INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH & DEVELOPMENT

ISSN 2278 – 0211 (Online)

Android Based Digital Dash Board

Snehal Koparde
Assistant Professor, Department of Electronics and Telecommunication,
Marathwada Mitra Mandal's Institute of Technology, Pune, India
Manoj Gambhire
Student, Marathwada Mitra Mandal's Institute of Technology, Pune, India
Gaurav Chaudhari
Student, Marathwada Mitra Mandal's Institute of Technology, Pune, India
Rahul Gurav
Student, Marathwada Mitra Mandal's Institute of Technology, Pune, India

Abstract:
Modern vehicles are getting increasingly smarter and evolving to become new mobile computing platforms. As the software platform of vehicles advances, the demands for reconfigurable cars, installing and upgrading vehicle software as plug-and-play have also grown. However, the existing automotive software platforms are not designed for dynamic reconfiguration and plug-and-play. This paper addresses the plug-and-play challenge in the recent automotive systems. Drawing from recent advancements in the mobile phone operating systems, we propose an Android-based software architecture which supports the play and- play in automotive systems. In our architecture, an application can be downloaded from cloud services and executed on ECUs in a distributed fashion. Before installing an application, our system performs schedulability analysis to check real-time performance requirements for recent vehicles. We also present a model-based development tool for designing a workflow of automotive applications for our system.

Keywords: *Automotive, plug-and-play, middleware, workflow, software platform*

1. Introduction

Modern vehicles are getting increasingly smarter and evolving to become a mobile computing platform. Several tens of Electronic Control Units (ECUs) are already embedded into today's vehicles, forming a distributed network to control various functional components of a vehicle, from operating a windshield wiper to programming automatic parking. In Consumer Electronics Show (CES) in 2012, General Motors report their plan that they will programmatically export the real-time position data of a vehicle through the On Star service, which detect and avoid car crashes by sharing the position data through cloud computing services. Recent trend shows that more programmable functionalities are expected to be available in the vehicles in the near future. As the software platformization of vehicles advances, the demands for reconfigurable cars, upgrading software versions, or installing new applications as plug-and-play have also grown. However, the existing automotive software platforms in use are not designed for such dynamic reconfiguration and plug-and-play. AUTOSAR is a de-facto software architecture standard for automotive E/E (Electrics/Electronics) systems as it allows reducing the application development efforts by separating the application layer from the underlying infrastructure, increasing the reusability of existing functional modules. In AUTOSAR, however, the system software like an operating system and runtime environment of an ECU is statically built into the applications that run on the ECU. This necessitates the rebuild of the entire software stack each time, we upgrade or install a new application. This paper addresses the plug-and-play challenge in the current automotive systems. We propose an Android-based automotive software architecture which supports the play-and play of applications. In our approach, an in-vehicle ECU network consists of one master ECU and several slave ECUs. We adopt Android for the master ECU to enable developers to use well-defined development environment for writing vehicle applications running on the master ECU. The master ECU provides the access to external networks, allowing users to download new applications from cloud computing services.

2. Conceptual Understanding of System

2.1. Block Diagram Explanation

Here we are using the ARM 7 as controller. To this controller the various sensors are connected through different ports of it. Sensors such as temperature, pressure, oil, fuel, gear switches and IR sensor are present here. To work with all this electronics component we require the power supply, i.e of 5v DC. There is crystal connected to microcontroller of particular frequency with reset. For displaying all parameters of vehicle we use the lcd of 16*2 with contrast adjustment. As all parameters which are displayed on the lcdare seen on the Android through the Bluetooth modem with serial interface to the controller.

