Autonomous control of CAN bus to control automobile through mobile

Godly K Thomas

ABSTRACT: Automotive electronics is the field which have evolved over the years. Earlier we used to have majority mechanical parts in a car but with government stressing more on the rule that companies must adhere to EURO VI (European emission standards)[1] norms it has gradually shifted towards electronics covering sections like infotainment of a car to a basic engine(fuel injector).As we go ahead with this revolution, this paper focuses on a cheap and viable way to control your car through mobile. To keep it simpler, this paper will tell you how to interface a microcontroller (ATMEGA328) with CAN [2] (Controller Area Network) bus, hence communicating with the car and performing operations (ignition ON, operations related to HVAC system).This project will provide a viable, cheap and efficient substitute to the various technologies present such as passive entry scheme, remote controlled entry (RF signal) which is costly. This method is much safer because it will use a GSM modem [3] and hence the message to operate the car is only with the user making it impossible for any other person to use it without user's consent. It can also have a GPRS module which tells the position of the car to the user whenever the user demands to, making it a product which supplements the needs of todays technologically dominated world.

Index Terms— CAN, EUROVI Standards, ATMEGA328 microcontroller, GSM, GPRS, CAN shield, MCP 2551 transceiver

— — — — — — — — ◆ — — — — — — — —

1. INTRODUCTION

There is no doubt that scientific advances depend not only on new ideas, conceptual leaps and paradigm shifts, but also to a large extent on technological advances that make these steps possible. Technologies that are easily adapted to simple and affordable everyday use have certainly changed the speed of scientific progress. When we talk of automobile section it's no different. The ACC (Automatic Cruise Control)[4] technology for example have made the drives simpler on highway and it just requires a button press to switch between your speeds. The technology incorporated in this industry is eventually driving itself towards autonomous vehicles.

Because of the technical advances across sub disciplines within this industry it is imperative to move forward. Earlier we had simple keys to access our car which was then replaced by remote control access. Currently we have passive entry scheme which also needs a key. This paper will device a method which would eliminate the use of key and uses a GSM modem to receive a message from the user and one could perform all functions in a car. Each operation of a car would be done by a particular encrypted message which would be only known to a user. When it receives a message it communicates to the CAN bus through CAN shield interfaced with ATMEGA328 microcontroller sending a unique message adhering to the CAN protocol.

———————————————

• *Godly k Thomas is currently working in Fiat Chrysler India ,
   PH9791968095 E-mail: godlythomask@gmail.com*

We are using standard frames for communicating with the CAN bus. We should know the addresses of each ECU we want to communicate with in CAN bus which is a two byte address. When communication starts in CAN bus each ECU starts with the arbitration process and every message has a destination node address source address, message data and size of the data sent. Communication in CAN bus is a broadcast communication which means the data sent is transmitted to each node. Each node can have a filter for a particular message and uses that particular message. We are using a CAN shield which has five filters and two masks

2. MOTIVATION

The motivation of this paper is to create a safe and technologically acceptable device when it comes to user using his car. When a user loses his keys, it may result in hefty financial losses to user. Since this device is actually implemented with the use of mobile phone, it is very easy to handle.

Passive Entry Scheme which is currently used in market uses RF frequency but the user must have the key in its vicinity and adding the fact that it is a costly feature, this feature is actually an excellent replacement of the current method used. Once the user sends a message, within fraction of a second it is received by the GSM modem which then communicates to the microcontroller sending the corresponding data serially to the CAN shield and hence performing the required operation. It can only accept specific messages hence making it very difficult to decode a particular message selected by the user himself. The user can change the password and request can be send to the user every month in order to make the system impenetrable.

3. PROPOSED SYSTEM OVERVIEW

In Figure1 we can see how the proposed system actually works. In this diagram the message is first sent to the GSM modem by the users mobile. After the reception of that message it will send the corresponding CAN message to the CAN bus for example if we want the different stages of Ignition we can send the different values for messages. Figure   shows the different values for ignition states which is broadcasted on the CAN bus. The second column defines the different stages of Ignition.

Value	State of Ignition
0X0	IGN_LCK
0X1	IGN_OFF
0X2	IGN_ACC
0X3	IGN_OFF_ACC
0X4	IGN_RUN
0X5	IGN_START
0X7	SNA

Figure: 1

This message is sent as a standard frame to the CAN bus. After receiving a particular message, each ECU will behave accordingly. Every message is routed through CBC which acts as a gateway between CAN high and CAN Low. CAN high has a speed of 500kbps baud rate and CAN low has 250 kbps speed. CAN High is used usually for messages which requires fast response. CAN shield has MCP2515 transceiver which has five filters and two masks in order to receive the desired signal. There is a time stamp calculation for every signal with a CAN ID which is decided based on the arbitration process when the CAN communication takes place. When we send any message it will be first send to ECM which will route it to proper destination according to the message given in the CAN message.

