Journal of Industrial Technology 15 (1), 2006, 73-85

ELECTRONIC CONTROL UNIT FOR VEHICLES: ISSUES AND THE WAY FORWARD

Ibrahim Mat, Ishak Aris, Mohd Khair Hassan, Roslina Mohd Sidek,
Nik Anis Alina Nik Othman and Syed Malek Faisal

Department of Electrical and Electronic Engineering
Faculty of Engineering, Universiti Putra Malaysia
43400, Serdang Selangor, Malaysia
(ishak@eng.upm.edu.my)

RINGKASAN : *Kenderaan masa kini menampakkan perkembangan yang pesat terutamanya pada sistem elektronik. Sistem elektronik berperanan mengawal sebahagian besar daripada fungsi-fungsi yang terdapat pada sesuatu kenderaan. Revolusi perlaksanaan pelbagai fungsi dan ciri-ciri pada sistem automotif moden ini berlaku kesan dari penggunaan pesat teknologi elektronik. Revolusi elektronik automotif telah menampakkan peningkatan mutu kenderaan dari segi prestasi, kebolehtahanan dan juga keselesaan. Pada masa yang sama perkembangan teknologi elektronik juga memudahkan lagi proses fabrikasi komponen elektronik serta proses pemasangan kenderaan. Unit Pengawal Elektronik (ECU) ialah sistem yang memainkan peranan yang sangat penting dalam kemajuan teknologi automotif. Keperluan kenderaan moden masa kini perlu mematuhi spesifikasi yang lebih mencabar atau dengan kata lain untuk meningkatkan mutu kenderaan dengan kos penghasilan kenderaan yang lebih baik. Ini memerlukan pembaikan dalam semua aspek terutamanya spesifikasi dan proses rekabentuk. Kertas kerja ini membincangkan secara menyeluruh tentang ECU termasuklah perkembangan serta cabaran terkini yang dihadapinya. Isu seperti peraturan emisi, algoritma dan perhubungan juga dibincangkan.*

ABSTRACT : The latest generations of road vehicles have seen a tremendous development in onboard electronic system, which controls increasingly large parts of the vehicle's functionality. The automotive electronic revolution has significantly improved the performance, reliability and comfort of the automobile. Electronic Control Unit (ECU) is a system that plays a more and more important role in the development of road vehicles. The increasing complexity and stringent quality and cost requirements mandate tremendous improvements in all aspects of specification and design process. This paper discusses the overall aspects of the new era of ECU, issues in emission regulations, algorithms and communications are addressed.

KEYWORDS : ECU, ECM, EFI, automotive, engine control

INTRODUCTION

An Electronic Control Unit (ECU) is the Electronic Control Module (ECM) responsible for engine control functions. The primary purpose of an ECU is to provide closed-loop control of the fuel and ignition systems in an engine to improve fuel economy and reduce airborne pollutants produced by the engine. Today, ECU functions are not limited to engine control but they cover all aspects of cars such as engine performance, safety, comfort, air pollution, infotainment and several names of end-user requirements.

Importance of ECU

The number of ECU in a car varies depending to the complexity and the price of the car. The minimum requirement for a car is one unit of ECU that is for the control of the engine. Therefore the ECU discussed in this paper will focus more towards the control of the engine.

Among the primary purposes of ECU are to improve fuel economy and reduce airborne pollutants. ECU fulfills its purpose by accurately controlling the fuel mixture (air-to-fuel ratio), and spark timing (spark advance and duration) based on feedback from sensors connected to the engine. Management of the fuel mixture and ignition timing is complicated. The ECU requires data from many sensors to provide optimal control of the system (Cuatto, 1998). The ECU needs to know ground speed, engine speed, crankshaft position, air quantity (oxygen content), engine temperature, engine load (for example, whether the air conditioner is on), throttle position, rate of change of the throttle, transmission gear, emitted gases from the exhaust, and the list goes on. The ECU, as stated earlier, is a controller, designed to manage a specific problem.

