Ministry of Education
Department of Higher Education (Lower Myanmar)
Yangon Institute of Education

Yangon Institute of Education Research Journal

Vol. 4, No. 1

December, 2012

Testingof a Controller Area Network (CAN)

Hnin Aye Khine[1]and Pho Kaung[2]

Abstract

Controller Area Network (CAN) is a serial bus protocol which is used in real time applications. A CAN has been designed and constructed for two nodes. The two nodes can communicate each other by using CAN bus. Each node consists of a PIC18F458 microcontroller, an MCP 2551 transceiver chip, an LCD module and a sensor. Two nodes are almost identical but only differ in sensor used. Voltage sensor (potentiometer) is attached to node1 and temperature sensor (LM 35) is attached to node2.The PIC18F458 microcontroller has been chosen for the constructed system because it has a built- in CAN module. One microcontroller reads the data from its sensor and sends it to another node via the CAN bus. The microcontroller also requests the data from the other node. MCP 2551 transceiver chip has been used to convert the logic signal to CAN signal. The data is transferred between two nodes by using CAN protocol and is shown on their respective LCD module. The data transfer rate is 200 kb s^{-1} that applies to a network up to 21 meters (70 feet).

Introduction

The Controller Area Network(CAN) is developed by German automotive system supplier Robert Bosch in the early 1980s. Thereafter, CAN was standard asISO-11898,establishing itself as the standard protocol for in-vehicle networking in the auto industry.The PIC18F458 microcontroller has been chosen for the constructed system because it has built in CAN module. MCP 2551 has been used to output the CAN signal. CAN communication is a good choice for the design needed to communicate between the sensors due to its reliability and high performance. The block diagram in fig.1 shows how two PIC nodes are linked over the Controller Area Network.

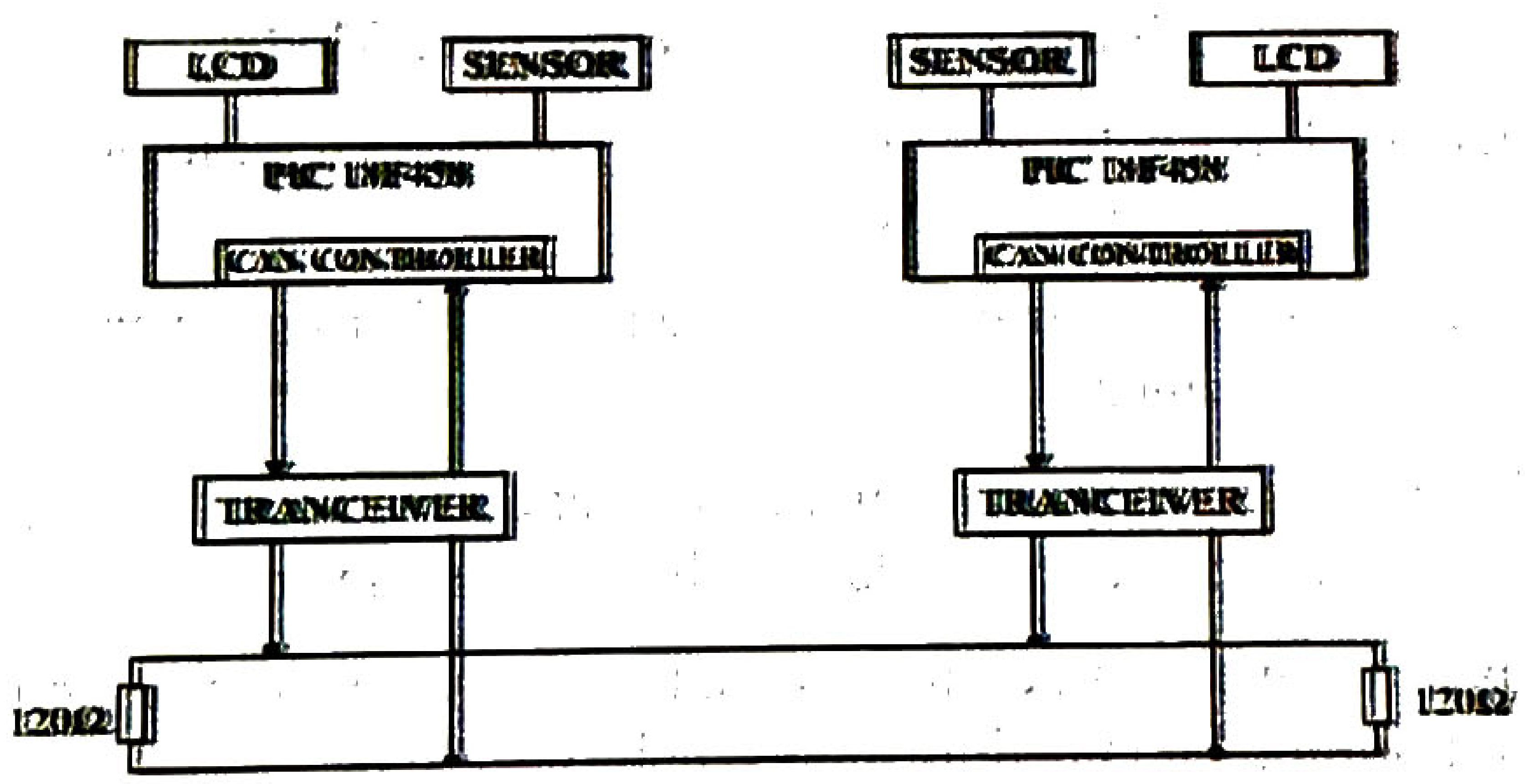


Figure 1. Block Diagram of CAN

1. Demonstrator, Dr., Department of Physics, Yangon Institute of Education
2. Pro-rector, Dr., Universities' Research Centre, University of Yangon

Layered Approach in CAN

Most network application follow a layered approach to system implementation. A standard was created by the International Standards Organization (ISO) as a template to follow this layered approach. It is called the ISO Open Systems Interconnection (OSI) Network layering Reference Model (Bosch R, 1991).

The CAN protocol specified the lowest two layers of this reference model.The lowest two layers are data link layerand physical layer.These two layers are as shown in fig (1).

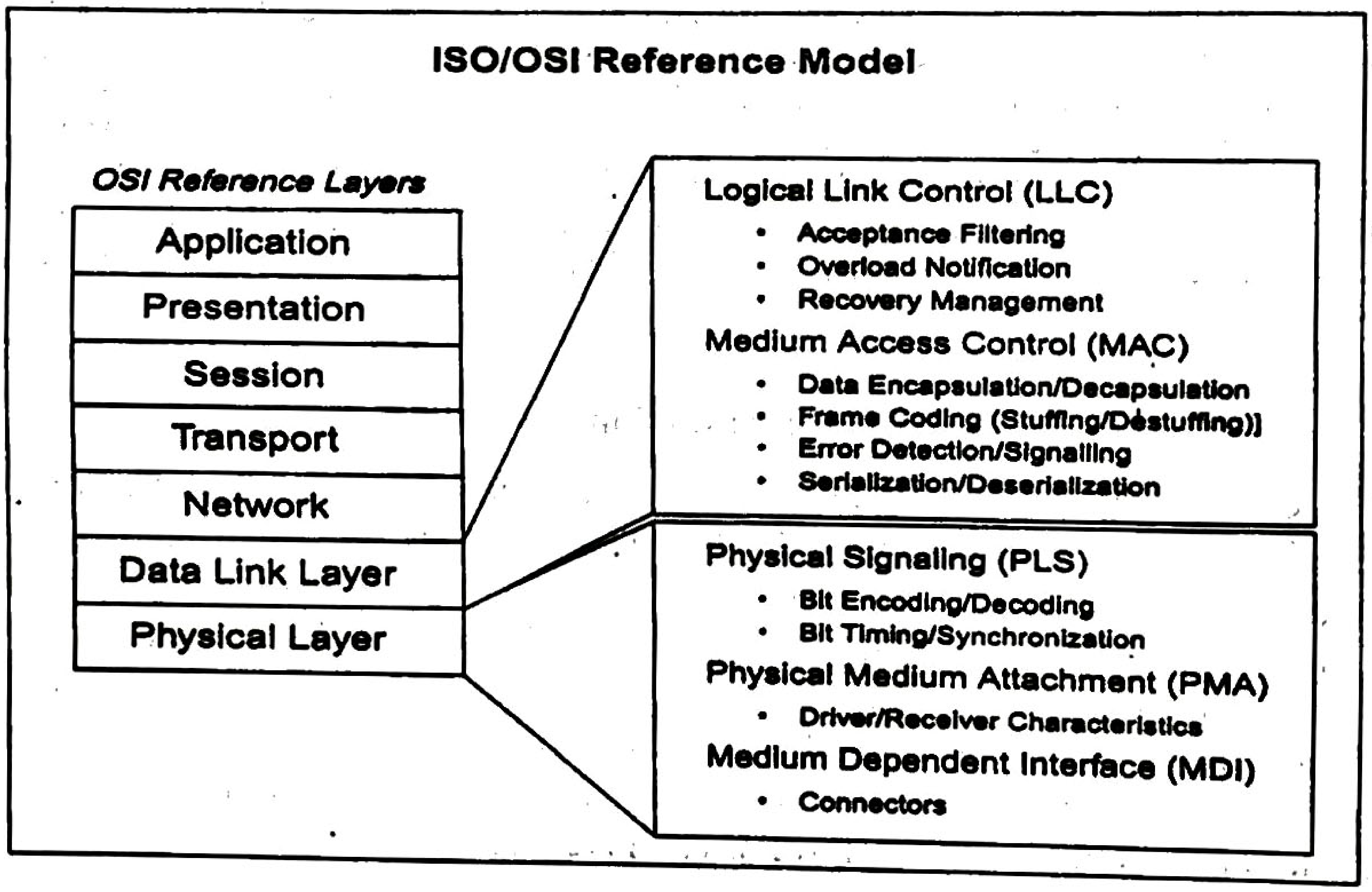


Figure 2. ISO/OSI Reference Model

The data link layer is further divided into two sublayers: logical link control (LLC) layer and medium access control (MAC) layer.

