



Raspberry Pi for Commercial Applications

Chaitanya Bysani¹, T.S.Rama Krishna Prasad², Sridhar Chundi³

¹M.Tech, Embedded Systems, Gudlavalleru Engineering College, Gudlavalleru

E-mail id: chaitanyabysani88@gmail.com

²Associate Prof., ECE Department, Gudlavalleru Engineering College, Gudlavalleru

E-mail id: ad2prasad@gmail.com

³Program Manager-System Integration, CMC Ltd., CMC Center, Gachibowli, Hyderabad

E-mail id: sridhar.chundi@gmail.com

ABSTRACT

The objective of this paper is to create a low cost commercial off the shelf data analyzer for improving automotive safety and design a user interface infotainment system by using Raspberry Pi. In this paper we propose Raspberry pi based application that monitor the vehicle ECUs through an OBD-II(On Board Diagnostics) interface, perform Diagnostics with DTCs (Diagnostics trouble codes). Infotainment system having functions such as audio and video playback, games, internet connectivity through either USB Wi-Fi dongles or USB Modems and dashboard camera operation. Raspberry Pi will transmit the data over Wi-Fi in real-time in xml format over Wi-Fi on a DHCP connected network.

Index terms

Raspberry pi, Infotainment System, OBD-II, Diagnostics trouble codes (DTC).



Council for Innovative Research

Peer Review Research Publishing System

Journal: INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY

Vol 11, No.2

editor@cirworld.com

www.cirworld.com, member.cirworld.com

INTRODUCTION

In the past twenty years the scope of automotive software/Hardware based functionality has grown continuously. In recent years this has specifically applied in automotive infotainment systems and diagnostic systems. In modern automotive engineering having some important elements. Those are applied to design, manufacture and operation of automobiles, motorcycles, buses and respective engineering subsystems. Elements are mechanical, software, electrical, electronics and safety engineering. Automotive electronics are electronics used in automobiles. This includes car entertainment, telematics, chassis electronics and other body electronics. Mainly automotive electronics is started with the need for engine unit for Fuel transmission and consumption. Now a day's electronic system controls various functions like driver assistance, passenger safety and comfort, infotainment system, chassis electronics, transmission electronics, engine electronics and active and passive safety. Each modern car having more than 100 electronic control units.

In this paper we propose combining existing vehicle electronics with Raspberry pi to create a low cost commercial off the shelf data analyzer for improving the automotive safety and design a media center with internet connection. In our solution the total application can be divided into four modules. In first module the Raspberry pi was connected to On Board Diagnostics connector, infotainment systems like music system, information systems. In second module accessing car diagnostics data and operation of infotainment system through programming. In third module sending the car diagnostic data to specified locations through E-mail and in fourth module dashboard camera operation.

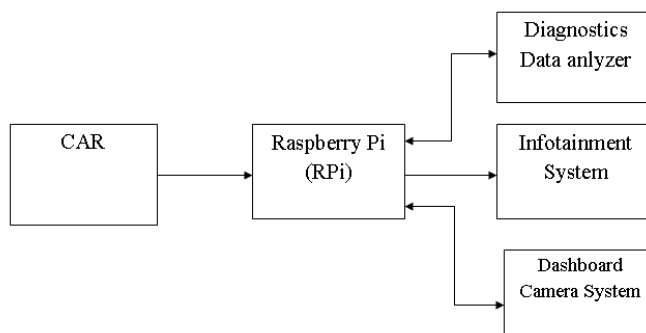


Fig 1: Block Diagram

Raspberry pi is a credit card sized computer. It is having Broadcom System on Chip that is BCM2835. System On Chip includes ARM1176JZF-S 700 MHz processor, 512 SDRAM, Video core IV Graphical Processing Unit. It is not having built in hard disk but it has SD card slot for long-term storage and booting. Raspberry pi uses Linux based operating systems. We have the option of six operating systems. According to our application we can choose the operating system and install on the SD card. Out of six operating systems Raspbian Operating system is useful for all applications.

