IMPLEMENTATION OF CAN BUS IN AN AUTONOMOUS ALL-TERRAIN VEHICLE

Likhana Latheef. T
Department of ETE
Bharath University, Selayur, Chennai- 73
Email- likhanalatheef6@gmail.com

Monica Devi
Department of ETE
Bharath University, Selayur, Chennai- 73
Email- monicadevi@gmail.com

Prathyusha V.S
Department of ETE
Bharath University, Selayur, Chennai- 73
Email- prathyushasrinivasan@hotmail.com

Dr. T. Saravanan, PhD
Department of ETE
Bharath University, Selayur, Chennai- 73
Email- saravanan@gmail.com

Abstract

The main purpose of this effort is to design an autonomous all terrain vehicle with the help of a CAN bus. The operation and advantages of the CAN protocol in automobiles has been described. The PIC microcontroller is used here and a CAN controller which creates a single two wire bus through which electronic control units (ECU) in the automobiles communicate. In this paper there two CAN slave nodes and a main control node. The ATV is controlled using Radio frequency technique. GPS receiver is used to find the location of the robot and the latitude and longitude value will be displayed in the LCD in the main node. The ATV is connected to a slave node 2 which is controlled using H-bride driver circuit. The slave node 1 has the range sensor and GPS receiver connected to it.

I- INTRODUCTION

The recent technology trends in the automobile industry are bringing more comfort in a vehicle by incorporating automation techniques like collision, advanced safety features, entertainment devices and lot more. As the technology is developing, the use of electronic control units (ECU) in vehicles is increasing rapidly, making the communication between them very complex. Multiplexed communication was eventually developed to decrease the interconnections (cables) and the complexity between the ECUs. But the multiplexed communication has not met the real time communication requirements. In 1980s, a multi master serial communication protocol called Controller Area Network (CAN) which can be used in real time and also reduces the amount of wires was designed by BOSCH. The CAN protocol which is widely accepted in automobiles due to its real time performance is an asynchronous serial communication protocol and also follows ISO 11898 standards, it is reliable and compatible with wide range of devices as well. Owing to the wide spread of android (open source and Linux based OS), anything can be possible with suitable applications.

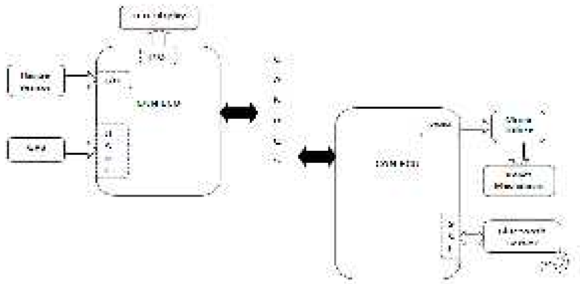
II. EXISTING SYSTEM

The ATV used for this design is a Honda Four Trax Rancher AT and a brief explanation of the parts in the ATV is provided. The Renesas RX62N is used as the microcontroller and a CAN controller which is a two cable bus through which electronic control units (ECU) in the automobiles communicate. The working of the CAN protocol has been explained with the help of a pseudo code

III. PROPOSED SYSTEM

The main purpose of this effort is to design an autonomous all terrain vehicle which uses a CAN bus. The operation and advantages of the CAN protocol in automobiles has been described. The PIC microcontroller is used here and a CAN controller which creates a single two wire bus through which electronic control units (ECU) in the automobiles communicate. In this paper there two CAN slave nodes and a main control node. The ATV is controlled using Radio frequency technique. GPS receiver is used to find the location of the robot and the latitude and longitude value will be displayed in the LCD in the main node. The ATV is connected to a slave node 2 which is controlled using H-bride driver circuit. The slave node 1 has the range sensor and GPS receiver connected to it.

