



Review on Embedded System

Harpreet Kaur¹, Jagatjit Singh²

Abstract: The purpose of this paper gives brief about the embedded system and expounds the main contents of embedded systems are real time operation system, application of embedded system, co design and design challenges etc. Embedded system is a part of large system. It is just try to know about the design of embedded system, real time operation system and also familiar with co design of hardware and software.

Keywords: Embedded system, Co-design, Design Challenges, The need and priorities in Research.

I. INTRODUCTION

Embedded system is a computer system which is the part of larger system and performs the particular task. In many countries, embedded industries have been a booming session in back few years. It have large fraction of a marking of digital system by the fact of embedded system represents a key technology in consumer electronics, automotive, military and aerospace, telecommunication ,data communication and office automation industries. All uses 32 bit microprocessor across the worldwide in embedded system. This processor provides the functionality for special purpose as comparative to applications used for general purpose like desktop user. The embedded processor have growth rate far exceeds than traditional computers.

Embedded System also characterized as real time systems, it means that they have real time response like time analysis, worse case execution time etc. are very important point for design. Embedded systems must meet the some important specifications safety, availability, reliability and also dependability. Because of small size and requirement of mobility, also extreme low production cost these systems require small and controlled resource consumption and also have limited capacity of hardware. With the increased complexity of real time embedded system tends to increase the demand with requirements of engineering, early error detection, high level design, integration, productivity, verification and maintenance which increase the importance of life cycle properties like maintainability, portability.

A. Basic concept for component based embedded system

In engineering trade component is a self trade part or subsystem can be used as a building block in larger system. The best understanding of component in the software industry of world is based on Szperski's definition. According to definition executable unit of component, that composition and deployment can be performed at run time. In the large group of embedded system followed this definition. But demands on executable or binary from are not followed directly. A component could be delivered in the form of source code written in HDL or Verilog of high level languages and allowed the composition of design time or build time.

The summarization of interface properties of component visible externally to the other part of system. Extra functional properties of embedded system are very

important like function and have a tendency to include the extra functional specifications for the component interfacing. During design of system it allows the other properties of system to be determined.

The technologies of general purpose component, implemented the interface like polymorphism of object interfacing by the late binding. In the late binding the component connections that are unaware of each other by the connecting interface, this flexibility of system increased the failure of system. It also affects on the system performance or other properties decreases when components composition occurs at run time.

The real time and limited resources have several reasons for the design time of composition and deployment of component performance instead of run time. The component composition enables the global optimization, component connections can be changed into direct call function rather than using all the notification, prediction and verification requirement's of system from the given properties of component. Resources constraints of large embedded systems are not primary concerns. The interpretability and complexity plays a very important role. Due to the complexity of development systems have very much cost and high priority in cutting development cost. Because of this reason, the component technologies for general purpose are more interesting than case of small system. The technologies used for the system belongs to the category of soft real time system.

The additional abstraction level support uses component technology like base and specified as propriety solution or set of standard the example of component based on technology is OLE Process control Foundation, responsible for the specification set of standards interfaces with auto mation depend on OLE/COM.

B. Real time Operating System

Real time operating system (RTOS) is a multitasking embedded system build for real time applications. Operating system can be considered as real time if it enables the program to particular task performed in given time. The RTOS reduces or drop certain functions when they could not able to execute the task in the given time constraint RTOS including the contents of basic concept, algorithm, task, kernel, transparent and interrupt of RTOS etc. In many case design system must take into worst performance. The predication of worst case cause



complicated architecture. The Mission Critical and Signal Processing are the significant requirement of real time operation to meet the specification of control stability and external interfacing. Most of the embedded system uses reactive components.

The main challenge of RTOS is worst case design analysis without undue pessimism in the face of software with statistical characteristics of performance (e.g. cache memory).