History of ECU

Years ago (Bortolazzi *et al.*, 1996), car engines were controlled by carburetors. These mechanical devices controlled how the engine performed activities such as ignition, and fuel injection. These devices were complicated and were getting harder to maintain as time went by. With the introduction of catalytic converters, as well as stricter fuel emissions laws, carburetors were having a tough time keeping up. Therefore, there had to be some other way to do engine control. In the beginning, carmakers developed electrically controlled carburetors that could adjust air-to-fuel ratio in real time, but these systems were way too complicated. Eventually, Engine Control Units that could get the job done were introduced (Bortolazzi *et al.*, 1996).

The first ECU came with 8-bit processors and was coded strictly in assembly language. In the recent years, ECUs have been moving towards 16 and 32 bit processors and the software trend is towards C language. As time passes, with the technological developments, ECUs have been becoming more reliable and efficient, but costs to produce them have been increasing (Bannatyne, 2002; National Instruments Corporation, 2003).

ECU also becomes a controller to other functions such as for Antilock Braking System (ABS), power window controller and suspensions (Bannatyne, 2002; National Instruments Corporation, 2003).

Operation of ECU

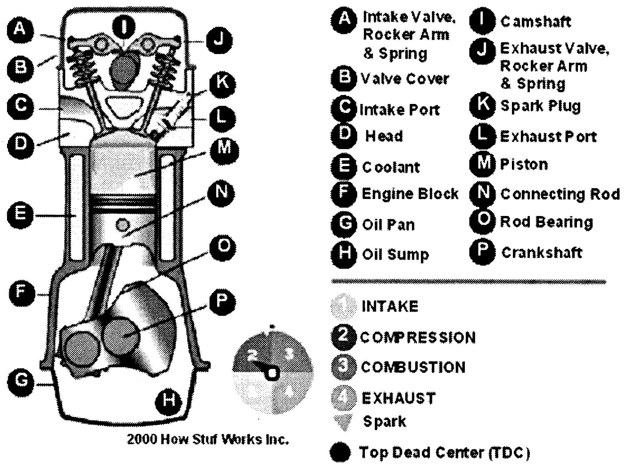


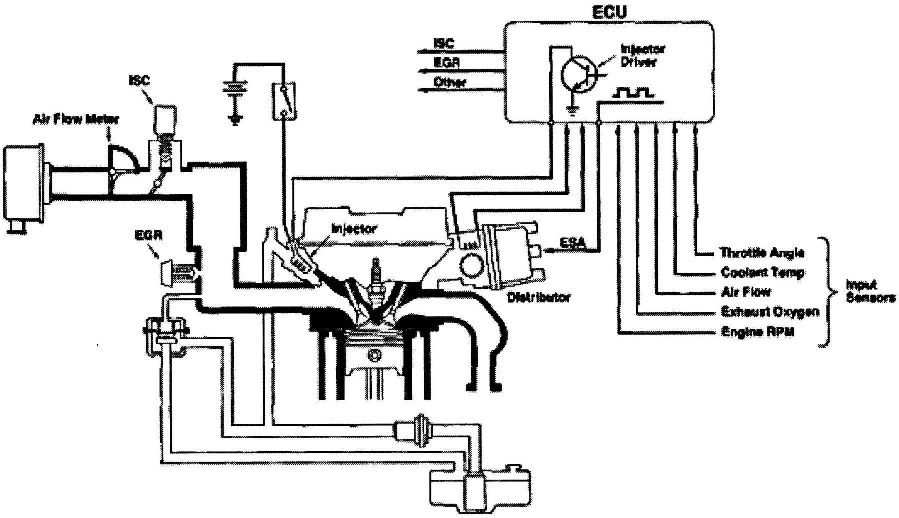
***Figure 1.** Single Cylinder Engine describing the components
(http://www.howstaffworks.com/, 2000)*

The ECU is an extremely reliable piece of hardware which has the capability to receive and process information hundreds of times per second. At the heart of the ECU is the microprocessor. It is the processing center of the ECU where input information is interpreted and output commands are issued.

