CAN COMMUNICATION MODEL TO HANDLE VEHICLE CONTROL SYSTEM & ITS APPLICATIONS USING RASPBERRY PI

S. N. Chikhale[1*], S. C. Hate[2]

[*]*Corresponding author:-*

Abstract:-

CAN communications helps us to engrave the complex wired source into the simple form. Using raspberry pi CAN bus system can work effectively to access the vehicle internal ECU parameters and now-a-days different applications with IOT compliance. We can control the vehicle system remotely using the raspberry pi application. Also, not only for the system of vehicle we can control the home equipment's using this technique. So automation which is the need of future can be effectively managed using the CAN with IOT convergence. We suggest some basic implementation of such scenarios over this paper.

Keywords: - *CAN (Controller Area Network), Raspberry pi 3, Automation, Internet of*

INTRODUCTION

Controller area network (CAN) was developed in 1986 by Robert Bosch GmbH which is a method of communication between electronic devices. Basically, this system was introduced to automate vehicle monitoring system. CAN is a message based serial field bus system where, it reduces the complexity of vehicle electronic/wiring system. Automation is a need of this era where we can handle the machine work remotely using the Internet of Things (IOT) technology which is booming all around the world. Thus in this paper we proposed to develop system which can introduce the automation in automobile industry using raspberry pi module.

We know about the CAN which is robust and vehicle based standards where application layer, object layer gets decomposed. Also, this bus can eliminate the need of host computer which is really a commendable work introduced in this standards. Due to this, the possibilities of research in this field get increased day by day and the convergence of this using a IOT is possible. A MultiMate serial bus standard for connecting electronic control units (ECU's) which is also known as nodes can communicate with this network. Using specific software and standards we can move towards the complex application soon.

The system can communicate with speed i.e. nothing but the acceleration, breaking and temperature all around the vehicle. A specific node will transmit the data to the CAN bus and another one will receive the output on display unit as Graphical User Interface (GUI). Remote server will direct the data to the user's device which helps user to monitor the vehicle efficiently.

So using remote monitoring user can handle or hack the vehicle for acquainted use.

System Design

System design is the important part where we connected the raspberry pi as a single board computing device which is connected to the internet and CAN bus application to monitor vehicle electronics system. Interconnected network i.e. internet is connected with centralized server and remote located user devices like PC, Tablets, Mobile devices etc. CAN bus is completely explored with its own integrated micro-controller and trans-receiver to communicate it with our system. Various electronic control system is attached to the CAN bus inside the vehicle which helps us to the run the project on-board.

All the excepted/ necessary information is gathered across single board computer which is on-board and is connected to remote internet which can store/analyses the data from server effectively. CAN bus is so organized that international standardization organization (ISO) defined it as a serial communication bus ISO 11898, originally developed for automotive industry replaces the complex wiring harness with a two wire bus. The signaling rate of CAN communication is as good as 1Mbps, with high immunity to electrical interference and an ability to self-diagnose with repairing data errors. So this protocol not only bind to automotive but also works for marine, medical, manufacturing and aerospace.

Raspberry pi as a SBC

Single Board Computer is the main stream of controller where all the processing activities is carried out according to the fundamentals provided by the inputs for the vehicle driven techniques. Hence eventually all the necessary inputs are provided to the raspberry pi 3 and all the processing activities are maintained through the super-board having the capacity of small intelligent computer. It is having around 40 low-level peripheral pins, called GPIO interface which are connected to the on-board BCM2386 chip's GPIO pins.

We can access this over the Ethernet as it consists of SSH client i.e. secure shell element. Its consists of linux –kernel based operating system known as Linux distribution raspbian based on debian. It is small computing device so does not have any CAN bus support. It requires special SPI pins to connect with a CAN bus using CAN controller and CAN transceiver. For connecting CAN with this raspberry pi we required special drivers known as MCP251x. which is to be installed in kernel of the raspberry pi. While latest raspberry can have inbuilt microchip CAN controller which reduces our work drastically.

Raspberry pi is basically developed by raspberry pi foundation in United Kingdom to teach basic computer science. The first generation raspberry pi was released in 2012 and now we have raspberry pi 3 model B which is far more superior and having much capabilities than first generation one. Without using high computational machine, we can use low cost, affordable and efficient one raspberry pi module to develop such kind of applications in the organizations. This raspberry pi 3 B is capable of on-board Wi-Fi 8002.11n and Bluetooth capabilities in recent design whereas, Broadcom provides its SoC (system on a chip) with an integrated ARM compatible to central processing unit (CPU) and on chip graphics processing unit (GPU). Processor speed is 1.2 GHz and 1 GB RAM as an on-board memory. Also it consists of 4 USB inputs and a power and HDMI cable with audio/video output. While SD card is operated as a basic memory storage of raspbian OS.
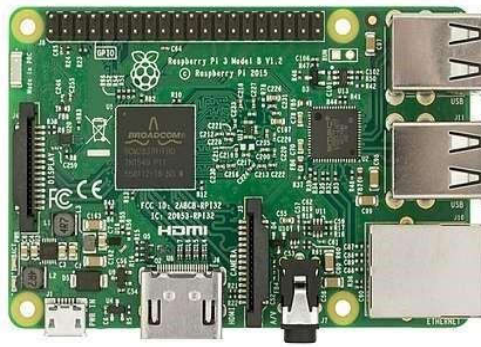
Fig. Raspberry Pi 3 Model B

CAN Controller with MCP251x

MCP251x is a Microchip technology's stand-alone controller area network that implements the CAN specifications in an organized manner. It is capable of transmitting and receiving both standards and extended data and remote frames. It consists of various acceptance masks and acceptance filters which are used to filter out unwanted message. MCP251x interfaces with microcontrollers via an industry standard serial peripheral interface. CAN module handles all functions for receiving and transmitting messages on the CAN bus. Massage buffer is first appropriately initiated by using control register bits via a SPI interface or by using transmit enable pins. Status and errors can be checked by reading the appropriate registers. Control logic and registers are used to configure the device and its operation. SPI protocol block is interfaced via a MCU device which is used to build standard read and write command.