BLOCK DIAGRAM



CIRCUIT DIAGRAM

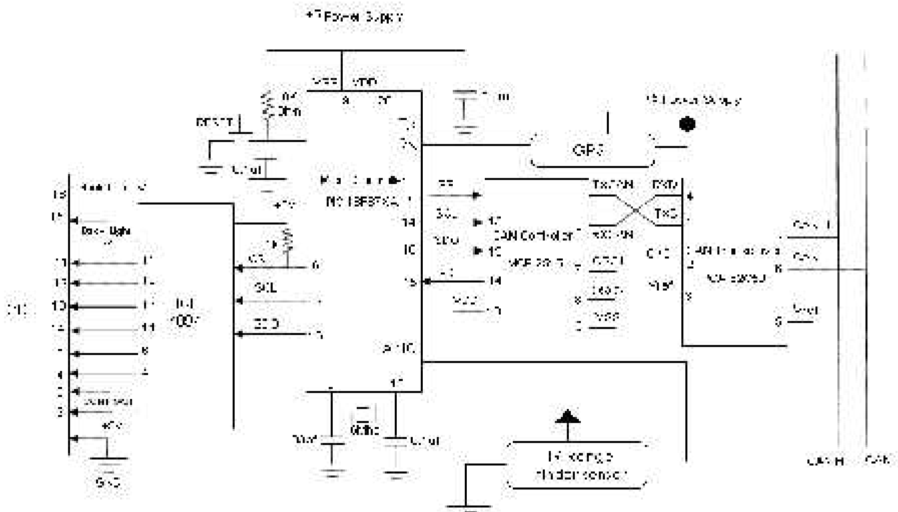


Fig 1.2 Circuit Diagram

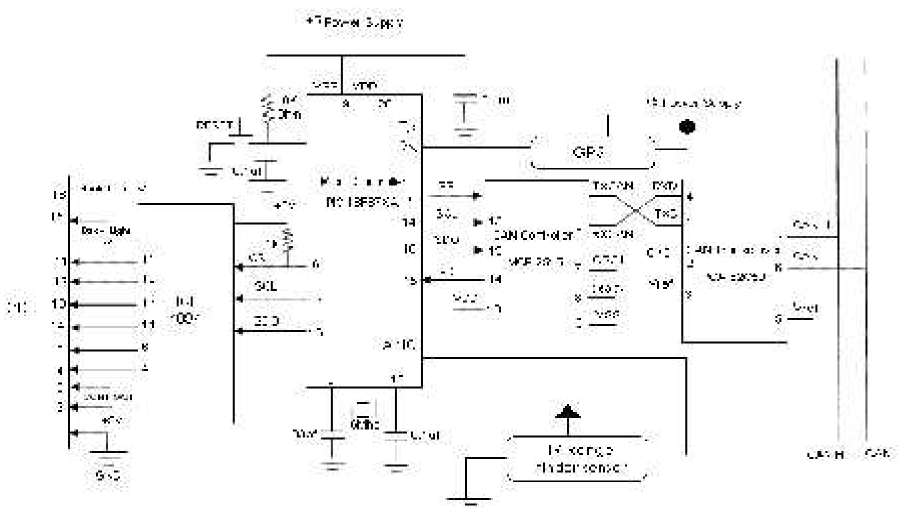


Fig 1.3 Circuit Diagram

IV- HARDWARE

PIC MICROCONTROLLER

PIC (Peripheral Interface Controller), which is made with Microchip Technology is a family of modified Harvard architecture microcontrollers, derived from the PIC1650.

Microcontroller Core Features:

•High-performance RISC CPU
•Very few instructions to learn
• Single cycle instructions and a two cycle instruction for program branch are used
•Operating speed: DC - 20 MHz clock input DC - 200 ns instruction cycle
•Interrupt capability (up to 14 sources)

• Eight level deep hardware stack

• Direct, indirect and relative addressing modes

• Power-on Reset (POR)
• Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
• Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
• Programmable code-protection
• Power saving SLEEP mode
• Selectable oscillator options
• Low-power, high-speed CMOS FLASH/EEPROM technology
• Fully static design
• In-Circuit Serial Programming (ICSP) via two pins
• In-Circuit Serial Programming capability is 5V
• In-Circuit Debugging via two pins
• Processor access to program memory is read/write
• Wide operating range of voltage : 2.0v-5.5v
• High Sink/Source Current: 25 mA
• Low-power consumption

CAN BUS

Controller Area Network (CAN) was initially created by German automotive system supplier Robert Bosch in the mid-1980s for automotive applications as a method for enabling robust serial communication.