- The LLC sublayer deals with massage acceptance filtering, overload notification, and error recovery management.
- The MAC sublayer presents incoming messages to the LLC sublayer and accepts messages to be transmitted forwarded by the LLC sublayer.

The physical layer defines how signal are actually transmitted, dealing with the description of bit timing, bit encoding, and synchronization.

CAN Protocol Basics

The CAN protocol is based on CSMA/CD (Carrier-Sense Multiple Access/Collision Detection) with Arbitration on Message Priority protocol.

Carrier Sense Multiple Access with Collision Detection (CSMA/CD)

The CAN communication protocol is a CSMA/CD protocol. The CSMA Stands for Carrier Sense Multiple Access. What this means is that every node on the network monitor the bus for a period of no activity before trying to send a message on the bus (Carrier Sense). Also, once this period of no activity occurs, every node on the bus has an equal opportunity to transmit a message (Multiple Access). The CD stands for Collision Detection. If two nodes on the network start transmitting at the same time, the nodes will detect the "collision" and take the appropriate action.

Non Destructive Bitwise Arbitration (NDA)

In CAN protocol, a non-destructive method is utilized. This means that messages remain intact after arbitration is completed even if collisions are detected. All of this arbitration takes place without corruption or delay of the higher priority message.

There are a couple of things that are required to support non-destructive bitwise arbitration. First, logic states need to be defined as dominant or recessive. Second, thetransmitting node must monitor the state of the bus to see if the logic state it is trying to send actually appears on the bus. CAN defines a logic bit 0 as a dominant bit and a logic bit 1 as a recessive bit.

When there is arbitration on the bus, a dominant bit state always wins over a recessive bit state. The process of arbitration is illustrated in fig.(3) by an example.

Example:

Node 1:11100110011 Node2:11100110001

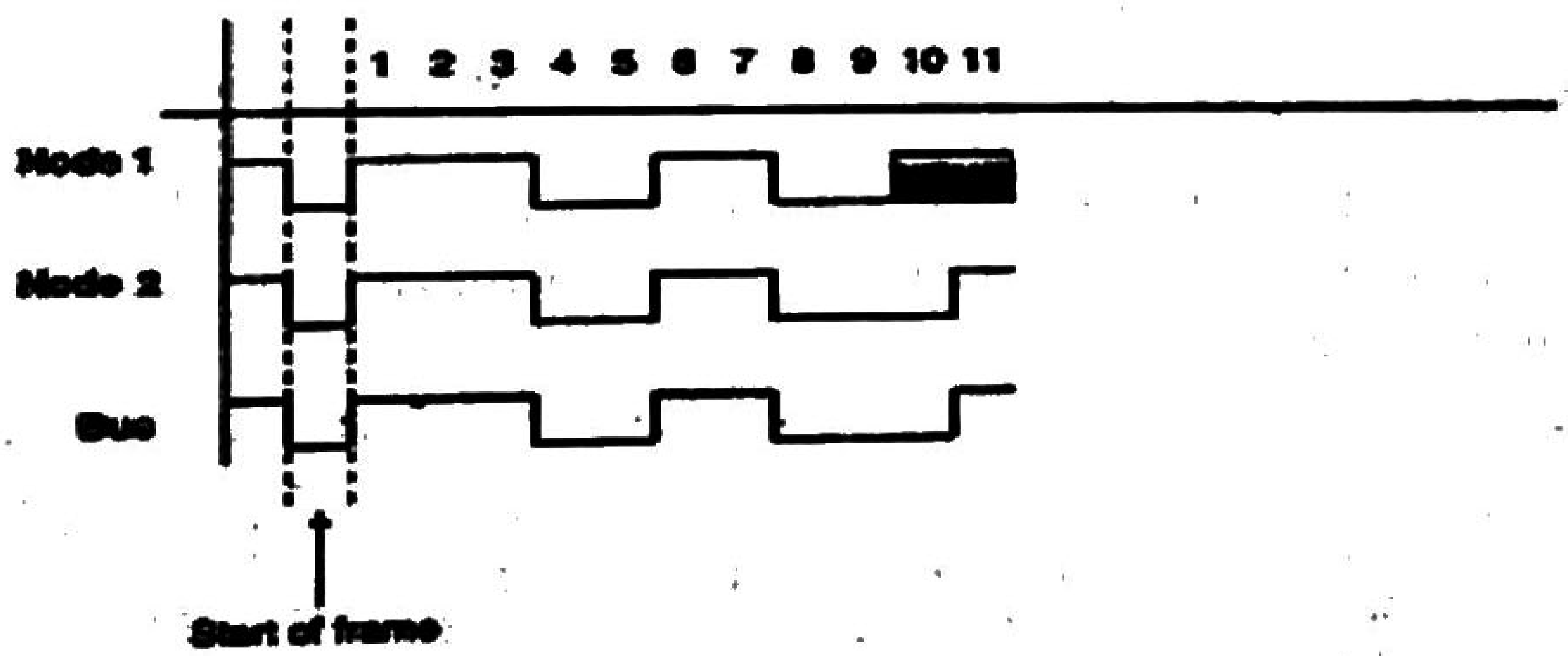


Figure 3. Example CAN bus arbitration

In the dominant state, the differential voltage CANH and CANL is greater than the minimum threshold. In the recessive bit state, the differential voltage CANH and CANL is less than the minimum threshold. CAN logic states are as shown in Fig (4).

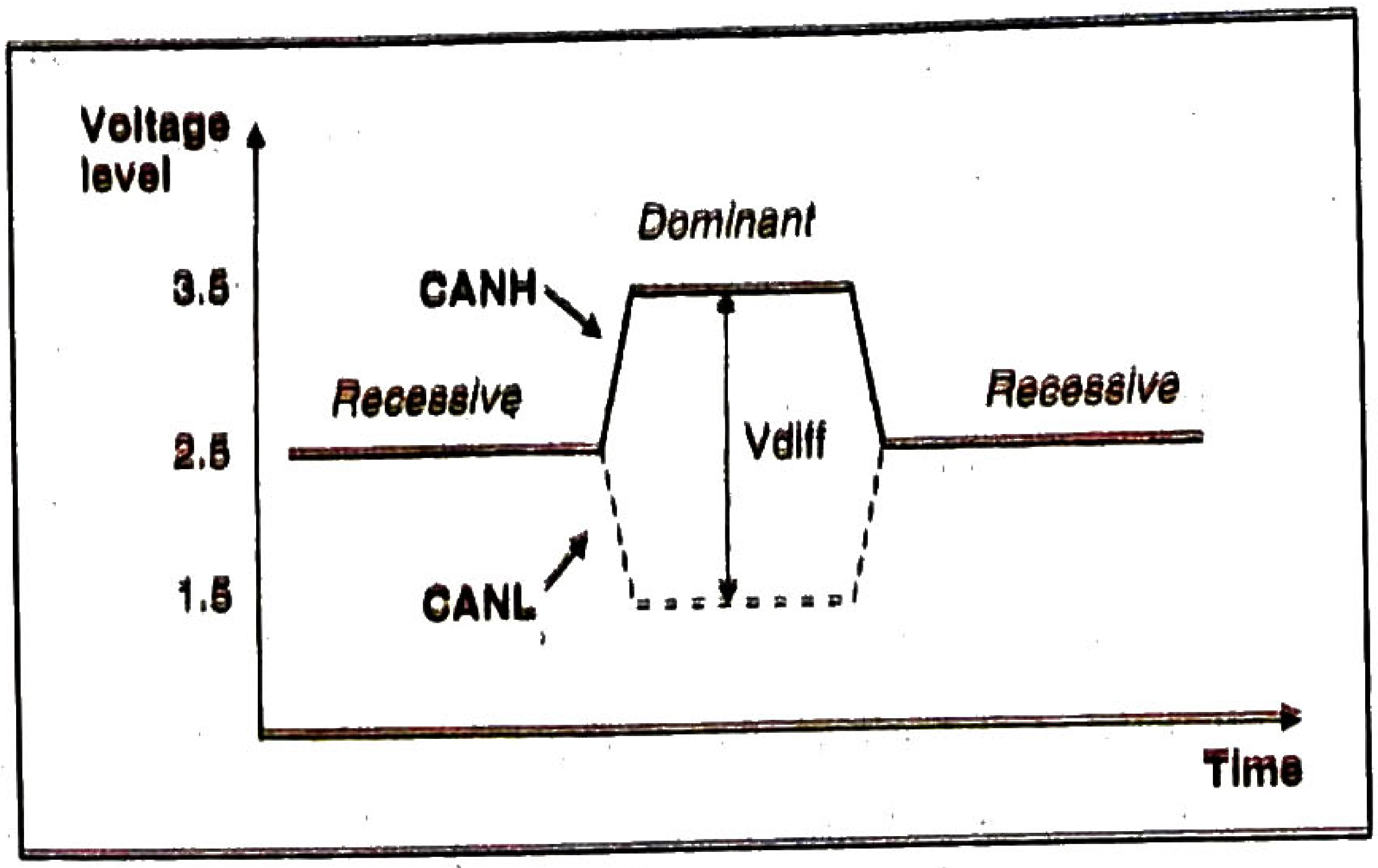


Figure 4. CAN Logic States

Message – Based Communication

CAN protocol is a message-based protocol, not an address based protocol. Embedded in the CAN message itself is the priority and the contents of the data being transmitted on the bus. It is up to each node in the system to decide whether the message received should be immediately discarded or kept to be processed.