II. CO DESIGN

Co design is based on the system specification, architectural design, hardware and software partition. The software and hardware integration completed this task. Software and hardware co design make it possible to make the Many embedded system could not tolerate the cost of handy devices carried out individually. The purpose of co design is to combine memory of CPU and controlled program Example of Hardware co design: The applications of software and hardware co design ranging from everyday life to machines for special purpose. The common examples are automobiles, GPS, television, network routers, and thermostats. Let us discuss the example of MP3 player. The player has a large memory which is capable of storing large number of songs. The audio present in the form of audio and raw audio is generated in the form of digital signal. Then information is displayed on the screen with the help software working in the memory. Implementation of MP3 audio decoder is with help of co design software and hardware which is helpful in specification of real time MP3 player. In co design, problems have different flavour according to the domain application, design methodology, implementation technology. Hardware circuits are described using programming languages or the modelling techniques; they are validated and implemented with help of software program, sometimes demand for specific hardware design. The designer of system required knowledge about hardware as well as software domains to make efficient system design tradeoffs.

III. REQUIREMENT AND DESIGN CHALLENGES

1. *Small size, low weight*

In the larger aircraft many embedded computer are located in the system. In portable and transportations system, weight is critical concern for fuel economy or human endurance. As a example, Mission Critical system require much more weight. and stringent size than the others because of it uses in flight vehicle, all the examples have this type of restrictions

Design Challenges:

- For every design, de rating components depending on operating environment.

2. *Safety and reliable*

Some systems have a risk of failure. Traditionally some systems have distributed census protocol or multiply redundant comports in order to ensure continued operation.

redundancy in processing for the technique of fault tolerance.

Design Challenges:

- Minimal redundancy for low cost.

3. *Harsh Environment*

In controlled environment, some embedded system does not operate. The problem is excessive heat, especially in those applications having combustion (e.g. transportation). The other problems can be caused for embedded computing by the need for protection from shock, vibration, power supply fluctuations, and general physical abuse.

Design Challenges:

- Accurately thermal modelling

4. *Cost Sensitivity*

Cost is always very important issue in embedded computers. A reason may be effect of computer cost on profitability is the function of proportion of cost changes compared to the total cost as comparative to digital cost.

Design Challenges:

- To permit the trade off between optimization of aggressive cost and product robustness.

5. *System Level Requirements*

In the competitive timing of market, embedded system required that designer take the account of entire system, when making the design.

- I. End product utility: The end product utility is the goal during the designing of embedded system, not capable of embedded computer itself. On the way of looking embedded system is the mechanism and associated their input/output are defined by the application. The software is used to coordination between the mechanism and their functionality.

Design Challenges:

- Driven hardware and software synthesis.
- II. System safety and Reliability: A bigger and difficult task is software reliability and safety. Normally, software does not break in sense of hardware but it may be complex software due to uncertain circumstances and cause of failure of software.

Design Challenges:

- Reliability of software
- Cheap and available system

IV. NEED AND PRIORITY IN RESEARCH

Fully utilized advantages of component based embedded system, need for a large number of issues that could be improved or solved. During these days, lack of widely adopted technology standards which are suited to embedded system. For smaller size of embedded system, its important that system composed of components can be optimized for memory consumption and performance, such a support is still missing in toady available component technologies. Some component technology



does not supported analysis and specification of extra functional properties.

To support advanced feature in embedded system based on component technology, the run time must provide the service, which uses limited resources. The number of other issues like component certification, platform and vendor independence, component interference, which solved for adopted for component technologies.

V. REFERENCES

- [1] Ahmed A Jerraya, Long Term Trends For Embedded Design, TIMA Laboratory, 46 av Felix Viallet 38301, Grenoble France, Proceeding of the EUROMICRO System on Digital Design System(DSD'04).
- [2] Ivica Cmvkovic, Component- based Engineering For Embedded Systems, Malardalen University, Department of Computer and Electronics Sweden, ICSE'05.
- [3] Parima Patel Embedded Systems Design Using FPGA, Proceeding of 19th Conference On VLSI Design (VLSID'06).
- [4] Philip Koopman Embedded System Design Issues, Engineering Design Research Center, Carnegie Mellon University, PA 1513.
- [5] Prof. Kirti Ramamritham, Prof. Kavi Arya , System Support For Embedded Applications, Indian Inst. Technology, Bombay, Proceeding of the 16th International Conference on VLSI Design (VLSI'03).
- [6] Wang Ping, Research on Embedded System Teaching, Electric and Information Engineering College, Chengdu, China, International Workshop on Education Technology and Training & International Workshop on Geoscience and Remote Sensing, 2008.