An Electronic Fuel Injection (EFI) engine as shown in Figure 2 is taken as an example to illustrate how ECU works in controlling the engine. The following is the sequence of how ECU works (http://www.autoshop101.com/autoshop15.html, 2004):

- Air enters the engine through the air induction system where it is measured by the air flow meter. As the air flows into the cylinder, fuel is mixed into the air by the fuel injector.
- Fuel injectors are arranged in the intake manifold behind each intake valve. The injectors are electrical solenoids which are operated by the ECU.
- The ECU pulses the injector by switching the injector ground circuit on and off.

- When the injector is turned on, it opens, spraying atomized fuel at the back side of the intake valve.
- As fuel is sprayed into the intake airstreams, it mixes with the incoming air and vaporizes due to the low pressures in the intake manifold. The ECU signals the injector to deliver just enough fuel to achieve an ideal air/fuel ratio of 14.7:1, often referred to as stoichiometry.
- The precise amount of fuel delivered to the engine is a function of ECU control.
- The ECU determines the basic injection quantity based upon measured intake air volume and engine rpm.
- Depending on engine operating conditions, injection quantity will vary. The ECU monitors variables such as coolant temperature, engine speed, throttle angle, and exhaust oxygen content and makes injection corrections which determine final injection quantity.



Note:
ISC : Idle Speed Control
EGR : Exhaust Gas Recirculation
ESA : Ignition Spark Management

Figure 2. *ECU with Sensors and Engine components*
(http://www.autoshop101.com/autoshop15.html, 2004)

This paper briefly describes the initiatives done and the experiments prepared for ECU development. A short review on the primary function and the history of ECU will also be presented. Next, issues and challenges faced in various aspects will be presented. Some market trend of ECU, which reflects the importance and its requirements, will also be discussed.

ISSUES AND CHALLENGES

The purpose of ECU is to achieve an optimum performance with less emission. There are several issues being faced by designers or developers as well as manufacturers such as fuel injection, control algorithms, methodology, approaches, hardware and software.

Issues in Emission Regulations

The car system has to satisfy emissions requirements for at least 120,000 kilometers, as well as meeting Environmental Protection Agency (EPA) of U.S.A. requirements (Toyota, Owners Manual Supplement T100/Tacoma, 1998) protect engines against abuse, and dozens of other smaller requirements. A closed-loop control method is used as a control scheme to monitor inputs and outputs of the system. From this feedback, ECU manages emissions and fuel economy of the engine. ECU will gather data from dozens of different sensors to perform millions of calculations each second, including looking up values in lookup tables (Intrepid Control System Inc., 2004).

Issues in Algorithms

The increasing complexity of automotive electronic systems has had a dramatic effect on the throughput requirements and peripheral integration of automotive microcontrollers. Algorithms are now required to handle the inputs from many sensors and communication systems, execute real-time control cycles and control the outputs of many actuators. The definition of a real-time control is a control system that responds to its inputs within a fixed time. This term is often used for systems that respond to inputs at very high speed. Most of the embedded systems in automobiles are based on real-time control systems (Cuatto *et al.*, 1998).