We can use MCP2515 microchip as a CAN controller which is having transmission speed upto 1 Mbit/s. It handles transmission normally using both normal and extended CAN frame and work both with high speed and low speed CAN transceivers. This CAN controller is having two three transmissions and two reception buffer.

MCP2551 is a CAN transceiver which is capable to handle a high speed CAN device that serves as the interface between a CAN controller and the physical bus. ISO-11898 standard is fully compatible with CAN protocol controller which is having differential signaling and receive capabilities for MCP2551 device.
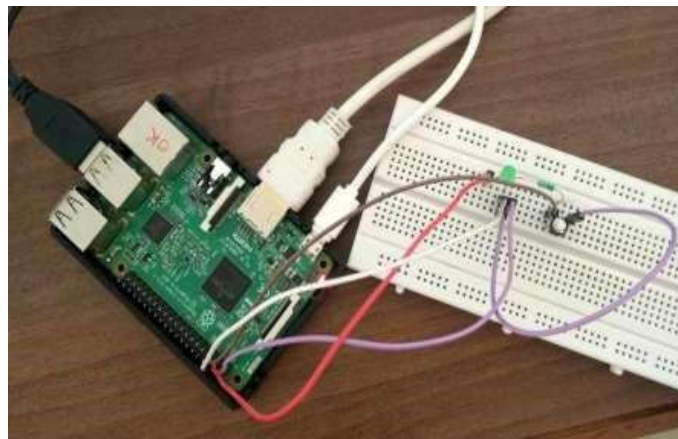

Fig. Designing Light sensor using raspberry pi 3 model B

Software Design

All software part is designed according to our SPI application using CAN controller on raspberry pi. Raspberry pi is loaded with linux kernel version with raspbian OS where we required to support the python language as a basic programming language. So based on the platform we have entitled to amalgamate the entire sensor section into the CAN to get it through the raspberry pi to control our devices efficiently. As we know we have to provide specific drivers for the SPI access to the module. So bcm2708 drivers we have to install on our board. Driver MCP251x is the special purpose drivers for CAN controller MCP2515.

Here in raspberry pi we have different GPIO pins which we have to assign for the specific tasks. So, we have to assign light, speed and temperature inputs to the system to get a desired outcome.

Pin 1 is assign for 3V power where pin 3 will be assigned for GPIO1 i.e. inputs to different set of controllers. Also, pin 6 will be ground for complete circuitry. We have to program the raspberry pi according to the input pins and provide commands accordingly.

```
e.g.- import RPi.GPIO as xyz

    import time

    xyz.setmode(xyz.BOARD)

    xyz.setup(3, xyz.OUT)

    xyz.setup(5, xyz.IN)

    while(1):

        if(xyz.input(5)==1):

            xyz.output(3,xyz.LOW)

        else:

            xyz.output(3,xyz.HIGH)
```

Conclusion

Using the effective nature of CAN communication now we can reduce the complexity of wired system and converge it with the IOT appliances in our daily routine to implementation of some system environment. System speed measurement, light applications and break status can be changed using the above application efficiently. Also, even linking it to the world of interconnected network we can track the details in real time if time permits us with the better connectivity options. So rather doing it for the on-board device compatible we can do the IOT in this form of application and even can some applications remotely using the raspberry pi 3. So, proposing such a scheme under the basic processing unit might be a challenge once again but will be the future research for this initiative and as we have to get the things for the effective IOT channel, it will be the innovative form of technology as we are having lots of demand for IOT applications.

References

[1].A. Salunkhe, Pravin P Kamble, Rohit Jadhav, 2016. Design And Implementation of CAN bus Protocol for Monitoring Vehicle Parameters. IEEE International Conference on Recent Trends In Electronics Information Communication Technology, 301-304.

[2].Hanxing Chen, Jun Tian, 2009. Research on the Controller Area Network. IEEE International Conference on Networking and Digital Society, 252-254.

[3].H.Ekiz, A.Kutlu, M.D.Baba, E.T.Powner, 1996. Performance Analysis of a CAN/CAN Bridge, IEEE, 181-188.

[4].S. Vijayalakshmi, 2013. Vehicle control system implementation Using CAN protocol, IJAREIE, 2532-2538.

[5].Kumar, M. A.Verma, and A. Srividya, Response-Time "Modeling of Controller Area Network (CAN). Distributed Computing and Networking, Lecture Notes in Computer Science Volume 5408, p 163-174, 2009.

[6].Tindell, K., A. Burns, and A.J. Wellings, Calculating controller area network (CAN) message response times. Control Engineering Practice, 3(8): p. 1163-1169, 2005

[7].Pazul, "Controller Area Network (CAN) Basics", Microchip technology Inc., AN713, May 1999.

[8]. B.Gmbh, "CAN specification" vol 1 Version 2.0, 1991.

[9].Wilfried Voss, A comprehensive guide to controller area network, Copperhill Media Corporation, 2005-2008.

[10]. Patrick Ciccarelli, et. al, "Networking Basics", 2nd edition.