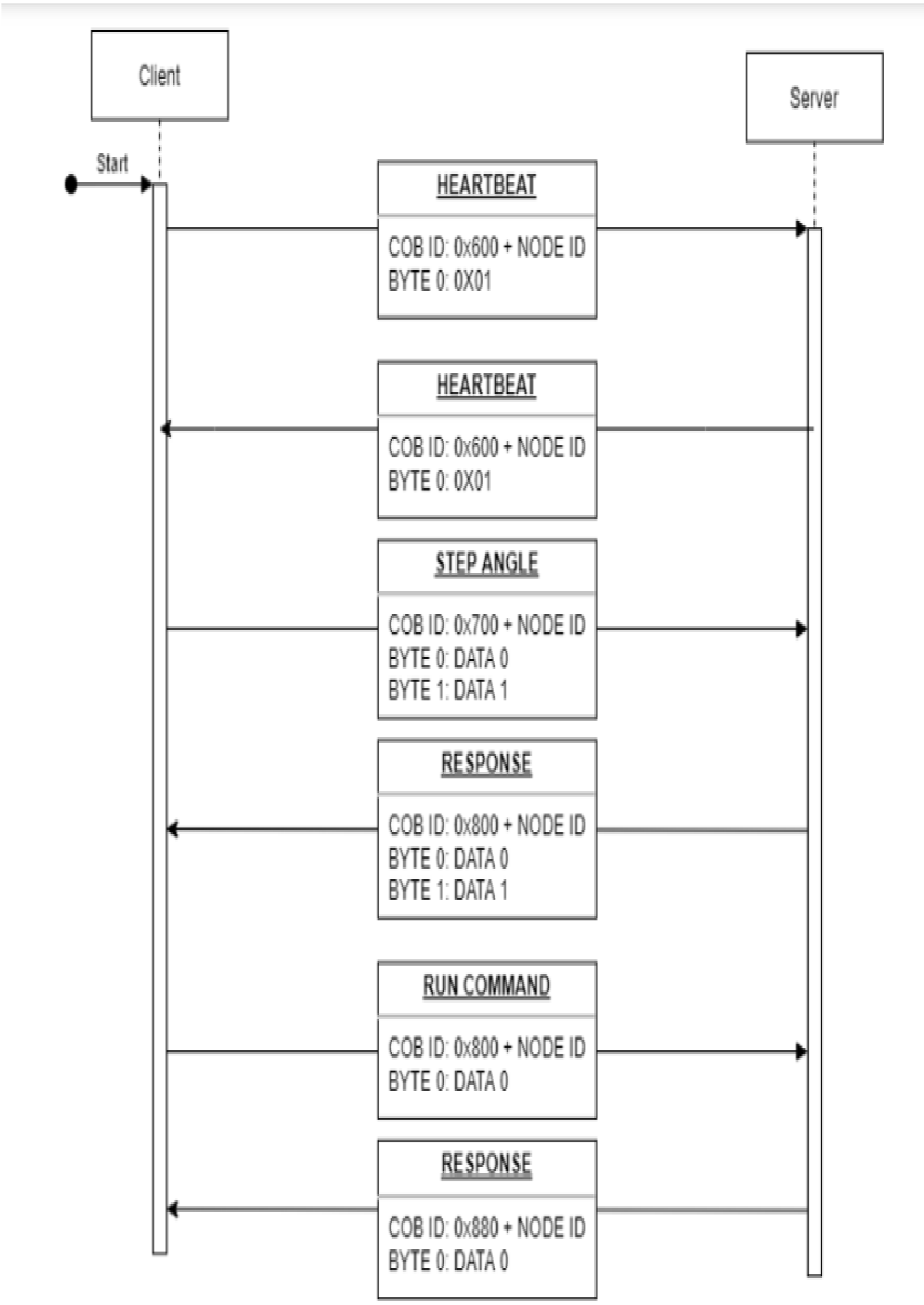

Figure 4: Work Flow

The master node will keep track of which nodes are communicating. Through HTTP, the master node will collect data from the stepper web server, such as node and stepper rotation angle. Based on the commands received, the master node will transmit suitable commands to the nodes. The master should deliver the stepper rotation angle to the necessary nodes first. The master node will transmit the Run Command after receiving a successful write event.

The libraries used to construct the CAN WiFi controller, as well as the actual development, will be discussed in the following parts

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⋇ Impact Factor 7.047 ⋇ Vol. 10, Issue 5, May 2022

DOI: 10.17148/IJIREEICE.2022.10554

IV. TEST BENCH

The hardware indicated in figure 5 was used to test the application's functionality and check for any issues. The components were put on top of a breadboard so that the ESP32 and CAN transceiver pins could be easily accessed.