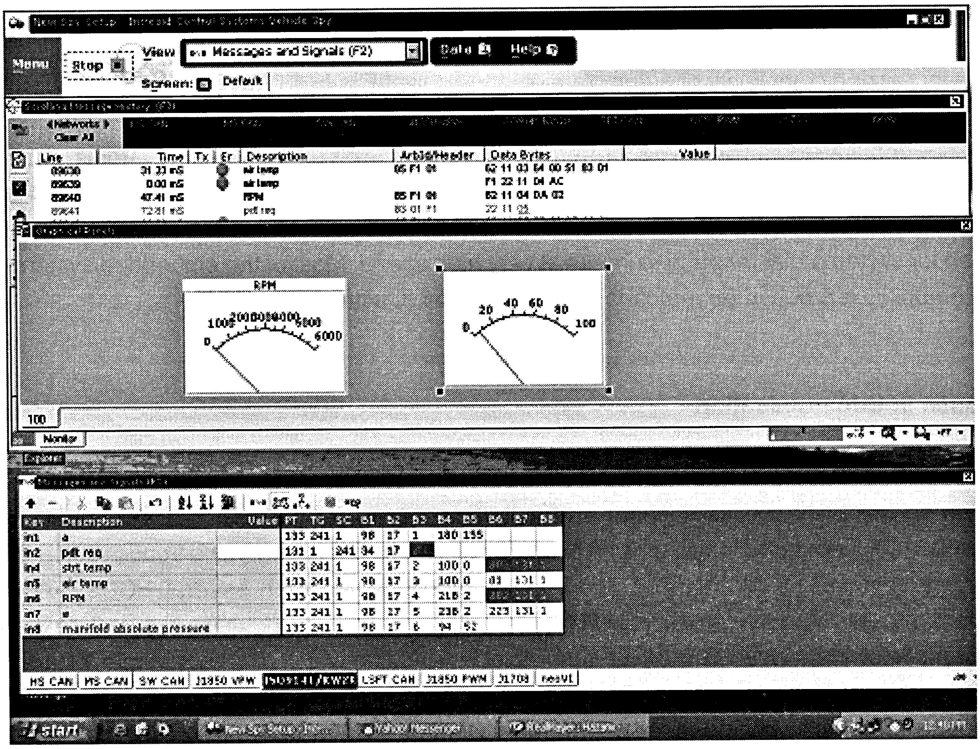
Issues in Communication

Most of the ECUs installed in cars today are still using point-to-point modes of communication between the ECU and its sensors and actuators. There are also various communication standards, for example Controller Area Network (CAN) which is an emerging trend in automobile communications. It can also be used to communicate between several ECUs. Another communication standard is Keyword 2000 protocol for diagnostic purposes. Figure 3 illustrates a sample of diagnostic Graphical User Interface (GUI) which shows sample of engine parameters such as Revolutions Per-Minute (RPM) and temperature. It also can be used for troubleshooting the engine functions (Intrepid Control System Inc., 2004). Keyword 2000 is based on serial Universal Asynchronous Receiver Transmitter (UART) communication with various limitations. The trend now is to perform the diagnostic through CAN Bus.

The CAN protocol allows communication speed up to 500 kilobits-per-second, which increases the reliability and speed of point-to-point. This is also an emerging trend in engine control. With this protocol all components can send fault signals to a central module. The central module stores the fault signals and then sends them to an off-board diagnostic tool (Bortolazzi *et al.*, 1996).

It is better to pass information between the ECU, sensors, actuators and others via a shared multiplexed bus rather than the numerous point-to-point connections. This will minimise the wiring. Ultimately, it will increase reliability while reducing cost, but moving to multiplexed buses introduces new problems. Amongst the problems is to ensure that the various signals, which share the communications medium, meet their various deadlines.

An efficient usage of the network bandwidth also calls for several signals to be carried in the same transmission unit on a network. In order to build a good communication system, there is a need to use techniques that allow the calculation of the worst-case time. This is to ensure that the network is efficient and everything is meeting its deadlines. Figure 4 shows a typical block diagram for various CAN node locations in a car.



Key	Description	Value	PT	TG	SC	B1	B2	B3	B4	B5	B6	B7	B8
in1	a		133	241	1	98	27	1	180	155			
in2	pdt req		131	1	241	34	27	[illegible]					
in4	strt temp		133	241	1	98	17	2	100	0	[illegible]	[illegible]	[illegible]
in5	air temp		133	241	1	90	17	3	100	0	81	131	1
in6	RPM		133	241	1	98	17	4	218	2	[illegible]	[illegible]	[illegible]
in7	e		133	241	1	98	27	5	238	2	223	131	1
in8	manifold absolute pressure		133	241	1	98	27	6	94	52			