.

4. SIMULATION RESULTS

Figure2 refers to the CAN communication which takes place in a car. Figure2 is the trace of the CAN bus communication which is taking place in CAN I and CAN C simulated through vector Canoe. Every ECU can be simulated on this software and each ECU sends virtual signals over the bus. In this figure we can see the length of each message transmitted and the address of different node with the specification whether it is a transmitter or receiver.
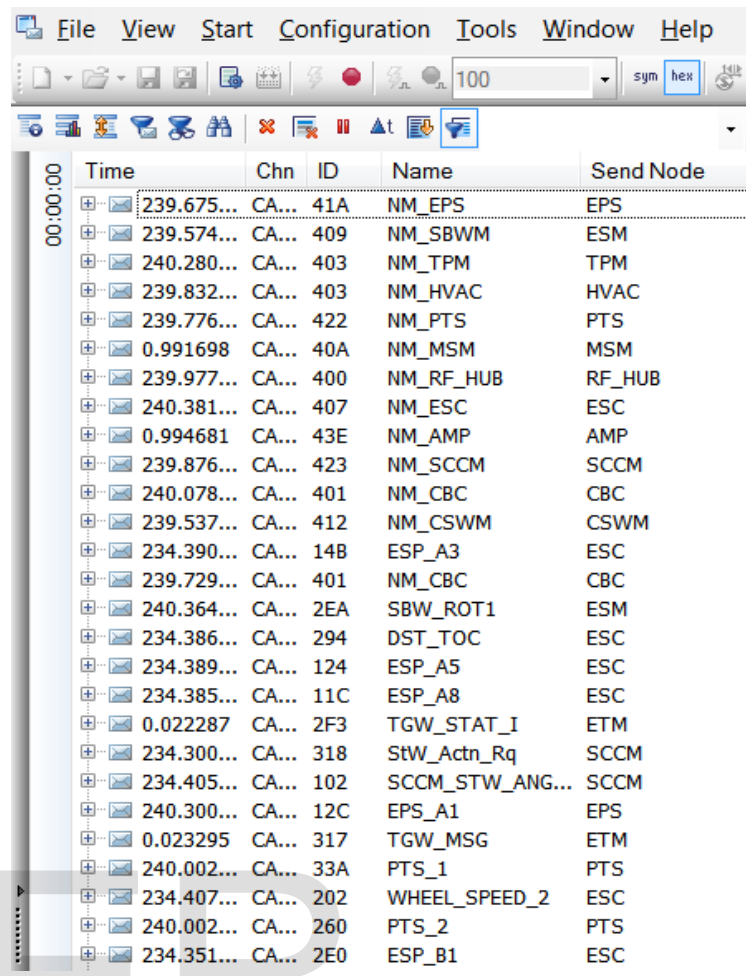

Figure: 2

Each message has corresponding signals whose nomenclature is given by the user. Each message can correspond to both CAN high and CAN low with CBC acting as a gateway. Each signal has a different value at different time which depends upon the operation that is simulated. In vector we could simulate any number of panels and for that we need to have local variables for each operation. This local variable transmits information whenever any change is made. For example, if we want to increase the speed virtually we need to have a panel with a speed variable which could transmit the changed value of speed to the CAN bus. Figure 2 tells the transmitter of each signal and also the signal attribute if we want to check the different values for each signal. Each variable value can be explored through symbol explorer which tells you the current value of a particular symbol. Each operation can be automated if we load an xml file telling the variables to change their signals according to the requirement. All operations in a vehicle can be simulated through software but certain things such as Ignition off draw tests requires actual hardware.

5. PROGRAMMING CODE

Code to send message in CAN BUS:

SENSORPIN-input from GSM modem

```
//a sample signal is sent in the code
#include <mcp_can.h>
#include <SPI.h>

int sensorPin = 12;
```

```
void setup()
{
  Serial.begin(115200);

  if(CAN.begin(CAN_100KBPS) ==CAN_OK)

Serial.print("can init ok!!\r\n");

else Serial.print("CAN initialization fail");

}

void loop()

{
 sensorValue = digitalRead(sensorPin);

unsigned char canMsg[1] = {0X05};

if(sensorValue==HIGH)

{
  CAN.sendMsgBuf(0x0FF, 0, 8, canMsg);
}
}
```