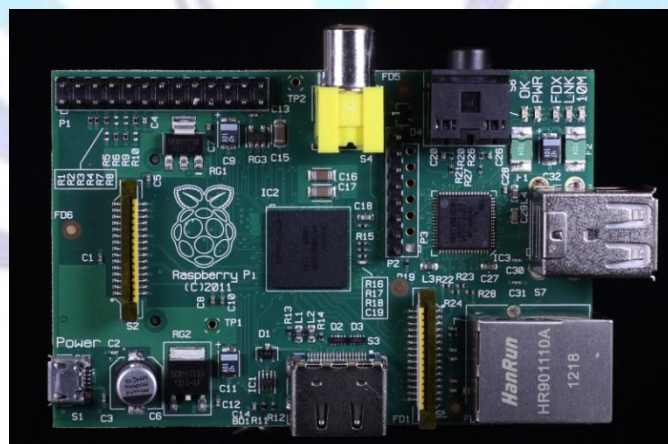


Fig 2: Raspberry PI Board

This paper is organized as follows: in the next section we briefly introduce the On board diagnostics (OBD-II). Our proposal to create low cost data analyzer and infotainment system through Raspberry Pi is presented and discussed in below sections.

ON BOARD DIAGNOSTICS (OBD-II)

The On Board Diagnostics regulations were developed in USA to detect engine problems and all other problems in a car. To conclude these problems, the system is uniformly monitoring engine management corresponding ECUs, gas emissions, fuel capacity, engine coolant temperature and other functions. When a problem is detected the system must store it such as EDR (Event Data Recorders) and Diagnostic Trouble Codes. So that automotive technicians may analyze it later on. The first generation of OBD, as known OBD-I defined only few parameters and failures were resulted in just a viewable warning to the driver and stored. The second generation of OBD, as known OBD-II defines different elements such as electrical signaling protocols, message format and it defines a different parameters that can be monitored and each parameter is having a specific code. These codes are called as Diagnostic Trouble Codes(DTC) is also defined in standard. Various operating modes are defined by the OBD-II to acknowledge for an easier communication with an system and defining the functionality. On-board diagnostic parameter ID's are codes used to get data from a vehicle. When automotive technician connects scan tool to the vehicles OBD-II connector and enter the parameter ID, then after the scan tool send to the vehicle bus. In vehicle the device on the bus will be recognizes the PID and reports the value to the bus. Scan tool reads those responses and displaying on the screen. Each Process ID (PID) having a minimum value and maximum value.

IMPLEMENTATION

The proposed application basically combines two things vehicle and raspberry pi to achieve a connection between both that is able to create a low cost data analyzer and infotainment system for vehicle safety and media center. The total application can be divided into three modules. In first module we have designed vehicle conditions detection is based on the parameters gave by the OBD-II connection. Such as engine coolant temperature, water coolant temperature, fuel level, tire pressure, indoor/outdoor temperature and acceleration accurate information gathered by raspberry pi. Initially the one end of the on board diagnostic cable is connected to vehicles OBD-II pin and another end of the OBD-II connector cable is connected to raspberry pi USB to get the data from vehicle bus. The information is packed and sent to an locations such as workshop, based on the DTC analysis workshop men/mechanics will get in touch with customer requesting to drop the vehicle to workshop to avoid ECU failures. This procedure is followed by an electronic mail.

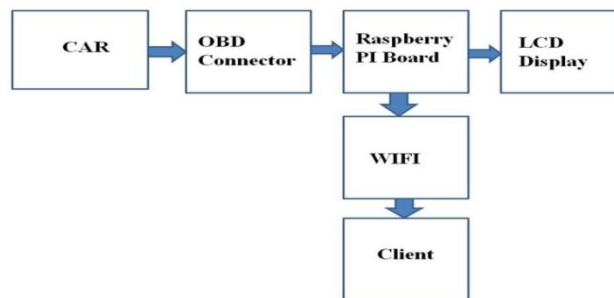


Fig 3: OBD-II Interface

The application also displays diagnostics information to the driver, including engine feedback data, speed levels, failures in mechanical elements, gas levels etc. The initial OBD-II connectors build on RS-232 or USB connections, now a days we find in market OBD-II connectors that provide wifi connectivity and Bluetooth. The price of these connectors are not high.

The second module is to build low cost dashboard video camera by using Raspberry pi and webcam along with FFmpeg and Motion. Motion is a program that monitoring the video signal from cameras and is able to detect if a particular part of the picture has changed. Motion is a command line tool. It has no GUI (Graphical User Interface). Everything is setup through command line or configuration file. Configuration is all done through the motion configuration file by entered the command is

```
Sudo nano /etc/motion/motion.conf
```

The output of the motion is mpeg video sequences, ppm format files and jpeg files. It has its own web server. We can access the webcam output from motion through web browser.