Figure 3. *Diagnostic Graphical User Interface*

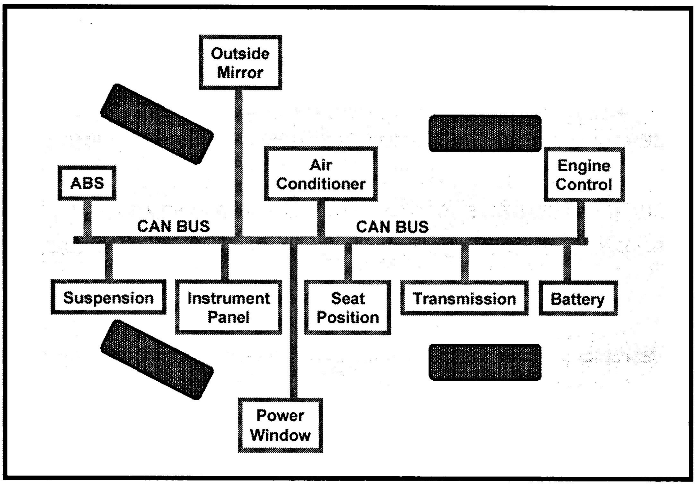


Figure 4. *CAN Bus connecting between ECU Nodes in a car (Intrepid Control System Inc., 2004)*

Issues in Engine Control Tasks Schedule

In early engine management systems, software architecture operated as "foreground/ background". Foreground mode is based on interrupt or event driven, whereas background mode runs all other calculations continuously. New systems implement multiple processes. Each process carries its own priority. The new system remains in use due to new techniques in scheduling the engine management tasks introduced by Robert Bosch (1998). This was not a new idea, but the paper showed how the notion of torque planning had been integrated into the software architecture. The physical architecture is the development of engine torque. Torque is produced when combustion takes place in the engine cylinders and will be modified as in-cylinder conditions are changed using the following controls: Spark timing, injection timing, cylinder cut-off and charge mass through throttle.

The most critical scheduling problem in an ECU is to miss task deadlines at higher engine speeds. The root cause of this is because engine manufacturers often use low-end processors for their application to reduce production costs. The low-end CPU has low clock speed with a limited ROM for program storage. Car manufactures now have techniques that can help overcome this problem. These solutions are applied during the software development phase and they mainly improve the software in order to balance out the need for more hardware even though this increases the work effort as a whole. The techniques are libraries optimization and processor specific features.

Another technique proposed by General Motor (GM) was dynamic scheduling. Dynamic scheduling reduces execution requirements at higher engine speeds. This technique employs a separate engine speed zones. In each zone, there is a different function level, and this simplifies the coordination of scheduling change. In the middle engine speed range, the functions are divided across multiple engine events. This will balance the load across multiple cylinder events. At higher engine speeds, it significantly simplifies some functions. This can be accomplished because engine states do not change in every cylinder.

However, the demand for more processes to be handled keeps increasing. Higher performance processor has taken place recently such as MPC555 or MPC565 (Paccard, 1999).

FUTURE ECU TRENDS

Future development trend

The current trend shows that electronics approach has replaced the traditional mechanical solutions. This includes safety systems (Chujo, 2002), such as stability control, fundamental vehicle functionality such as brakes and steering. Electronics can provide increased functionality and at the same time reduce cost and weight. However, new electronic systems, such as fault-tolerant buses, are likely to be needed to reach sufficient safety levels for these crucial functions. To reduce service times and unscheduled stops, improved vehicle diagnostics is also necessary.

Another area where large changes are anticipated is in information to the driver and passengers, and also in entertainment functionality. This will affect the technology used in the electronic systems since current networks have insufficient bandwidth. Most likely, fiber optical solutions will be required to reach speeds in the megabit range.

Future Trend for System Integrators and Suppliers

In the automotive industry, there is a long tradition to work with suppliers, and a number of suppliers exist that are comparable in size to the vehicle manufacturers themselves and deliver similar systems to all of them, thereby reducing the cost. The role of the vehicle manufacturer is then to specify the subsystems, and integrate them in the vehicle (Axelsson, 2001).