Another useful feature built into the CAN protocol is the ability for a node to request information from other nodes. This is called a Remote Transmit Request (RTR)

CAN Message Frame

CAN protocol defines four different types of messages (or Frames). The first and most common type of frame is a data frame.

Data Frame

The data frame is used by the transmitting device to send data to the receiving device. The data frame is in two formats: Standard (having an 11bit ID) and extended (having a 29bit ID) are as shown in fig (5) and fig (6).

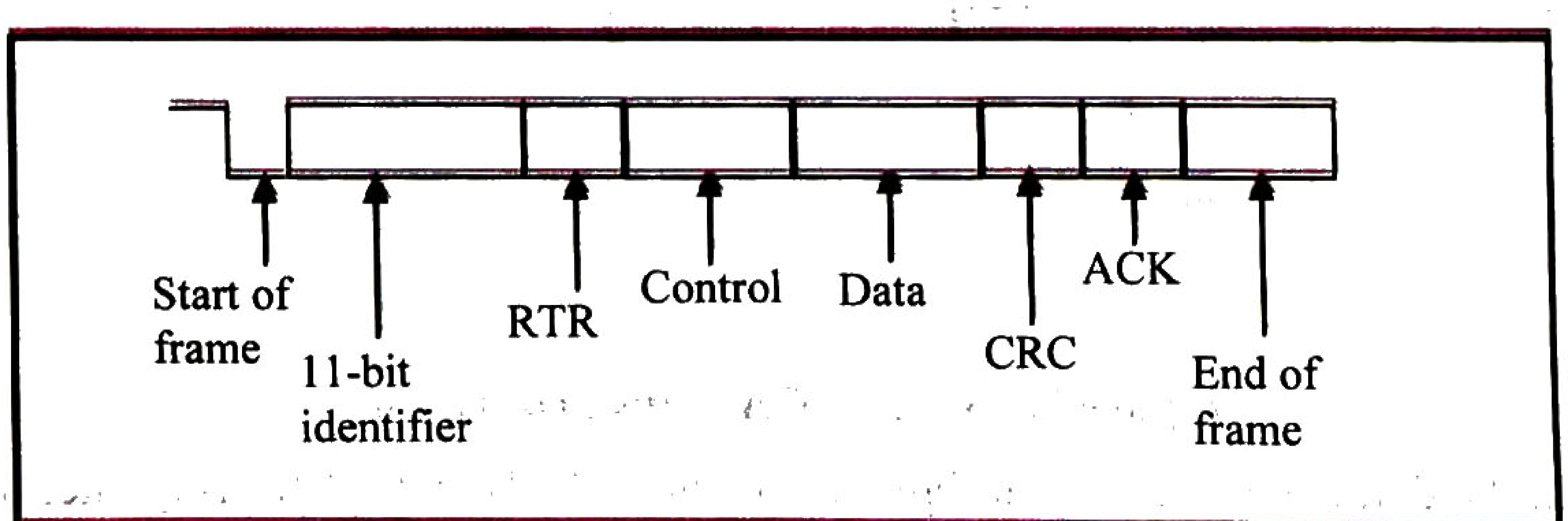


Figure 5. Standard data frame

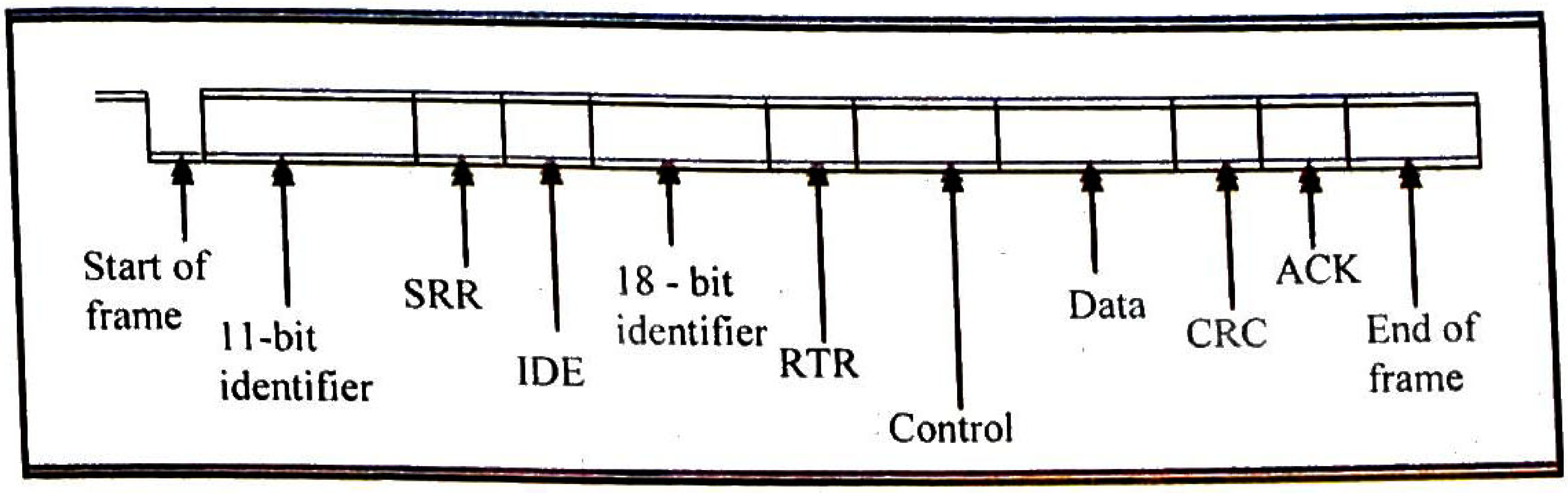


Figure 6. Extended data frame

Data frame consists of fields that are Arbitration field, control field, Data field, CRC field, Acknowledge field and End of frame. The data field can be 0 to 8 bytes.

Remote Frame

Remote frame is used by the receiving device to request transmission of a message from the transmitting device. A remote frame is the same as a data frame except that it lacks a data field and the RTR bit is at the recessive state. Fig (7) shows the remote frame.

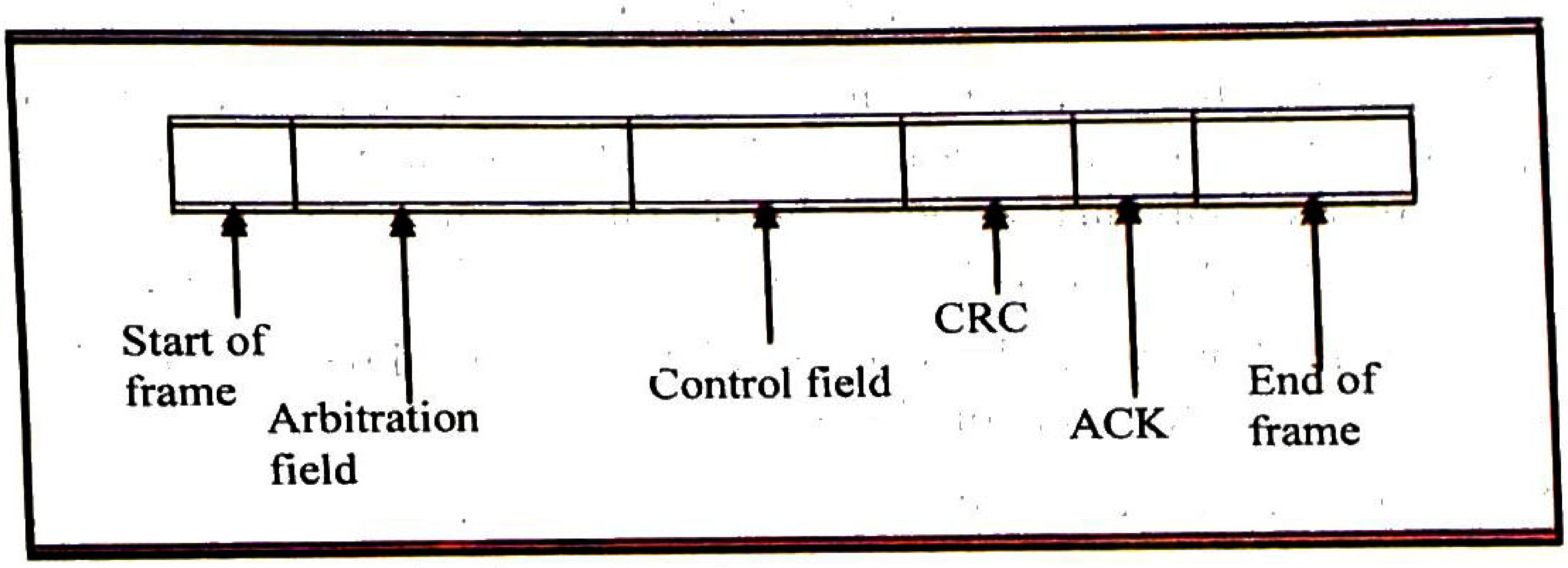


Figure 7. Remote frame

Error Frame

Error frames are generate and transmitted by the CAN hardware and are used to indicate when an error has occurred during transmission. An error frame consists of an error flag and error delimiter.

Overload Frame

The overload frame is used by the receiving unit to indicate that it is not yet ready to receive frames. This frame consists of an overload flag and an overload delimiter.

Bit Stuffing

The CAN bus makes use of bit stuffing, a technique to periodically synchronize transmit- receive operations to prevent timing errors between receive nodes. After 5 consecutive bits with the same level, one bit of inverted data is added to the sequence. If, during sending of a data frame or remote frame, the same level occurs in 5 consecutive bits anywhere from the start of frame to the CRC sequence, an inverted bit is inserted in the next (i.e, the sixth) bit. If during receiving of a data frame or remote frame, the same level occurs in 5 consecutive bits anywhere from the start of frame to CRC sequence, the next (sixth) bit is deleted from the received frame. CAN bit stuffing is as shown in fig (8).