CAN OVERVIEW

Most network applications follow a layered approach to system implementation. This approach enables the exchange and use of information between products from different manufacturers. An ISO standard was created for this approach as a template to follow.

Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)

The CAN communication protocol is a CSMA/CA protocol. The CSMA stands for Carrier Sense Multiple Access. This means that all the nodes on the network must monitor the bus for a period of no activity before trying to send a message on the bus (Carrier Sense). Also, when the no activity period occurs, any node on the bus will have an equal chance to transmit a message (Multiple Access). In CAN protocol, a nondestructive bitwise arbitration method is used, that is, the messages remain undamaged after arbitration is completed even if collisions are detected.

CAN define a logic bit 0 as a dominant bit and a logic bit 1 as a recessive bit. Logic bit 0 will always win arbitration over logic bit 1, therefore the priority of the message increases when the value is low.

Message-Based Communication

CAN protocol is a not an address based protocol but a message based protocol. This implies, messages are not transmitted from one node to another based on addresses. The priority and data to be transmitted are embedded in the CAN message . All message transmitted on the bus are received and acknowleged by the nodes in the system. Later, each node will decide as to how soon the received message should be discarded or retained for processing.

Remote Transmit Request (RTR) is another useful feature built into the CAN protocol which allows a node to request to get information required from other nodes. This is different from the example in the previous paragraph because this node does not wait for information from a particular node, instead it requests data to be sent to it specifically. For example, a safety system in a car gets frequent updates from critical sensors but it does not receive frequent updates from other sensors like the oil pressure sensor or the low battery sensor to make sure they are functioning properly.

(i) CAN Message Frame Description

Standard Data Frame

The frame begins with a start of frame bit like all other frames and is of the dominant state. SOF allows hard synchronization of all nodes.

Arbitration field:

The SOF is followed by the arbitration field. Arbitration field consisting of 12 bits; the 11-bit identifier and the Remote Transmission Request (RTR) bit. The RTR bit is used to distinguish data frame (RTR bit dominant) from a remote frame (RTR bit recessive).

Control field:

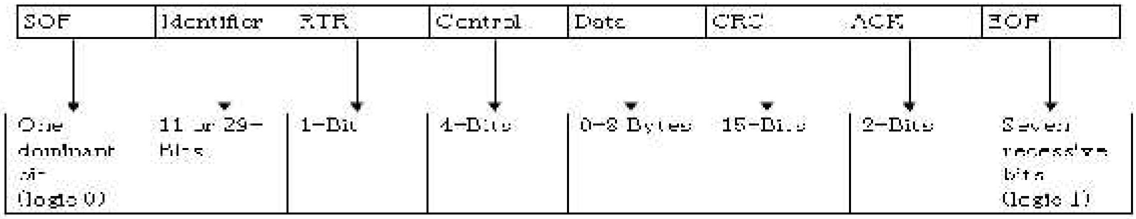
Control field Consists of six bits - the first bit of this field is the Identifier Extension (IDE) bit which must be dominant to specify a standard frame. The following bit, Reserved Bit Zero (RB0), is reserved and is defined to be a dominant bit by the can protocol. The the other four bits of the control field are the Data Length Code (DLC) which specifies the number of bytes of data contained in the message.

Data field:

This contains any data bytes that are being sent, and is of the length defined by the data length code, above 0-8 bytes.

Cyclic Redundancy Check (CRC) Field

It follows the data field and is used to detect transmission errors.



Error Frame

An Error Frame is generated by any node that detects a bus error.

Overload Frame

It is used to request delay

Errors Detected

The CAN protocol is highly reliable and error resistant. The following series are used to detect errors

.