Code to send message to GSM shield

```
#include <GSM.h>
#define PINNUMBER ""

GSM gsmAccess;
GSM_SMS sms;
char remoteNumber[20];

void setup()
{
    Serial.begin(9600);
    Serial.println("SMS Messages Receiver");

  boolean notConnected = true;

  while(notConnected)
  {
if(gsmAccess.begin(PINNUMBER)==GSM_READY)
    notConnected = false;
   else
   {
    Serial.println("Not connected");
    delay(1000);
   }
  }
  Serial.println("GSM initialized");

 Serial.println("Waiting for messages");
}

void loop()

{
 char c;
 if (sms.available())
 {
   Serial.println("Message received from:");

 sms.remoteNumber(remoteNumber, 20);

   Serial.println(remoteNumber);

   while(c=sms.read())
```

```
    Serial.print(c);

   sms.flush();

   Serial.println("MESSAGE DELETED");
 }

  delay(1000);
}
```

Integrating both these codes we could send a CAN message whenever a particular message is received by the GSM modem. It is up to user to code it accordingly for example if user wants to have an ignition on, he can send a message '1' .On receiving this message, CAN shield will send the corresponding CAN message in the correct format telling the ECM to start the ignition. The board has six digital pins in total which a user can configure it accordingly to be an input or output. GSM shield can also be connected to internet as well as send a message to a recipient according to the requirements. The code given for GSM shield is open source and could be altered to receive any signal. The code is flashed on board using USB connection to the PC .The board has MAX232 IC which brings the TTL compatibility since the board needs a Transistor-Transistor Logic. Code must be debugged before flashing on the ATMEGA328 controller. After flashing the code, it is ready to use. It must be kept in mind that user needs to mention the number correctly through which it will receive the message .On receiving the message from mobile ,it should send a message to CAN shield instructing the CAN bus to perform the needful. The message sent could be in standard form or the extended format depending on the number of bits used. Total number of bits used for extended frame is 29 bits whereas for a standard frame it is 11 bits containing the data ID and message to be sent. Data ID is decided by the arbitration process when the CAN communication starts. Data ID with more number of zeroes will get the highest priority. When the frames are sent in CAN bus and if three consecutive ones occur it will take automatically the next bit as start of frame. This is a protocol followed during the transmission of any CAN message.

6. HARDWARE IMPLEMENTATION

For implementing this project, I have used a CAN shield [8] (Figure: 3) supporting MCP2551 microcontroller [5] which is basically a transceiver supporting messages on CAN bus. GSM shield (Figure: 5) is used along with the microcontroller ATMEGA328 (Figure: 4) used in ARDUINO board to receive any message from the user.

Figure:3

Figure:4

The microcontroller is programmed to send signal to the CAN bus whenever it receives a specific message from the user. CAN shield used also has MCP2515 which is a second generation stand-alone CAN controller. It is pin and function compatible with the MCP2510 and also includes upgraded features like faster throughput, data byte filtering, and support for time-triggered protocol. It has 2 masks, six filters and five masks to support proper communication and the reception of specific messages

Figure: 5

The GSM modem after receiving the signal from mobile send a signal to the CAN shield and the message sent to the can bus is according to the code which is flashed on the ATMEGA328 microcontroller. The CAN shield which we use has also a joystick along with it which could be used too actually control any device through simple logics. It also has GPRS module which we can use efficiently to tell the position of the vehicle. It has a port compatibility for the LCD panel to display any information as per requested by the user.

7. CONCLUSIONS AND FUTURE SCOPE

This project gives a keyless method to control a car which is secure and economically viable. We can perform any operations we want according to the CAN message sent on the bus. On reception of the CAN message, ECM sends the corresponding signal to the associated ECU like ETM, IPC ICS etc.

Future scope: In order to make it more secure we can have GPRS module interfaced with the controller which could send the exact coordinate of the vehicle to the user whenever the user demands. This will help the user track his car even when he is not using it.

8. REFRENCES

[1] Euro III / Euro IV Emissions **-** Piotr Bielaczyc Jerzy Merkisz SAE Technical Paper1999-01-1073, 1999, doi: 10.4271/1999-01-1073.

[2] Introduction to the Controller Area Network (CAN) Texas Instruments SLOA101A–August 2002–Revised July 2008

[3] International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-3, July 2012 GSM based e-Notice Board

[4] International Journal of Information and Education Technology, Vol. 1, No. 1, April 2011 ISSN: 2010-3689Cruise Control Operation from Zero to Preset Speed-Simulation

[5]CAN communication for vehicle automation, IJETAE (ISSN 2250-2459, Volume 2, Issue 2, February 2014

IJSER