The suppliers have mostly developed isolated subsystems, e.g. consisting of an ECU, its software, and possibly mechanical parts, too. However, from other thinking they predict that in the future it is expected that the number of ECU will not grow drastically, simply because it becomes too expensive and heavy to have separate hardware for each functionality, and the available space in the vehicle is also limited. Therefore, ways must be found to integrate software modules that participate in the implementation of different functionality in the same ECU. The software modules may be delivered by different suppliers, and the integration must ensure the integrity and performance of all the functions despite the sharing of hardware resources (Axelsson, 2001). This is a particular challenge when it comes to safety-critical functionality, and the requirements on the software modules are thus not limited to their functional interface, but touch their overall behaviour.

Rapid Prototyping - Experiment Platform

In order to accelerate the time-to-market as well as guaranteeing passenger safety, the rapid prototyping technique is used in ECU design. It is used for modeling, simulation and verification in early development stage. This will give an early validation concept, a better understanding and maintainability of the specification, and thus a considerable reduction of the overall development costs. One typical example of rapid prototyping model is shown in Figure 5.

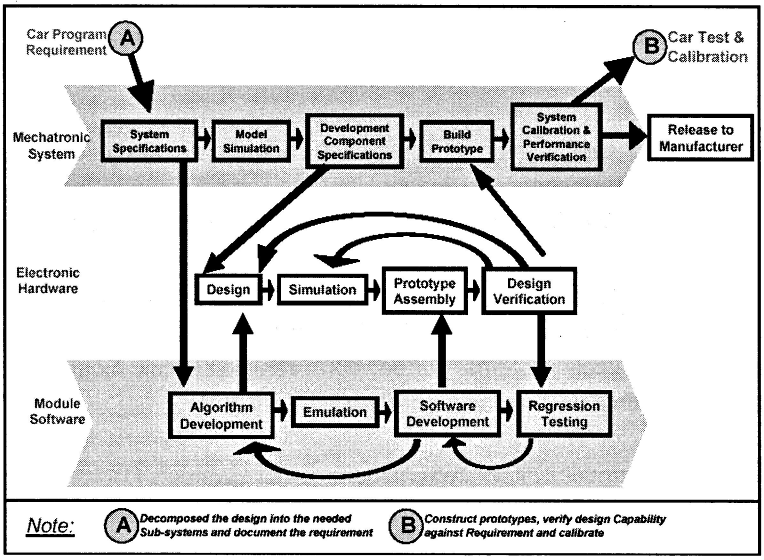


***Figure 5**. Rapid prototyping development model
(Bortolazzi et al., 1996)*

Future Trend in Software for ECU

Today's automotive software systems are mainly developed independently of each other, which explain the low information exchange between control units. Their main tasks are data acquisition and system regulation. The software represents almost half of the overall R&D system cost and this share will increase in the future, along with software complexity, while the hardware price decreases (Bell, 2002). Figure 6 shows the prediction of ECU software in future.