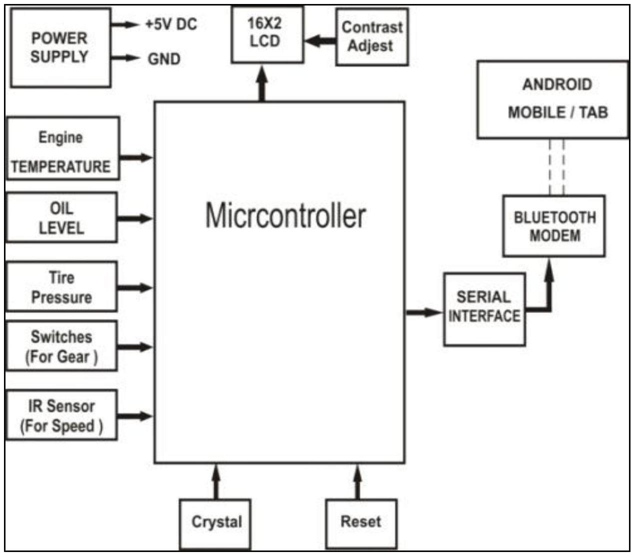

Figure 1: Block diagram

<u>2.1.1. Microcontroller</u>

16-bit/32-bit ARM7TDMI-S microcontroller in a tiny LQFP64 package. It has controls on all the sensors like float sensor, level sensor etc. Bluetooth module interfaced to a microcontroller through serial ports. It has 2 ADC's inbuilt.

<u>2.1.2. LCD</u>

LCD is used to display the data. LCD we have used is 16x2 i.e. 16 characters in 1 line, total 2lines are there. We could have used a better resolution LCD but due to limitation of money and for project requirement 16x2 LCD is sufficient. This LCD has 8-bit parallel interface. It is possible to use all 8 bits plus 3 control signals or 4 bits plus the control signals. It requires +5V to operate. It is connected to port 2 of microcontroller.

<u>2.1.3 Sensors</u>

- Water Float Switch

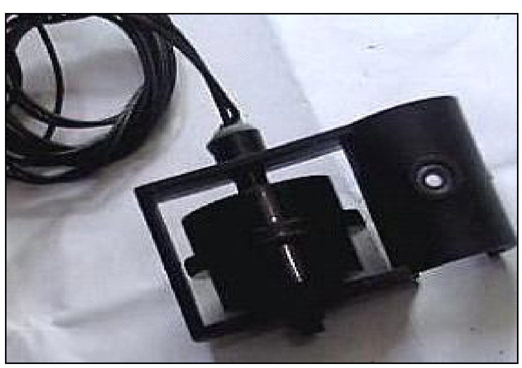

Figure 2: Float sensor

Float is a free movement sensor as shown in fig. When Float is free o/p will be open NO. When the Float is in UP with liquid – o/p will be closed NC. It is easy to fix using Wiring Pipes Available in Electrical Shops. It can be used to detect any liquid level.

- Temperature Sensor

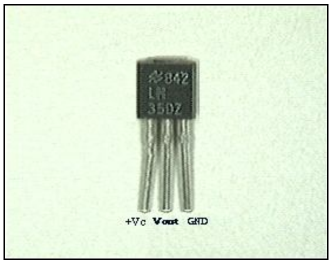

Figure 3: Temperature sensor

The LM35 is an integrated circuit sensor that can be used to measure temperature with an electrical output proportional to the temperature (in ^{o}C). The LM35 generates a higher output voltage than thermocouples and may not require that the output voltage be amplified. You can measure temperature more accurately than a using a thermistor.

- Pressure Sensor

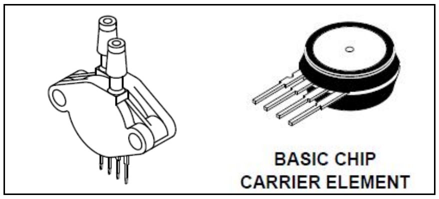

Figure 4: Pressure sensor

It has a patented silicon shear stress, strain gauge design used for air movement control.