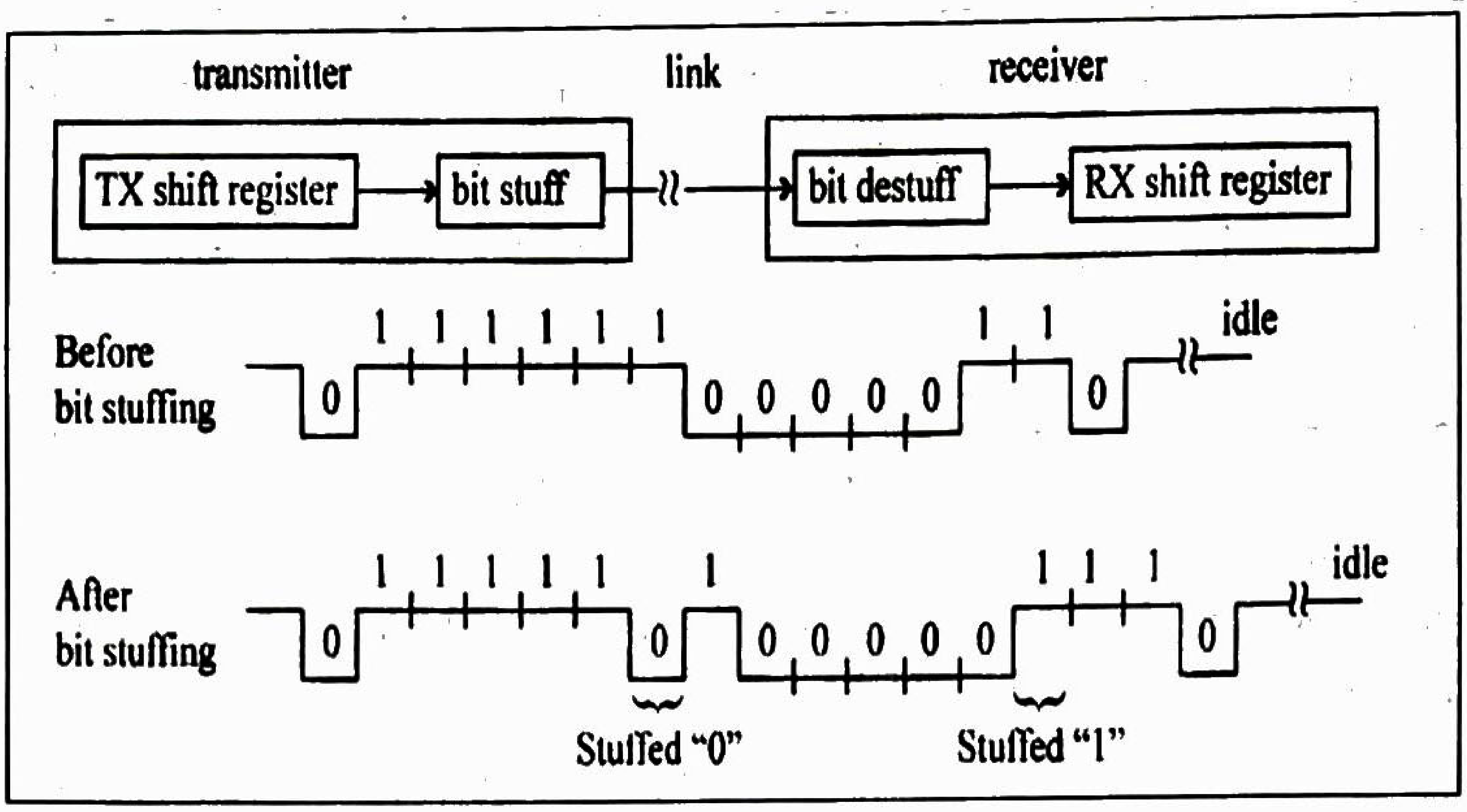


Figure 8. CAN Bit Stuffing

Baud Rate Setting

All devices on the CAN bus must use the same bit rate. However, all devices are not required to have the same master oscillator clock frequency. For the different clock frequencies of the individual devices, the bit rate has to be adjusted by appropriately setting the baud rate prescalar and number of time quanta in each segment.

The nominal bit rate is the number of bits transmitted per second. The nominal bit rate is defined to be a maximum of 1Mb/s.

The nominal bit time is defined as:

TBIT = 1/Nominal bit Rate.