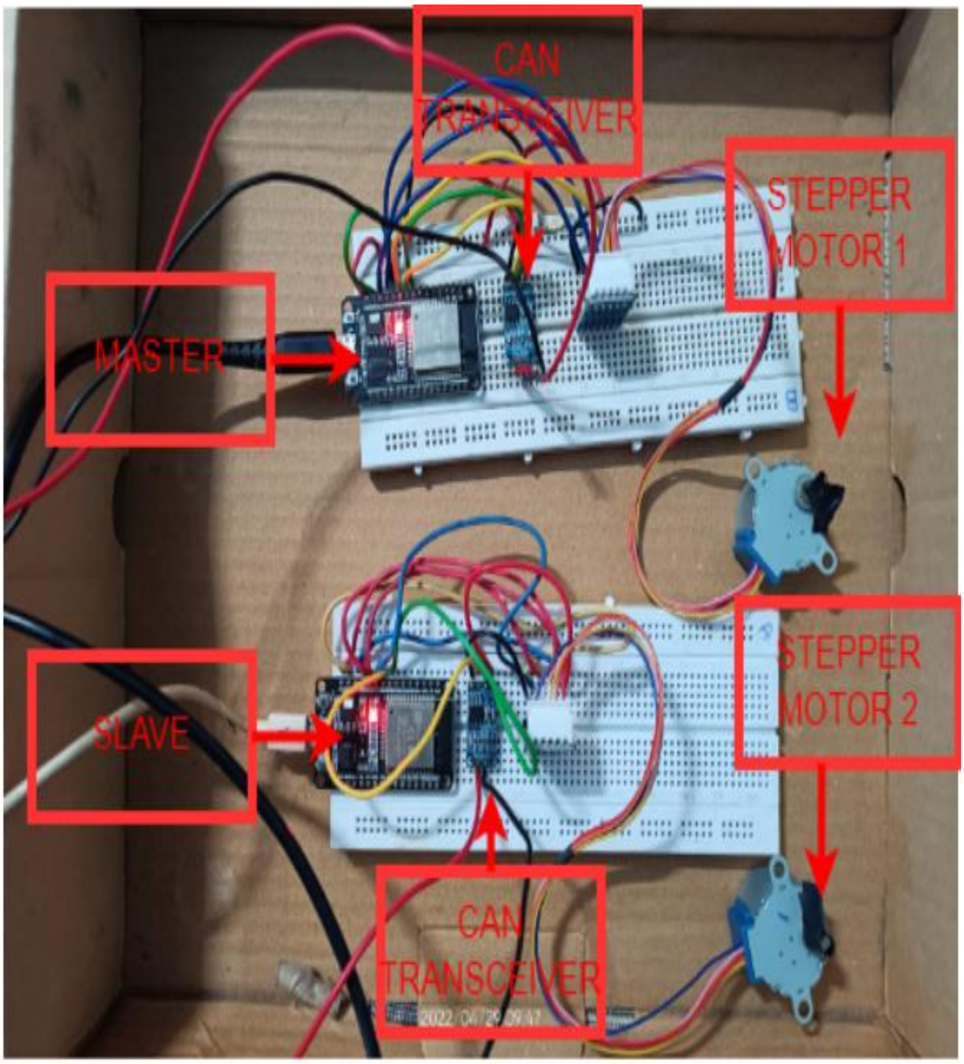

Figure 5: Test Bench

As shown in Figure, there are two ESP32 boards, one on the upper side for transmission and the other on the lower side for reception. Of course, both nodes can send and receive data, but this setup is the ideal for testing. Each nodes share the same power source, which might originate from one of them being connected to a laptop or a power source through the micro-USB connector. In order for the CAN Tranceiver to work correctly, the VCC pin must be connected to +5V. The ESP32 nodes may now be linked to the CAN H and CAN L lines thanks to the CAN transceiver. Despite the fact that both nodes are terminal nodes, they" close" the CAN network, hence no 120 resistance is required. The CAN H and CAN L pins on the transceiver already have one built in.

The upper node will be linked to an existing Wi-Fi network, and an HTTP web server will be created. The HTTP web server will send data and instructions to the top node. If the received commands are for the second node, it will use CAN Link to deliver the necessary data to the connected node.