2.1.4. Bluetooth

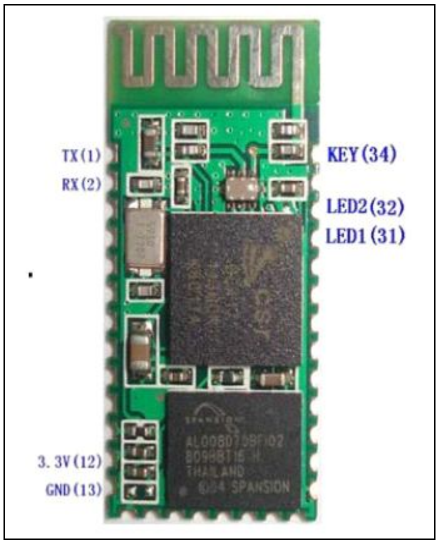

Figure 5: Bluetooth module

Bluetooth is a serial interface module used for wireless communication with android system. It has wireless range upto 10 meters.

2.2. Main Circuit Diagram

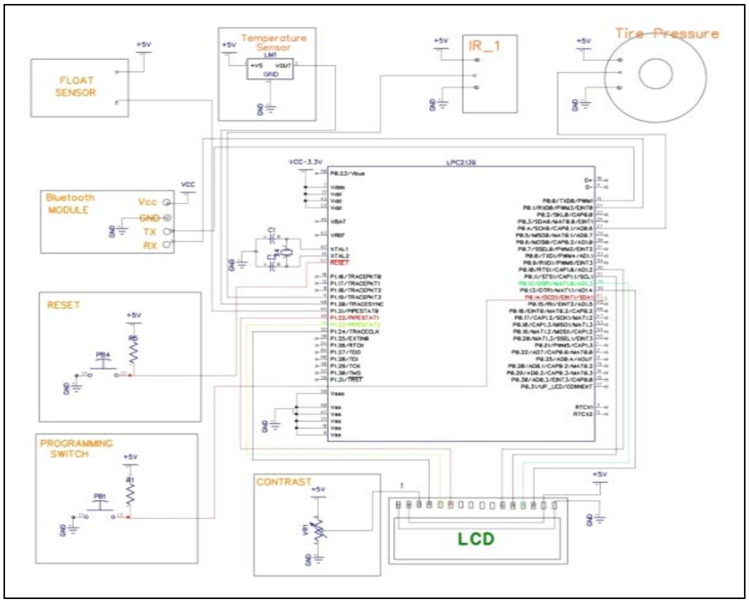

Figure 6: Circuit diagram

Here we are using the Lpc2148 as microcontroller. This controller is connected with different sensors which are placed at different parts of vehicles. As shown in the above circuit diagram the connections are done. Here we make a use of 16*2 LCD display, programming switches for gearing purpose. The microcontroller used has various special features such as low power consumption.