Device Overview of PIC18F458

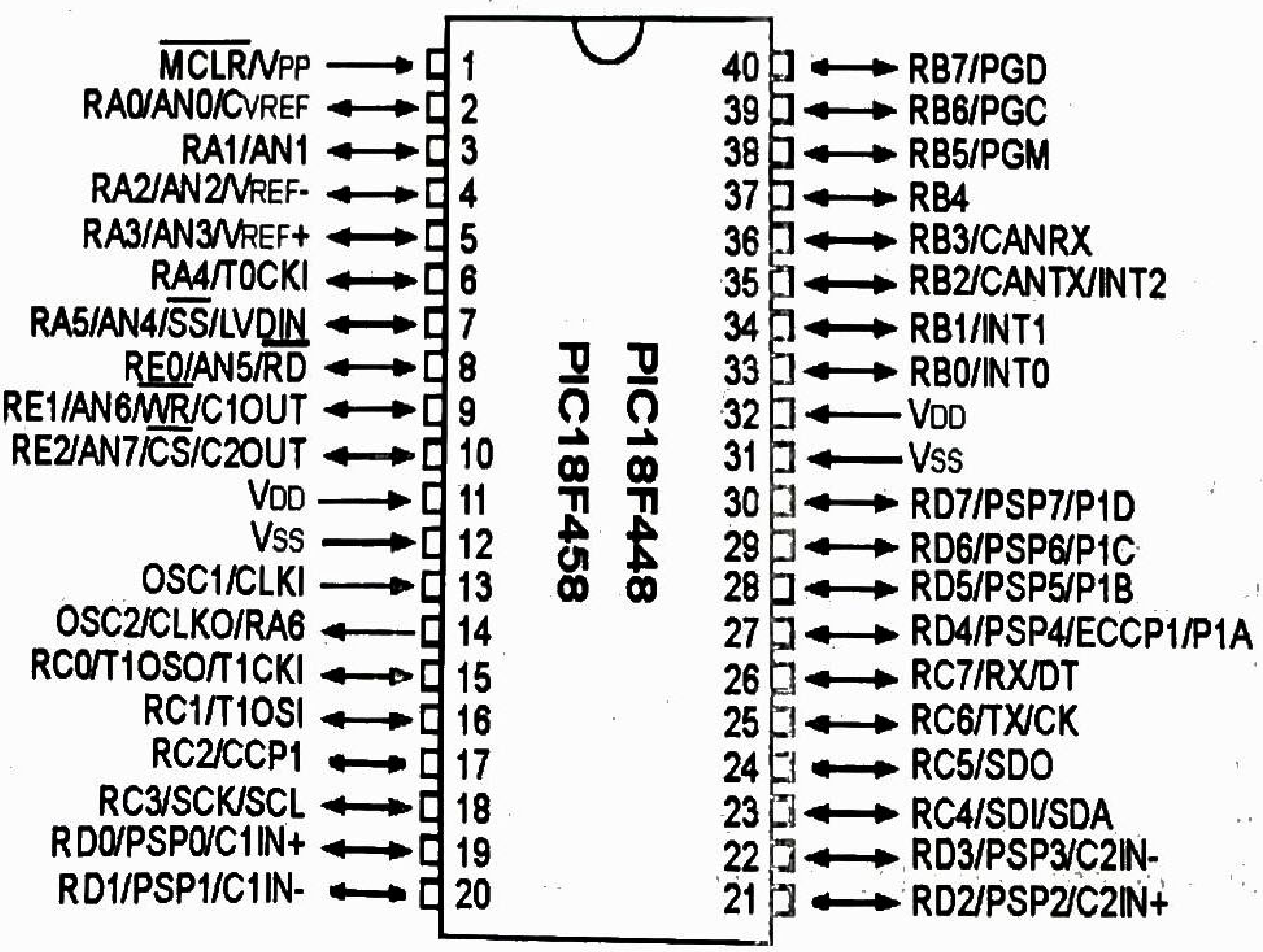


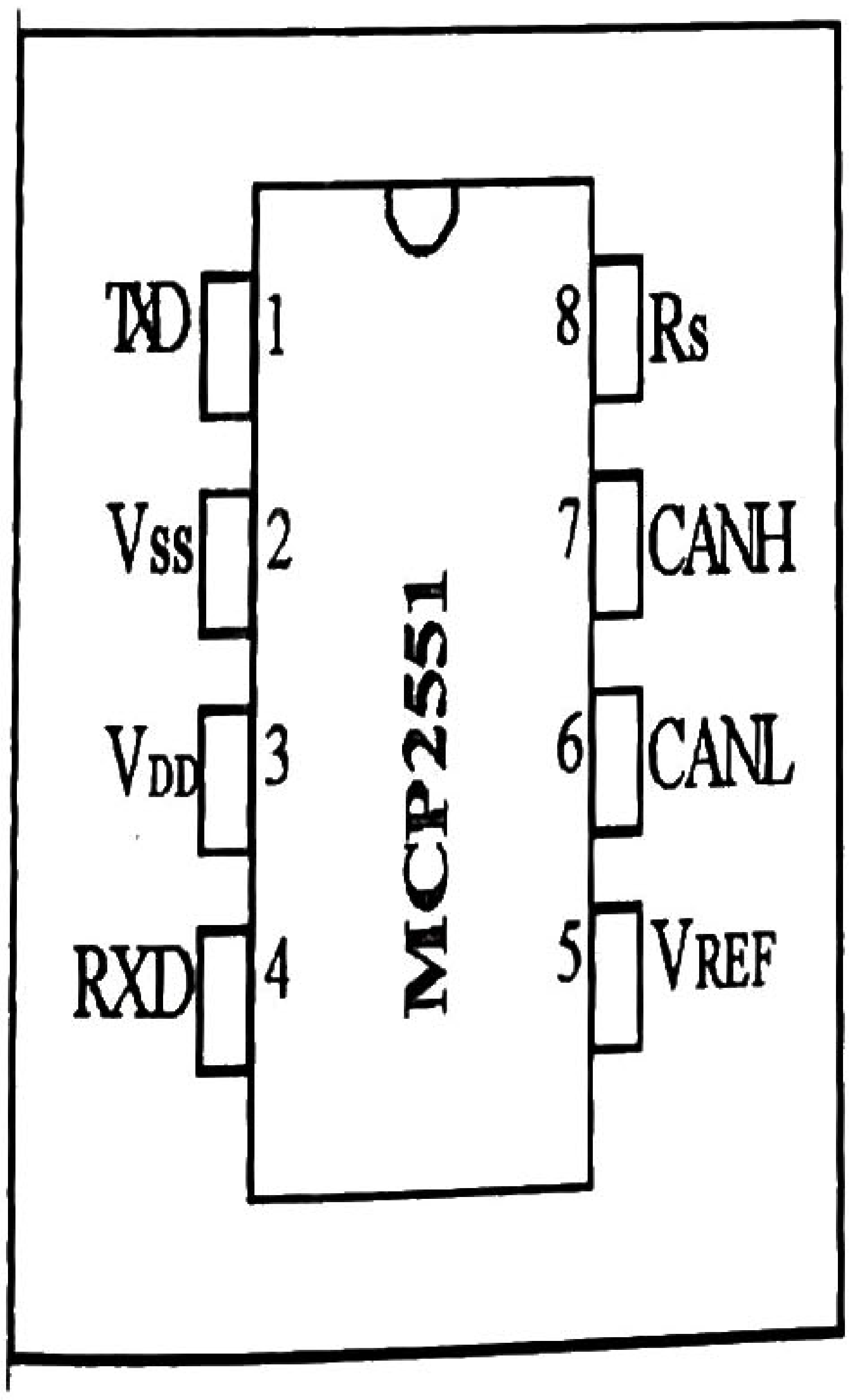
Figure 9. Pin Diagram of PIC18F458

The PIC18F458 is used in CAN (Controller Area Network) communication. The device is chosen for the CAN system because it is a high performance 8-bit microcontroller with integrated CAN module.

The PIC18F458 devices are available in 40pin and 44-pin packages. The device has the following features

- 40 MHz Operating frequency
- 32 K bytes Program memory
- 16384 Program memory instructions
- 1536 bytes Data memory
- 256 bytes EEPROM memory
- 21 Interrupt sources
- I/O ports (A,B,C,D,E)
- 4 Timers
- 1 Capture / Compare/ PWM Modules
- 8 input channel 10-bit Analog-to-Digital Converters
- 75 Instruction set

Device Overview of MCP2551



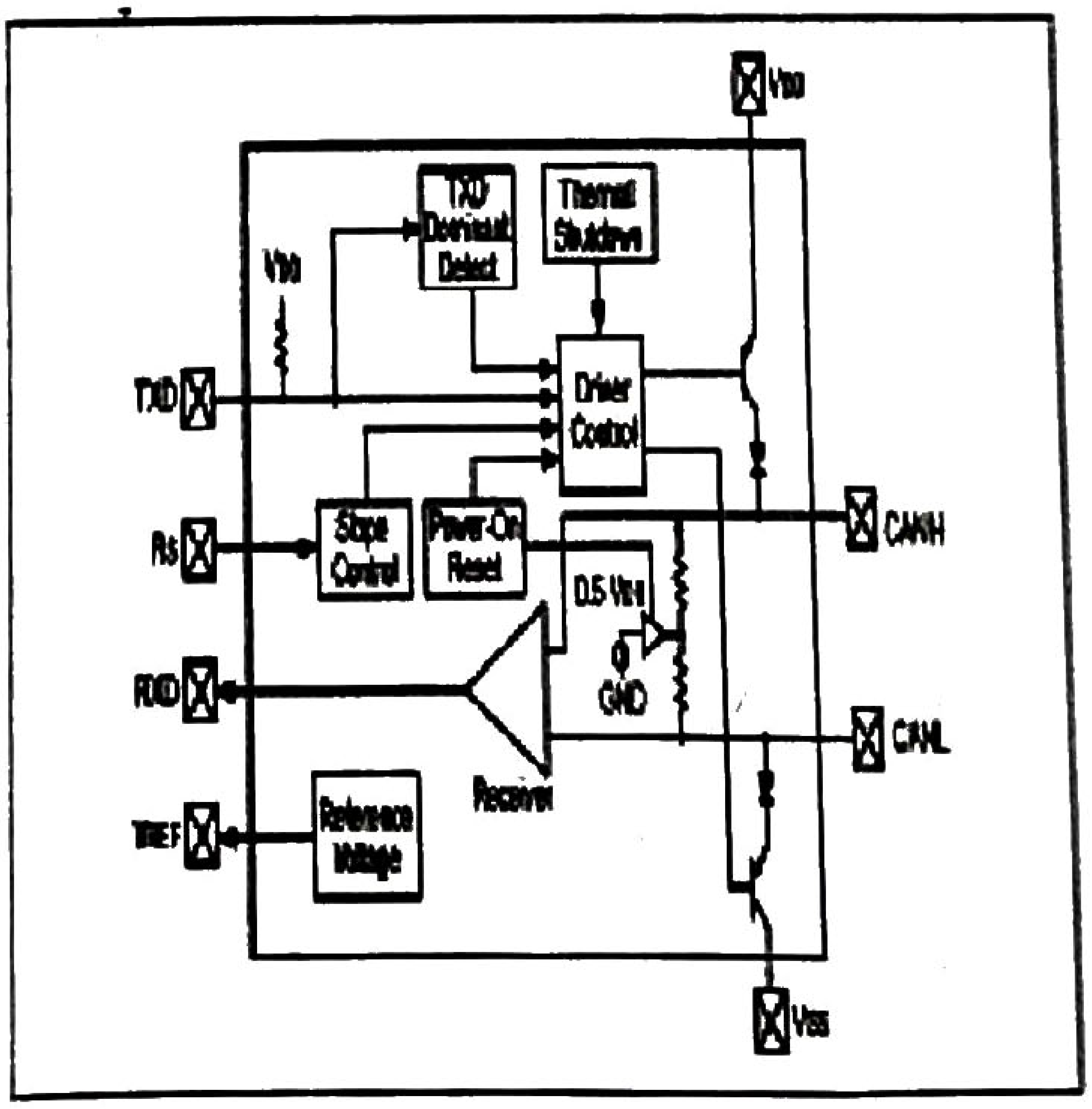


Figure 10. Package Type of MCP2551 **Figure 11. Block Diagram of MCP2551**

The MCP2551 is a high speed CAN, fault - tolerant device that serves as the interface between a CAN protocol controller and the physical bus. The MCP2551 provides differential transmit and receive capability for the CAN protocol controller and is fully compatible with the ISO-11898 standard.

Typically, each node in a CAN system must have a device to convert the digital signals generated by a CAN controller to signals suitable for transmission over the bus cabling (differential output).

The device has the following features.

- Supports 1 Mb/s operation.
- Implements ISO-11898 standard layer requirements.
- Detection of ground fault (permanent dominant) on TXD input.
- Power-on reset and voltage brown -out protection.
- Low current standby operation.
- Protection against damage due to short -circuit conditions. (positive or negative battery voltage)
- Protection against high-voltage transients
- Automatic thermal shutdown protection.
- Up to 112 nodes can be connected.