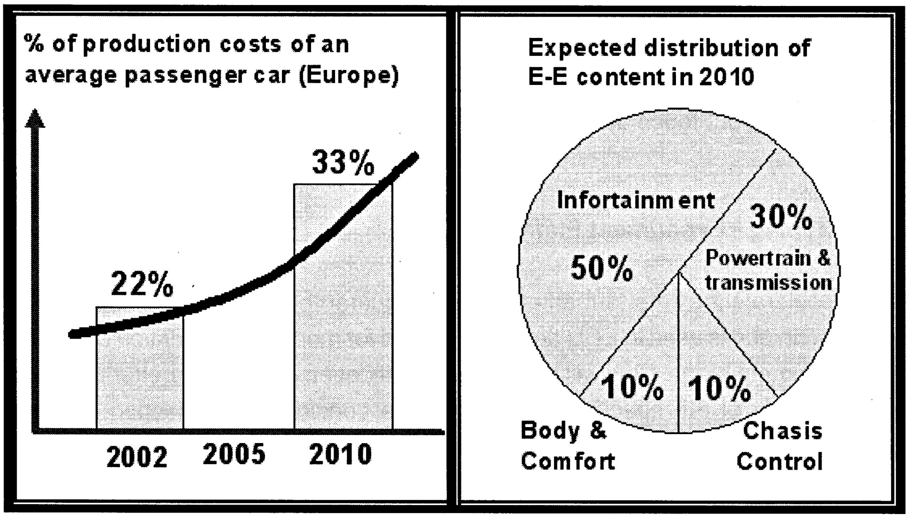


Figure 6*. Forecast on Electronic and Software in automotive.*
(Bell, 2002)

In the latest development, most of high-end cars have complex embedded electronic systems with embedded network interconnecting dozens of ECUs and communicating with the outside world via GPS-based and wireless technologies, thus increasing software complexity.

A major effort concentrates on adapting the methods and tools to vehicle specific aspects. This includes a model scheme representing the functional and architectural aspects specific to distributed electronic vehicle systems. Furthermore, vehicle specific model libraries as well as interfaces for integrating existing functionality are developed.

The next generation will introduce integrated car management systems followed by autonomous driving and brake-by-wire or steer-by-wire systems. A standard real-time operating system will also be part of future ECUs (Frank *et al.*, 2004).

A major improvement will be based on client/server architectures, which allows an optimized usage of the resources available in the car as illustrated in Figure 7. This approach enables the flexible implementation of a specified functionality on alternative constellations of processing power in the car. It is important to notice that safety critical applications like ABS and closed loop control applications mandate specific strategies in this scenario.

Microcontroller Trend in ECU Design

Due to advanced system requirements, microcontroller technology requirements will be impacted. Microcontroller with high throughput and fast interrupt handling capabilities is

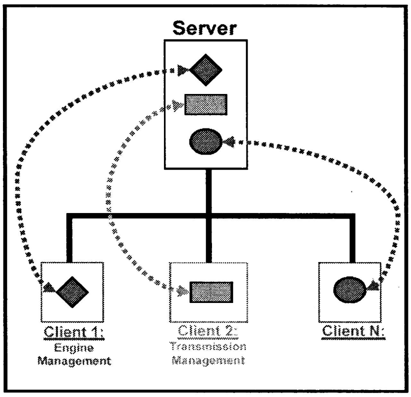


Figure 7*. Client/Server Architecture for future Electronic Control Unit (Bortolazzi et al., 1996)*

required. The CPU must also be conducive to generate an efficient code with C compiler as a high-level language.

Traditionally, what has set RISC apart from CISC is the ability to execute an instruction in a single clock cycle and the fact that RISC machines do not use microcode to decode instructions, but are hardwired. It has been argued that the vast majority of most software consists of very simple instructions. The philosophy of RISC is to produce processors that can execute these simple instructions (such as 'add', 'subtract', 'shift', etc.) in one clock cycle. More complex instructions such as 'multiply' and 'divide' were not available on early RISC processors. A typical microcontroller growth in automotive industry is shown in Figure 8.