3. Flow Chart

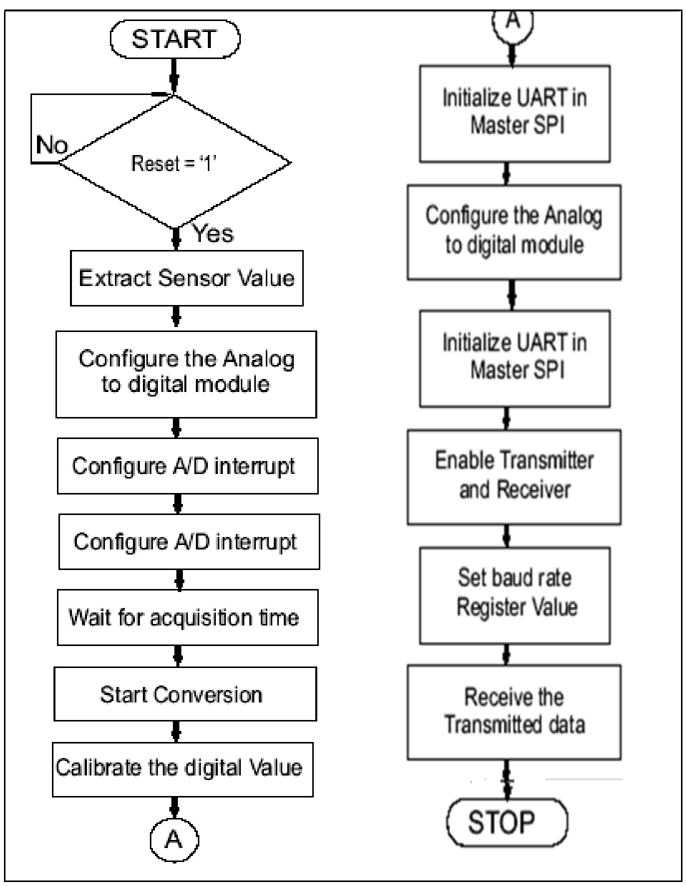

Figure 7: Flowchart

4. Software's

KEIL was founded in 1986 to market add-on products for the development tools provided by many of the silicon vendors. Keil implemented the first C compiler designed from the ground-up specifically for the microcontroller. The Keil Compiler generates code for any device that is compatible with the 8051, 251, C16x/ST10, or ARM microcontrollers. The KeiluVision IDE supports two distinct methods of program testing: simulation and target debugging.

5. Conclusion

As a learning experience, this project was definitely a success. We learn about all aspects of the design process, this system is user friendly, Safe & secure. Definitely this project will provide better compatibility with system for all types of vehicles.

6. References

1. H.-M. Pham, S. Pillement, and D. Demigny, "Reconfigurable ECU communications in Autosar Environment," in International Conference on Intelligent Transport Systems Telecommunications (ITST), Oct.2009, pp. 581–585.
2. G. Macario, M. Torchiano, and M. Violante, "An In-Vehicle Infotainment Software Architecture Based on Google Android," in IEEE
   International Symposium on Industrial Embedded Systems (SIES'09), July 2009, pp. 257–260.
3. Y.-H. Cheng, W.-K.Kuo, and S.-L. Su, "An Android System Design and Implementation for Telematics Services," in IEEE International
   Conference on Intelligent Computing and Intelligent Systems (ICIS), vol. 2, Oct. 2010, pp. 206–210.
4. J. Choi, H. S. Park, Y. Hwang, and K.-H. Kim, "Exhibition Speaker:Driver-Oriented Intelligent Human-Vehicle Interaction System," in
   International Conference on Intelligent Systems, Modelling and Simulation. IEEE Computer Society, 2012, pp. 14–16.
5. R. Anthony, A. Rettberg, D. Chen, I. Jahnich, G. de Boer, and C.Ekelin,"Towards a Dynamically Reconfigurable Automotive Control System Architecture," in mbedded System Design: Topics, Techniques and Trends, ser. IFIP Advances in Information and Communication Technology. Springer Boston, 2007, vol. 231, pp. 71–84
6. H. Seebach, F. Nafz, J. Holtmann, J. Meyer, M. Tichy, W. Reif, and W.Sch?er, "Designing self-healing in automotive systems," in Autonomic and Trusted Computing, ser. Lecture Notes in Computer Science.Springer Berlin/Heidelberg, 2010, vol. 6407, pp. 47–61.
7. B. Becker, H. Giese, S. Neumann, M. Schenck, and A. Treffer, "Model-Based Extension of AUTOSAR for Architectural Online Reconfiguration,"in Proceedings of the 2009 International Conferenceon Models in Software Engineering, ser. MODELS'09. Berlin,Heidelberg: Springer-Verlag, 2010, pp. 83–97.
8. RT-Druid, "RT-Druid: A tool for architecture-level design of embeddedsystems" White Paper, Evidence S.r.l. 2004.
9. J. Lehoczky, L. Sha, and Y. Ding, "The Rate Monotonic Scheduling Algorithm: Exact Characterization and Average Case Behavior," in Real Time Systems Symposium, Dec. 1989, pp. 166–171.
10. Y. Shimizu and T. Kawai, "Development of electric power steering," in SAE Transactions, no. 910014.
11. S. ichiro Sakai, H. Sado, and Y. Hori, "Motion control in an electric vehicle with four independently driven in-wheel motors," in IEEE/ASME Transactions on Mechatronics, vol. 4, Mar. 1999, pp. 916.