Circuit Design of a Controller Area Network

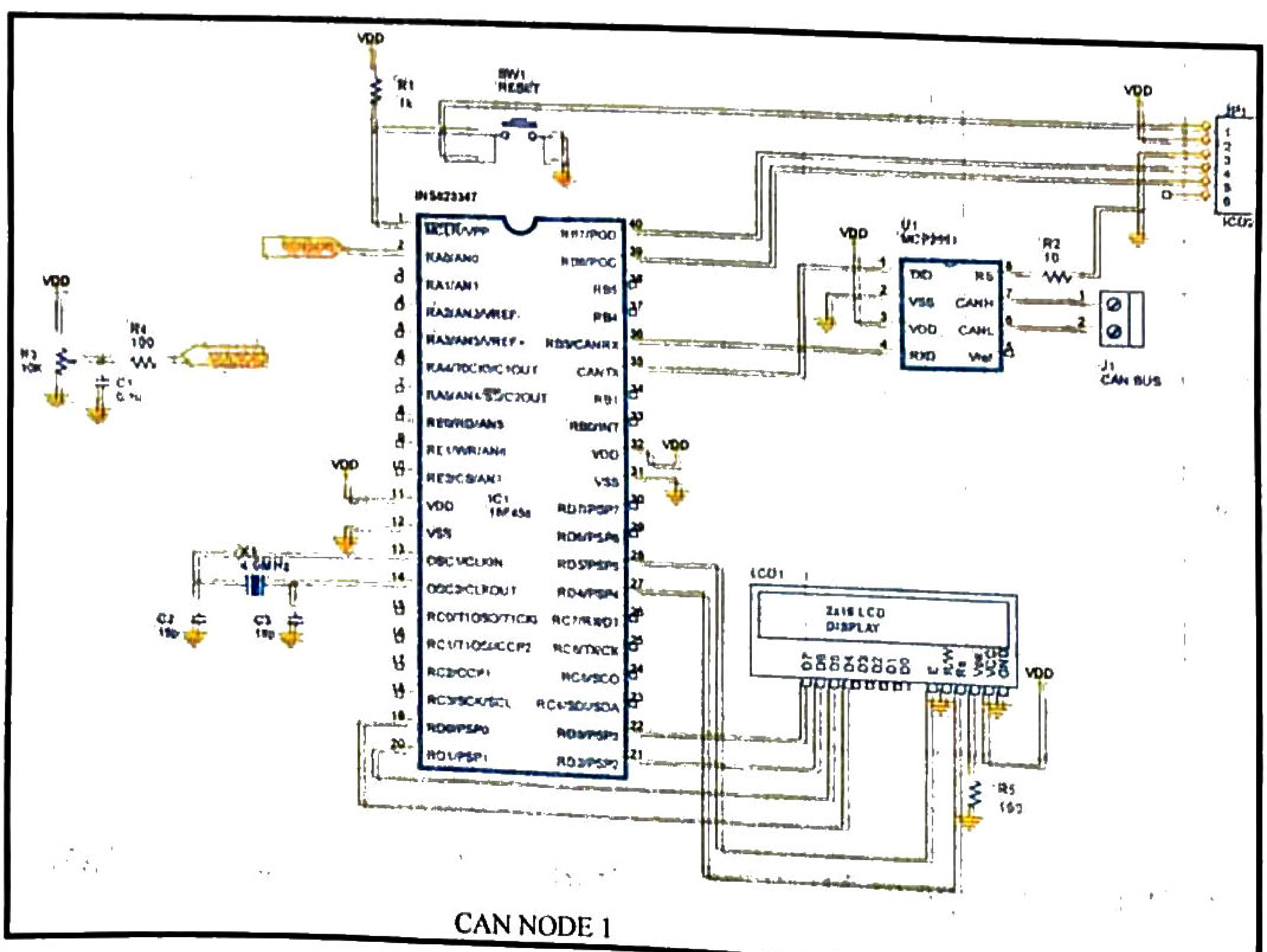


Figure 12. Circuit Diagram of CAN NODE 1

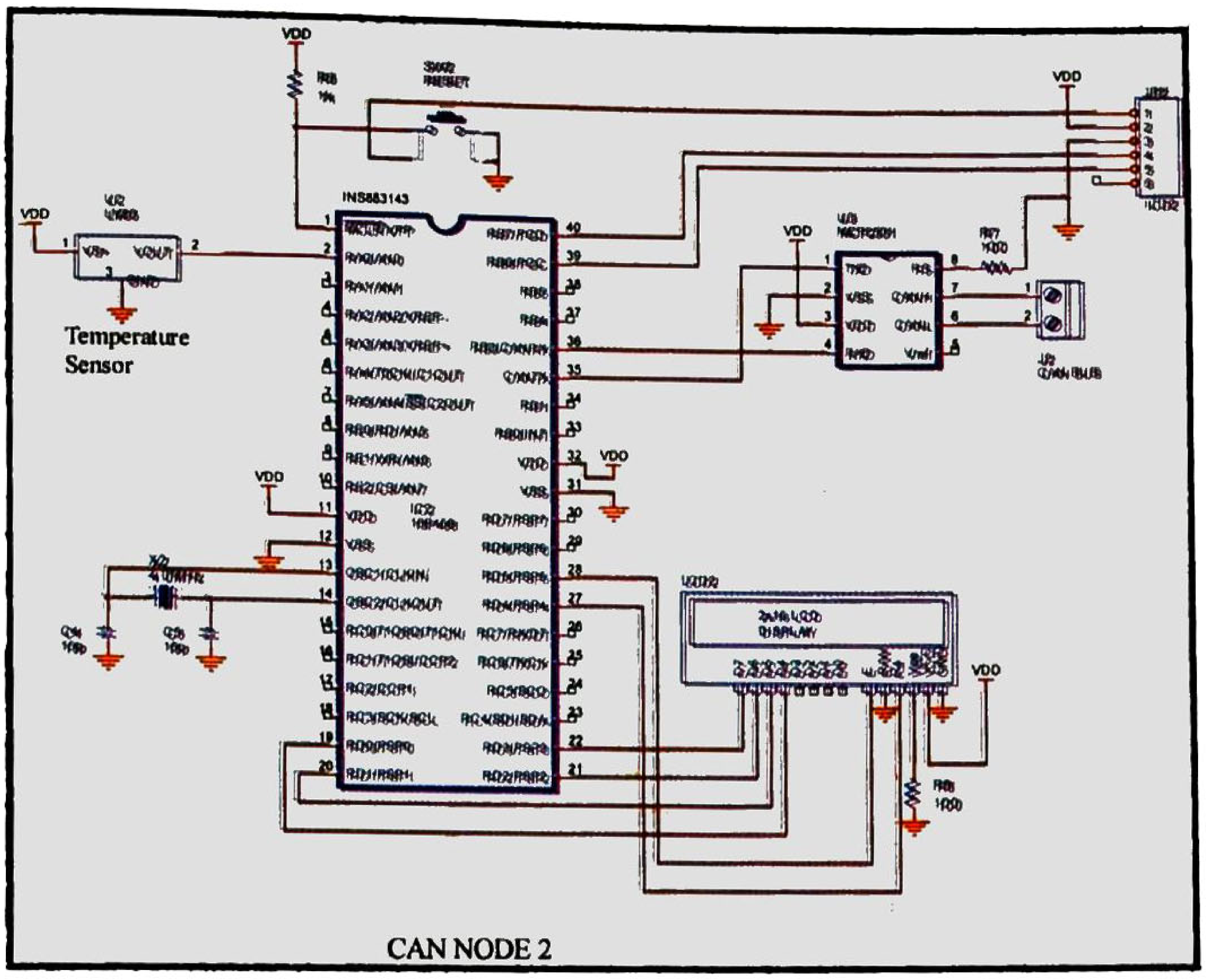


Figure 13. Circuit Diagram of CAN NODE 2

This circuit was designed to developsoftware for CAN (Controller Area Network) communication. The constructed two nodes are shown in fig(12) and fig (13). They are almost identical only differ in sensor used. In each node, the sensor is connected to the (RAO/ANO) pin of the PICI8F458. A potentiometer is used for node 1 which can measure in the range of 0 to 5V and LM35 is used for node 2 which can measure in the range of -60°C to + 150°C. The 4MHz crystal is connected to OSC0 and OSC1 pins which is used to operate the microcontroller CANTX and CANRX pins of the microcontroller are joined to the TXD and RXD pins of the MCP2551. The output of the microcontroller is logic signal which cannot be sent directly to the bus. So MCP2551 converts logic signal to CAN signal and puts on the CAN bus. CANH and CANL pins of MCP2551 drive the bus level. LCD module is connected to the portD pins of the microcontroller to display the data. Only two wires are used to transmit the data which length is 70feet (21 meter) and the transmitting rate is 200kb/s.

Results and Discussion

CAN in Operation

When a request is received from the node 2, the PIC18F458 microcontroller reads the voltage from the sensor and outputs a serial data stream to the logic TXD input of the MCP2551. The transceiver chip converts the logic signal to the CAN signal and sends to the node2 viathe CAN bus. The current temperature from the node 2 is requested by the PIC18F458 microcontroller over the CAN bus. And then the microcontroller formatsthe received data and sends to its LCD module. The above process is repeated every second. NODE1 LCD module displays the voltage as the local and the temperature as theremote. (See Fig (14))

Like the NODE1,When a request is received from the node1, the PIC18F458 microcontroller reads the temperature from the LM 35. And then the microcontroller outputs a serial data stream to the logic TXD input of the MCP2551.The transceiver chip converts the logic signal to the CAN signal and sends to the node1 via the CAN bus. The current temperature from the node1 is requested by the PIC18F458 microcontroller over the CAN bus. And then the microcontroller formats the received data and sends to its LCD module. The above process is repeated every second. NODE2 LCD module in Fig (15) displays the temperature as the local and the voltage as the remote.

Node1 can request the temperature value from node2 by using RTR and node2 can also request the voltage value from node1 by using RTR as shown in Fig (16). CAN protocol has the ability 'Remote Transmit Request (RTR)' to request the information from other nodes. This means that instead of waiting for information to be sent by a particular node, this node specifically requests data to be sent to it.

Figure (18) depicts the experimental set-up to observe the data pattern in the CAN. The standard data frame is used when the data is transmitted between node1 and node2 as shown in Figure (19). This frame consists of fields that provide additional information about the message as defined by the CAN specification. The data can be 0 to 8 bytes.

CAN specifies the two logic states (dominant and recessive) depending on a differential voltage on CANH and CANL. CANH and CANL are the two signal lines. By using the differential signaling, CAN successfully cancel out noise. The bit time is 5μs as shown in Fig(20). So data is transmitted with 200 kilobits in one second between the two nodes .

Testing the CAN's BUS

ISO-11898 defines a differential voltage to represent recessive and dominant states. In the recessive state, the differential voltage on CANH and CANL is less than the minimum threshold (<0.5V receiver input or <1.5V transmitter output). In the dominant state, the differential voltage on CANH and CANL is greater than the minimum threshold. Figure (22) shows the differential voltage of the output waveform. This voltage is between the reference differential output range (1.5<ΔV <3V). So this output waveform confirms the correctness of the CAN communication.