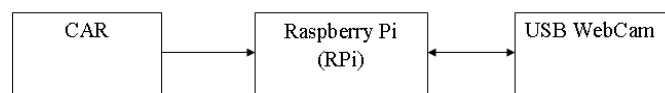


Fig 4: Dashboard camera Interface

FFmpeg is used to create videos from the captured images. It is having time elapsed video settings. One of the problem we faced while execution process doing is how to save the time elapse videos and individual captured frames created by motion, FFmpeg when raspberry pi is turned on and off. Previous videos and captured images are deleted when new motion is detected.

The main thing is to store each captured images and time elapse videos. We need a way to store the data created by FFmpeg and motion, before they are rewritten. To overcome this limitation. Listed few steps. First step is to change the default location where motion stores images and videos to other than a temporary directory and next step is to create startup script that will move images and videos to other location when restarting the raspberry pi. Last step is to configure the raspberry pi debian operating system to run this script startup before motion is starts.

The third module is to create a low cost infotainment system in a car. That means add a computer to a vehicle. For this application we are using raspberry pi, car DVD player, small TFT screen, video cable, audio cable, wireless USB dongle, mini key board and USB car charger. The 7" TFT screen can be placed in a dash board. Car DVD player can be placed in a dash board. DVD player and 7" TFT screen devices have RCA video input and for audio. We have 3.5 mm audio cable from the Raspberry pi to the auxiliary port of radio. So audio is played through car speakers. Raspberry pi placed in center console board of the vehicle and routed cables under the center console. Mini wireless keyboard with built in touchpad mouse was connected to the raspberry pi. So the total system now connects to any wireless network and also connects through mobile phone using Wi-Fi hot spot.

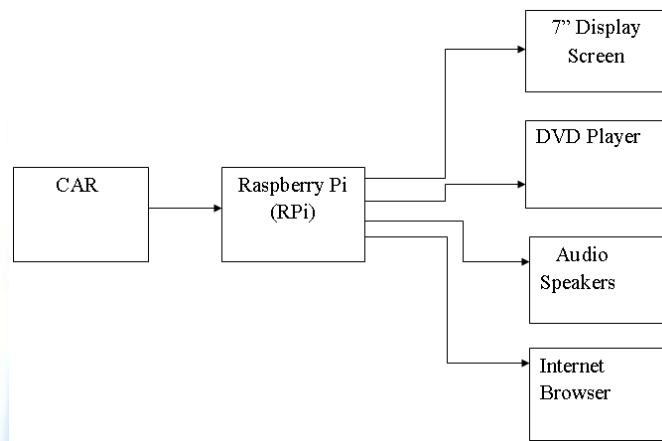


Fig 5: Infotainment System Architecture

RESULT

The below figures shows the raspberry pi in vehicle with mobile phone preview of dashboard web camera.

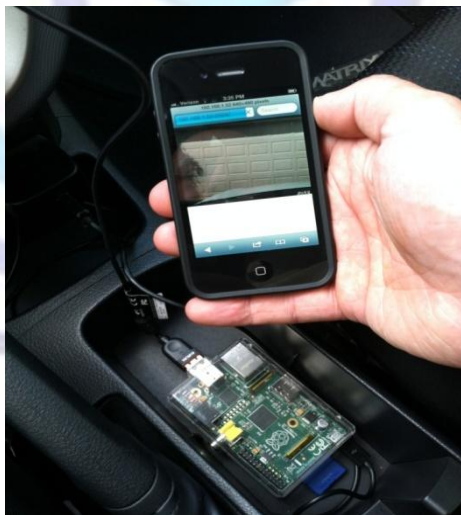


Fig 6: Snapshot of Experimental setup. Raspberry pi in Vehicle with preview of dashboard camera

The below figure shows motion output and it is monitoring through mobile web browser.