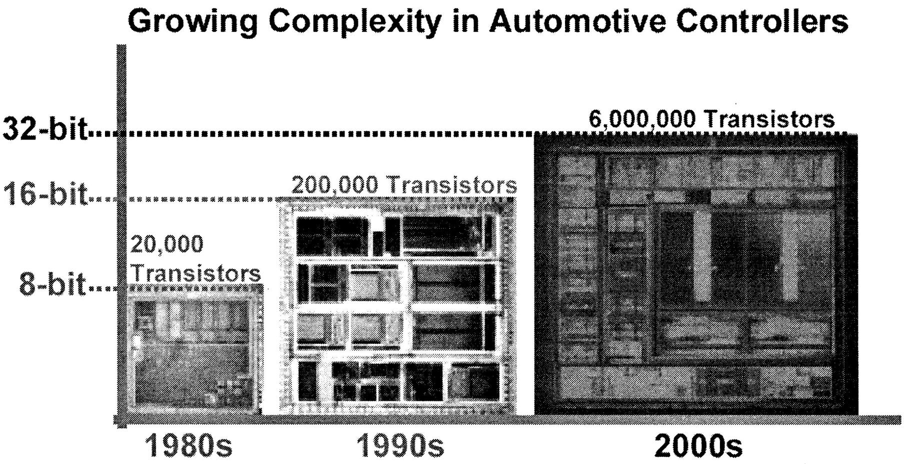


Figure 8*. Growing Complexity Level of Microcontrollers in Automotive Industry (Bannatyne, 2002)*

CONCLUSION

This paper has given an overview of the current activities and trends as well as issues in the area of ECU design.

The dynamic evolution of ECU systems with respect to growing functionality and an increased number of safety critical functions mandates new methods and tools for the development as well as appropriate system architectures for prototyping and productions systems. Future applications heavily demand the following improvements:

- Library based reuse of functionality and software
- Open interfaces for model exchange and data access for specific optimisation and processing
- Production code generation and interaction with target system instrumentation and debugging
- More intelligent sensors to reduce processing time by the CPU, such as MEMS technology
- Incorporation with intelligent features in software
- Rapid prototyping provides great potentials to fulfill the goals to reduce development costs, increase quality, and reduce development time
- Strong need for automatic code generators, to get rid of the manual code generation

Issues and challenges presented will create more opportunities for the improvement and advancement of the ECU. Successful implementation requires utilizing the appropriate strategy to address all issues with respect to environmental requirements, functional partitioning, and control system development relationships. No single strategy will universally apply and each situation must be evaluated carefully to understand which approach is most appropriate.

ACKNOWLEDGEMENT

Great thanks to The Ministry of Science and Innovation, Malaysia for providing research grant for this project.

REFERENCES

Axelsson,J. (2001), HW/SW Co-design for Automotive Applications: Challenges on the Architecture Level, IEEE Proceedings of the 4th International Symposium on Object-Oriented Real-Time Distributed Computing. pp. 2, 3.

Bannatyne,R. (2002). Trends in Automotive Electronic Control Systems. 5th Edition of Business Briefing: Global Automotive Manufacturing & Technology. pp 1-4.

Bell,P. (2002). Modern Software Practice - the need for a new approach in automotive. Total Vehicle Technology Conference 2002. pp. 2, 3.

Bortolazzi,J., Hirth,T., Raith,T. (1996). Specification and Design of Electronic Control Units. EURO-DAC '96 with EURO-VHDL '96. pp. 1, 2.

Chujo,N. (2002). Fail-safe ECU System Using Dynamic Reconfiguration of FPGA. R&D Review of Toyota CRDL **37** pp. 1, 2.

Cuatto,T., Passerone,C., Lavagno,L., Jurecska,A., Damiano,A., Sansoe,C., Vincentelli,A.S. (1998). A Case Study in Embedded System Design: an Engine Control Unit. DAC'98. pp. 1-3.

De Vos,G.W., Helton,D.E. (1999). Migration of Powertrain Electronics to On-Engine and On-Transmission. International Congress and Exposition 1999. pp. 1, 2.

Frank,J., Winters,F.J., Mielenz,C., Hellestrand,G. (2004). Design Process Changes Enabling Rapid Development. Convergence Transportation Electronics Association. **21**. pp. 2, 3.

Intrepid Control System Inc. (2004). Controller Area Network (CAN), pp.8-12.

National Instruments Corporation (2003). Building Flexible, Cost-Effective ECU Test Systems. pp 3, 4.

Paccard,E. (1999). Technology for a New Automotive Era. Real-Time Magazine **99-3**. pp.1,2.

Toyota (1998). Owners Manual Supplement T100/Tacoma. pp. 11.

http://www.autoshop101.com/autoshop15.html, (2004)

http://www.howstuffworks.com/, (2004)