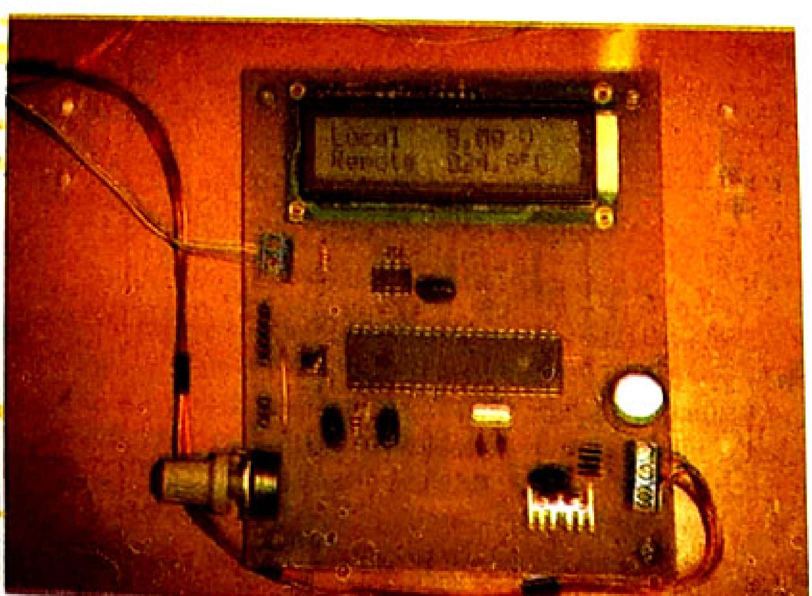


Figure 14. NODE1 LCD in action

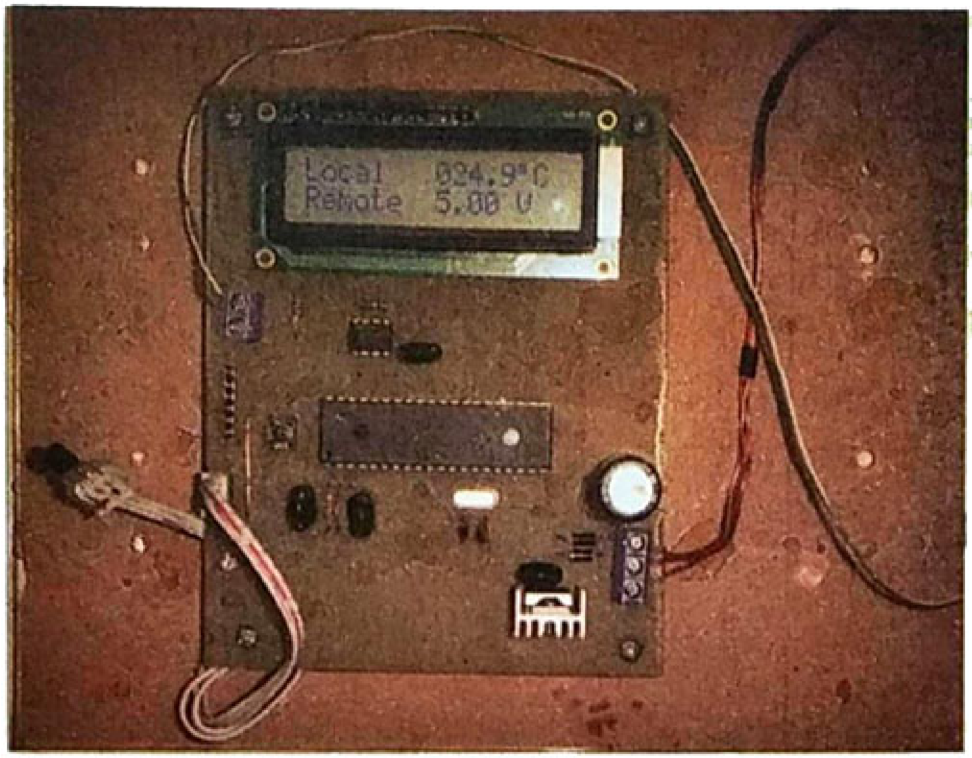


Figure 15. NODE2 LCD in action

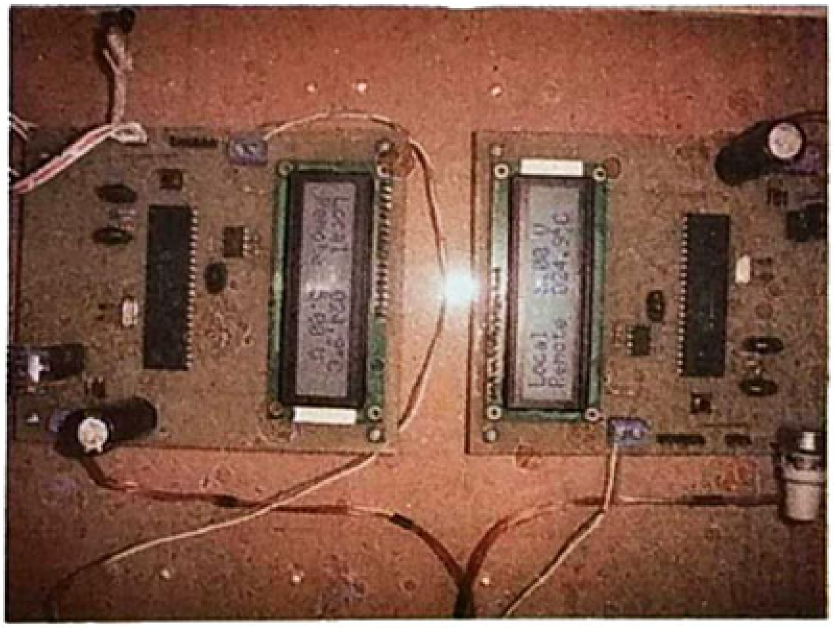


Figure 16. Controller Area Network in action

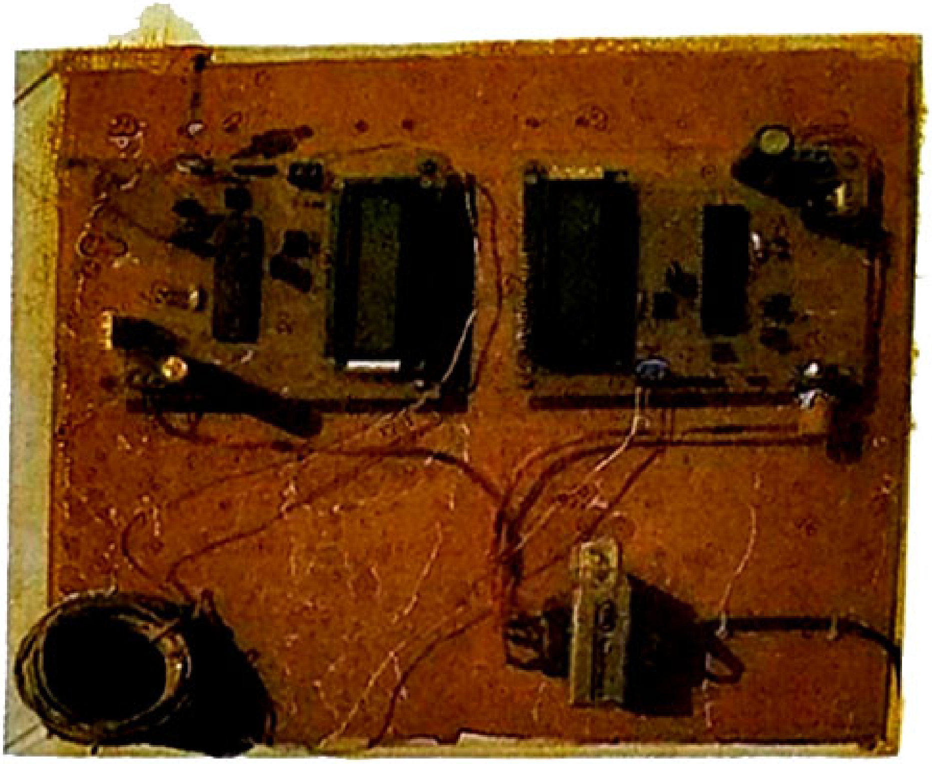


Figure 17. Controller Area Network in the initial state

Figure 18. TestingtheController Area Network

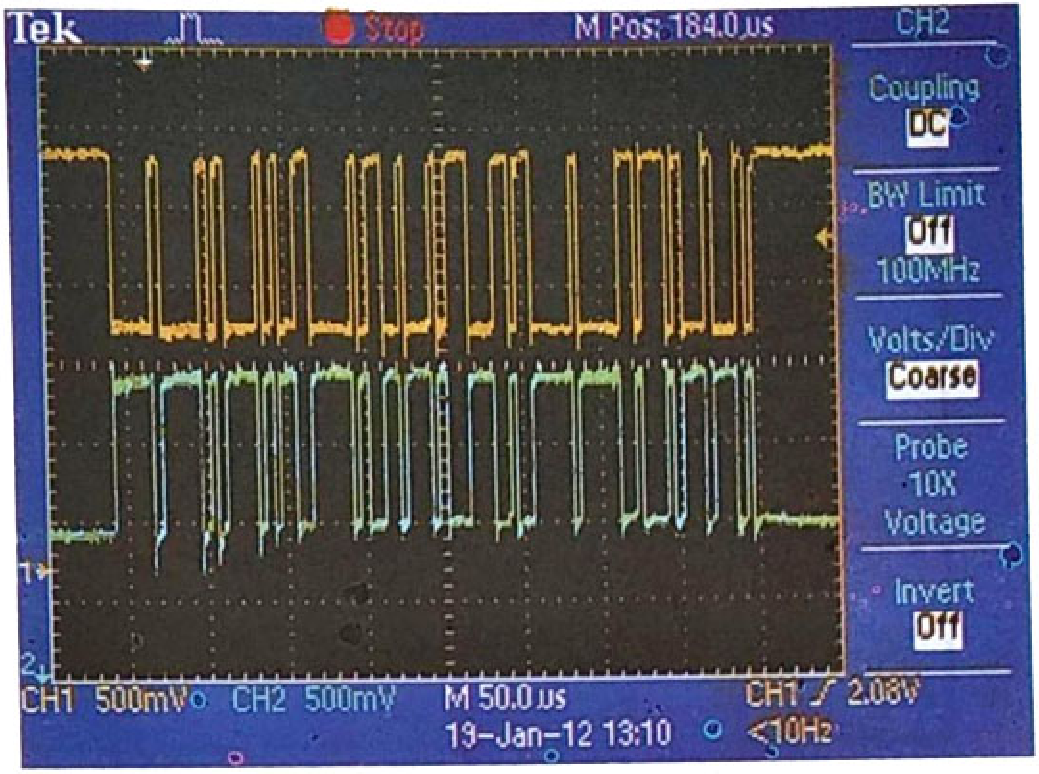


Figure 19. Data frame observed on the oscilloscope

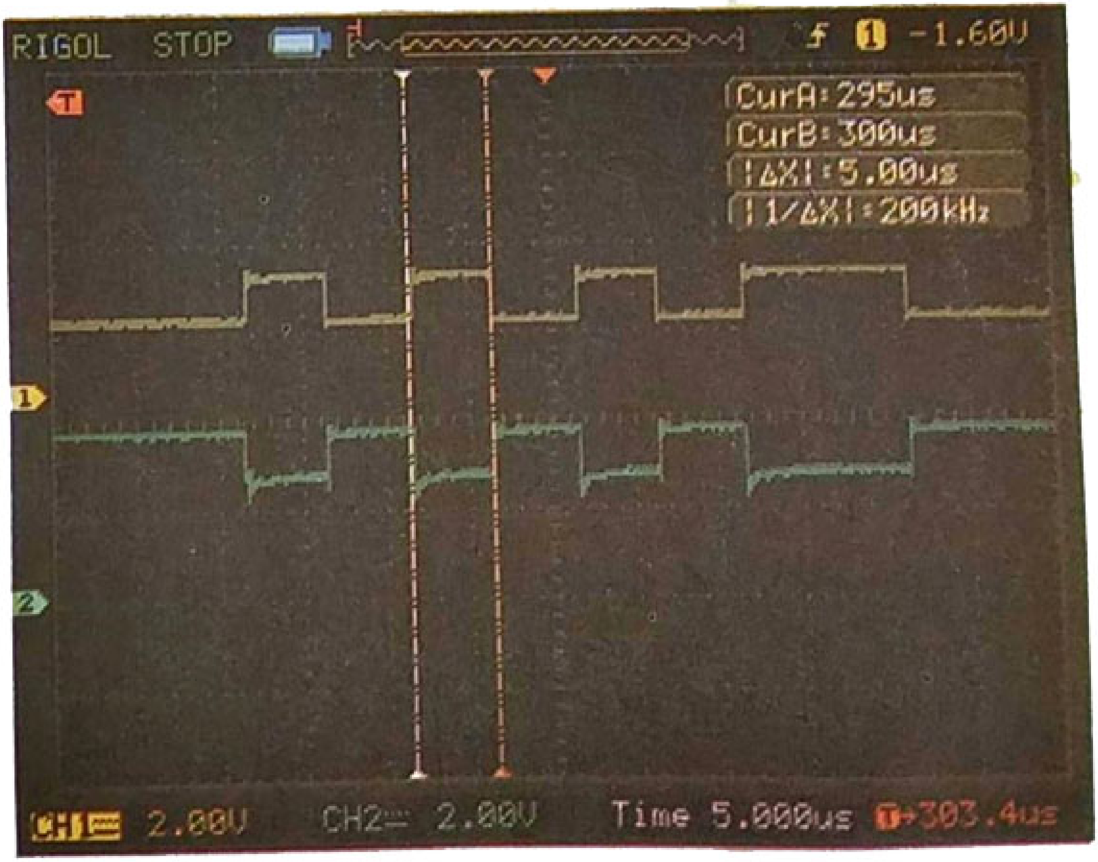


Figure 20. Measuring the pulse width

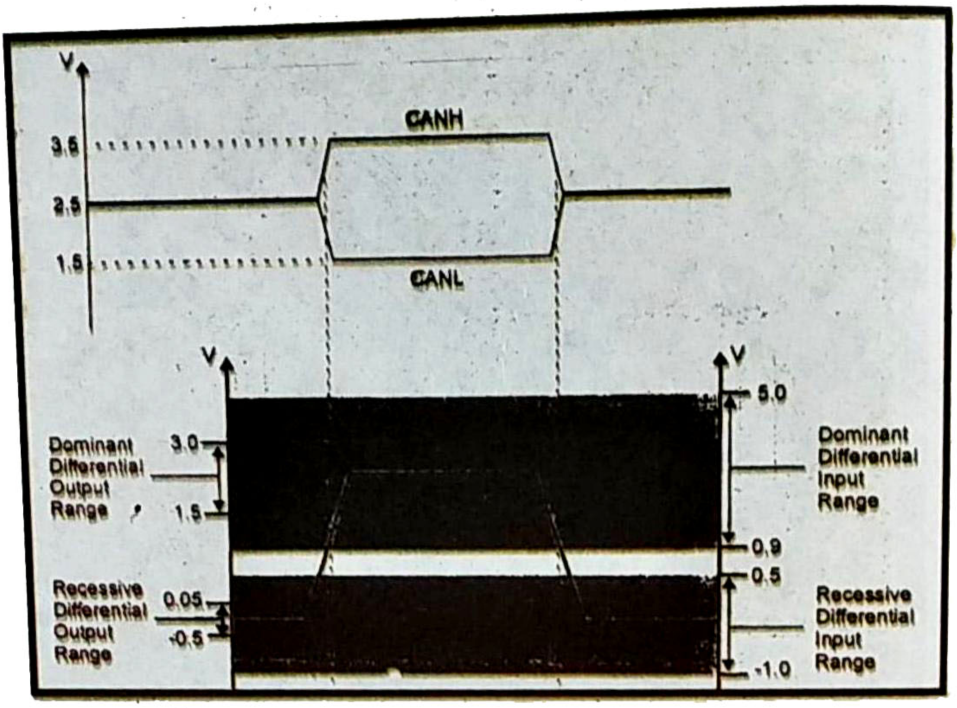


Figure 21. ISO-11898 nominal bus level

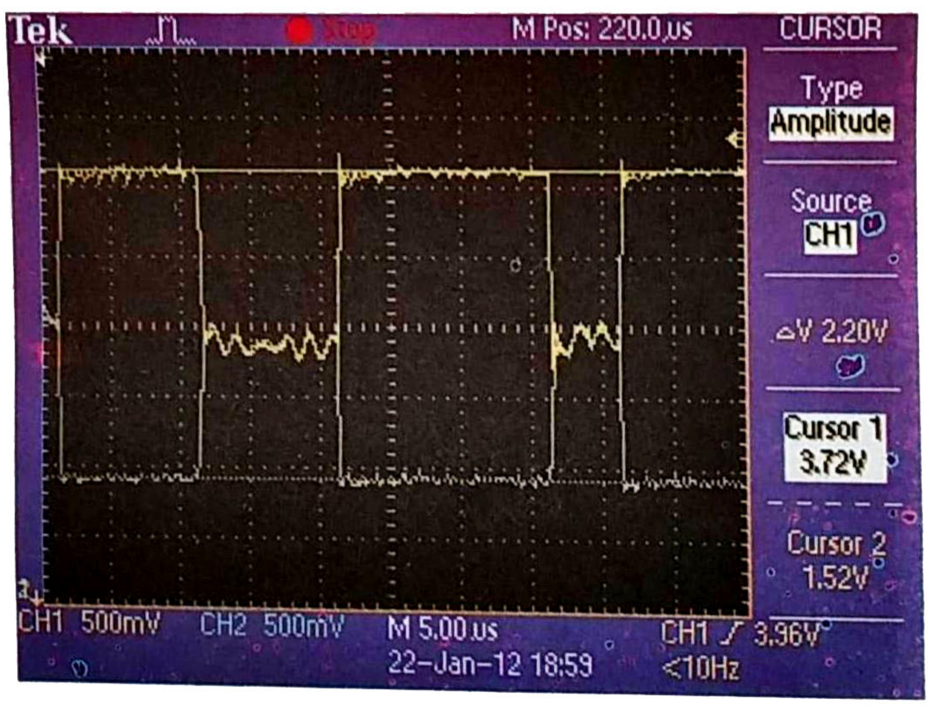


Figure 22. Measuring the differential output voltage waveform

Conclusion

The Controller Area Network has been designed and constructed by two nodes to test the CAN protocol. By testing, the data could be sent correctly between two nodes and vice versa. Waveform and timing were verified for correctness of the testing . CAN improves noise rejection as the differential signal. So the Controller Area Network can be used reliably in long distance communication.

Acknowledgements

This project was fully supported by Department of Physics ,Yangon University, Myanmar (2011-2012).

Reference

Bosch R 1991 "CAN Specification, version 2.0", (Stuttgart: BOSCH)

Eady F 2004 " Networking and Internetworking with Microcontroller" (New York: Newnes)

Ilett J 1998 "How to use intelligent L.C.D.S", (London: Winborne Publishing Ltd)

Pazul K 1999 "Controller Area Network (CAN) Basics, DS00713A" (New York: Microchip)

Richards P 2000 "A CAN Physical layer Discussion, DS00228A" (New York:, Microchip)

Stallings W 2000 "Data and Computer Communications" (New Jersey: Prentice-Hall)

http://www.eetime.com/design/embedded/4023975/Serial-Protocols-Compared

http://www.ems-wuensche.com/Controller-Area-Network. html

http://www.gaw.ru/data/interface/CAN-BUS-PDF.

http://www.ixxat.com/can-controller-area-network-introduction-en.html

http://www.pdfdata.datasheetiste.com/pdf1/Microchip/PIC18LF458T-I-PT.pdf

Contents

Page

Yangon Institute of Education
Research Journal, 2012
Vol. 4, No.1