- CRC Error
- Bit monitoring
- Checksum check
- Bit stuffing
- Frame check
- Acknowledgment check

(ii) Interframe Space

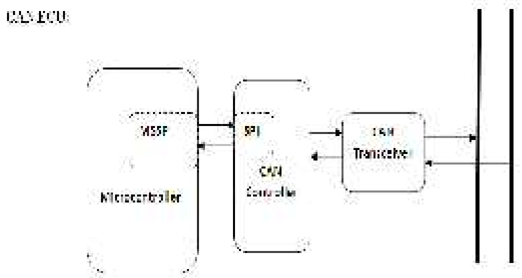
Interframe Space is defined as the delay between the two message frames

(iii) Fault Confinement

If the node will detect any error or if it is unable to read the message then the entire bus will know about the errors and it will be retransmit the message again. If higher priority message is corrupted then, as it is higher priority the low priority messages won't get the bus access. In such cases the CAN protocol will disconnects this node from the Bus. If any node is reported as error then, for some time the node will be disconnected from the Bus and it will be in bus off state.

CAN Module

Messages are transmitted and received on the CAN bus with the help of the CAN module.



(V) RESULT

This project describes about implementing the CAN bus on automated vehicles. The operation of the CAN protocol has been tested on PIC Microcontroller. As the CAN protocol is compatible with many of the devices it can be implemented in any of the embedded systems for real time transmission of data with less number of interconnections and large number of devices to communicate

(VI) CONCLUTION AND FUTURE WORK

This research work proved that we can design an autonomous All-Terrain vehicle by implementing CAN bus. The operation of the CAN bus is tested using the PIC Microcontroller. In this we are using two nodes which is being connected by using the CAN bus. Since it uses bus for inter connection it will decrease the number of wires. The Can bus will increase the processing power. In this project we are also using a GPS Modem, LCD Display, Range Sensor, Motor Driver, Bluetooth Device and a DC Motor. This device is controlled by using Bluetooth. We are using a Bluetooth device to control the movement of the vehicle. The GPS will find the location of the vehicle (GPS value will displayed on the LCD display) and the range sensor will detect the obstacle hence we can control the vehicle by using a Bluetooth. Motor driver is used to control the movement of wheels.

The future of this technique is very bright. It can be make it as an automated vehicle by using a camera and LIDAR. At present the steering of the All-terrain vehicle is of steady state error, hence the Proportional Integral Derivative (PID) controllers are able to control the steering this PID is used to increase the time response. There is a Bluetooth device which is already implemented to stop the vehicle but it doesn't apply the brake if the radio signal is either not detected or an emergency switch on the Bluetooth device is thrown and also this current stop function is implemented in the software even though an emergency stop function should be implemented separately then should shut down the engine and can apply the brake. And the rotary encoders can be used as a backup in case of GPS outages. This GPS is helpful to find the accurate location of the latitude and the longitude value for short distances and are often sampled at a faster rate than most GPS's.

(VII) REFERENCES

(I) www.barrproup.com/Embedded-systems/How-To/Can-vs-SPI

(II) Wilfried Voss, A Comprehensive Guide to Controller Area Network

(III) http://www.technologyuk.net/telecommunications/industrial networks/

(IV) Renjun Li, Chu Liu and Feng Luo, A Design for Automotive Can Bus Monitoring system

About Authors:

Dr. T. Saravanan PhD is currently working as the H.O.D of Electronics and Telecommunication Department at Bharath University, Selayur, Chennai

Likhana latheef. T is currently pursuing B.Tech in Electronics & Telecommunication Engineering from Bharath University, Chennai. She is doing her project in Implementation of CAN bus in an autonomous All-Terrain Vehicle under the expert guidance of Dr.T.Saravanan at Bharath University

Prathyusha.V.S is currently pursuing B.Tech in Electronics & Telecommunication Engineering from Bharath University, Chennai. She is doing her project in Implementation of CAN bus in an autonomous All-Terrain Vehicle under the expert guidance of Dr.T.Saravanan at Bharath University

Monica Devi is currently pursuing B.Tech in Electronics & Telecommunication Engineering from Bharath University, Chennai. She is doing her project in Implementation of CAN bus in an autonomous All-Terrain Vehicle under the expert guidance of Dr.T.Saravanan at Bharath University