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.047 ⁜ Vol. 10, Issue 5, May 2022

DOI: 10.17148/IJIREEICE.2022.10554

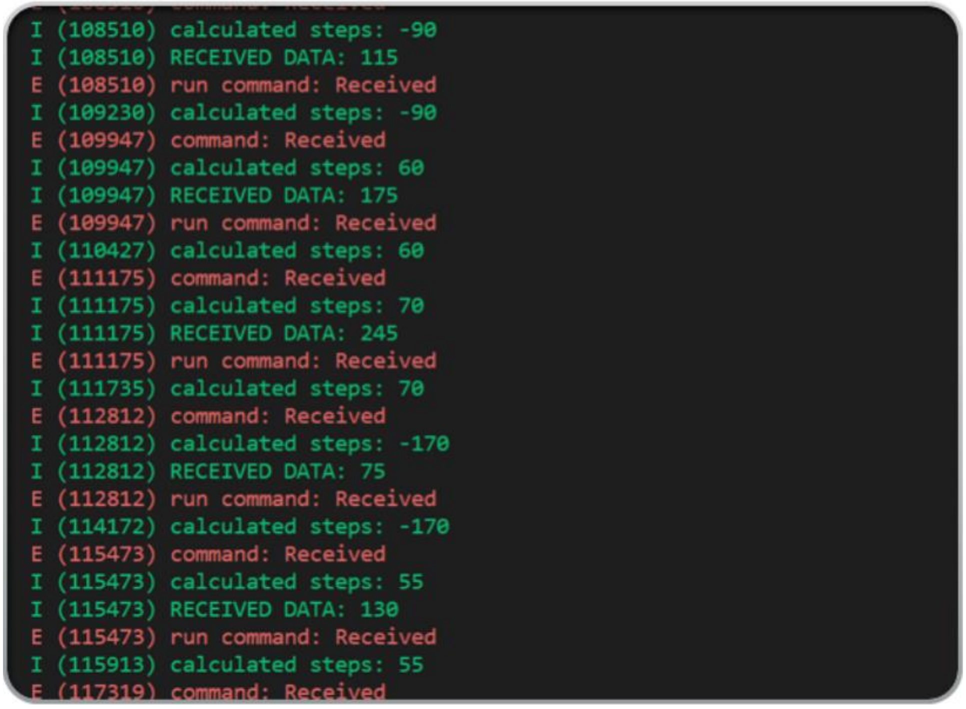

Figure 6: CAN Log

Figure shows the commands received from slave mode. The master node delivers the rotation angle first, followed by the run command.

IV. CONCLUSION

In order to address the limitations of existing stepmotor control systems, this article proposes a control system for numerous stepmotors based on ESP32 and CANbus. The ESP32 is employed as the main control chip in the system, and the CAN bus serves as the link for interconnecting each stepper motor, allowing for real-time and high-speed interaction with each control node. The control system thus accomplishes simultaneous control of multiple stepper motors during system testing and can regulate and alter the functioning condition of the stepper motors in real time. The experimental findings indicate that the control system achieves the predicted effect, validates the design's viability.

V. FUTURE SCOPE

The proposed system has the potential to grow to 40 nodes. Because of the lack of encryption and authentication, the CAN protocol has a number of security flaws. As a result, any attacker node has the potential to cause severe accidents as well as economic loss. This may be avoided by creating advanced filters.

REFERENCES

[1]. Wang X, Lu S, Zhang S. Rotating Angle Estimation for Hybrid Stepper Motors with Application to Bearing Fault Diagnosis[J]. IEEE Transactions on Instrumentation and Measurement, 2020, 69(8):5556-5568.
[2]. J. Ko lota and S. Stepie´n, "Analysis of 3D model of reluctance stepper motor with a novel construction," 2017 22nd International Conference on Methods and Models in Automation and Robotics (MMAR), Miedzyzdroje, 2017: 955-958.
[3]. Gaogao Wang. Multi stepper motor control system based on CAN bus [D]. Huainan: Anhui University of Science Technology,2018.
[4]. Dandan MA. Dissertation Submitted to Hangzhou Dianzi University for the Degree of Master[D]. Hangzhou: Hangzhou Dianzi University, 2013.
[5]. CHAI Wenfeng, DING Xueming. Design of CAN Communication Control Network Based on STM32 MCU [J]. Electronic ence Technology, 2017,30(3):142- 145.

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁙ Impact Factor 7.047 ⁙ Vol. 10, Issue 5, May 2022

DOI: 10.17148/IJIREEICE.2022.10554

[6]. Ashwini S. Shinde, Prof. vidhyadhar B. Dharmadhik "Controller Area Network For Vehicle Automation" IJETAE ISSN 2250-2459, Volume 2, Issue 2, February 2012.

[7]. S. Vijayalakshmi "Vehicle control system implementation Using CAN protocol" International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering Vol. 2, Issue 6, June 2013.

[8]. Guang Liu, Ai-ping Xiao, Hua Qian "Communication System Design Based on TMS320F2407 with CAN Bus". 2012 AASRI Conference on Modeling, Identification and Control.