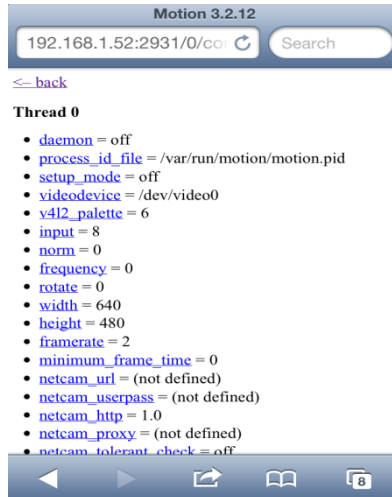


Fig 7:Using motion HTTP based control on mobile phone browser

The below figure shows infotainment system designed in car dashboard



Fig 8: Raspberry pi in vehicle with infotainment system

The below figure shows diagnostics reading taken from the vehicle through interfacing OBD-II to raspberry Pi.

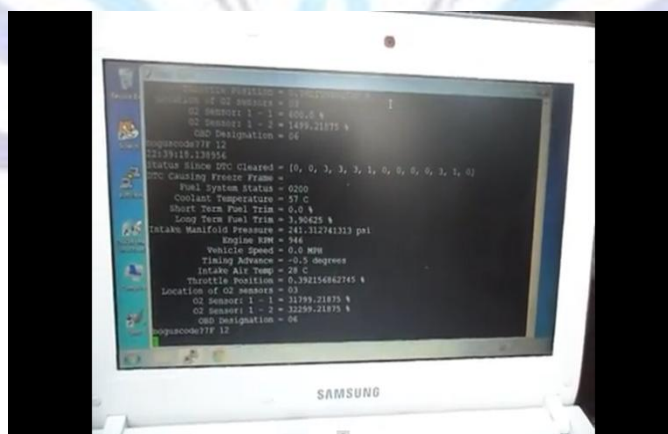


Fig 9: Snapshot of Diagnostics data

The below figure shows video streaming process output log screen

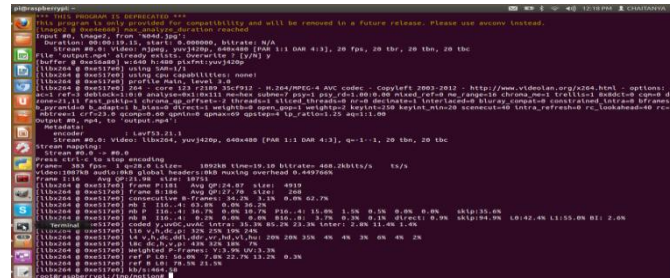


Fig 10: Snapshot video streaming process

CONCLUSION

In this paper, the low cost diagnostics data analyzer, infotainment system and dashboard camera in a vehicle are designed using raspberry Pi. We thought the raspberry pi is a perfect device for automotive applications because it is Low cost, small, less power requirements.

ACKNOWLEDGMENT

I express my sincere thanks and profound gratitude to my supervisor Mr.T.S.R.Krishna Prasad and co- supervisor Mr. Sridhar Chundi for his guidance and support.

REFERENCES

- [1] Matt Richardson and Shawn Wallace, Getting Started with Raspberry Pi (USA, O'Reilly Media, 2012)
- [2] BCM2835 ARM Peripherals, Broadcom, 2013
- [3] International Organization for Standardization, "ISO 15675: Road Vehicles, Diagnostics on Controller Area Network (CAN)", 2004
- [4] U.Hernandez, A.Perallos, N.Sainz and I.Angulo, "Vehicle on board platform: Communications Test and Prototyping", in Intelligent Vehicle Symposium (IV), 2010 IEEE, pp.967-972, 2010.
- [5] C.Spelta, V.Manzoni, A.Goggi and S.Savaresi, Smartphone-based vehicle-to-driver/environment interaction system for motorcycles, Embedded Systems Letters, IEEE, Vol. 2, no.2, pp. 39 – 42, 2010.

BIOGRAPHY



Chaitanya Bysani has completed B.Tech (ECE) and is pursuing M.Tech (ES) in Gudlavallur Engineering College, AP.



Mr. T.S.R. Krishna Prasad is working as Assistant Professor in Gudlavallur Engineering College, AP. So far he has 10.5 years of teaching experience and published 2 IJ paper, 3 NC and 4 IC papers.



Mr. Sridhar Chundi is working as Program Manager-System Integration in CMC Limited, Gachibowli, Hyderabad, Andhra Pradesh.