# REVITALIZATION OF PROCESSOR FOR CYBER PHYSICAL SYSTEMS USING RTOS

**V. Shanmuga Priya[1], T. Selvasankari[2]**

[1]Student, M.E Embedded systems and technologyt, K.C.E.T, Tamilnadu, India
[2]Assistant Professor, Department of E.C.E, K.C.E.T, Tamilnadu, India

## Abstract

Cyber-physical systems (CPS) are physical and engineered systems whose operations are monitored, coordinated, controlled and integrated by a computing and communication core. Just as the internet transformed how humans interact with one another, cyber-physical systems will transform how we interact with the physical world around us. These systems will have a constant sensor sampling rate and processor response time. Hence when they are deployed to processors that are shared amongst control and non-control tasks it will potentially increases CPU usage, such as cache, interrupts and task management through schedulers and results in interference between tasks. This reduces system stability. . Hence Experiments are done in control and computing communities with the scope of maximizing CPU utilization and maintaining system stability. In this paper we analyze how a Processor maintains its stability when maximum resource is utilized. It can perform various operations such as border security, Rescue operation, Firing, Obstacle detection, Vibration detection and they communicate with the Tankers through RF and also uses Zigbee technology to communicate with the server.

Keywords: —CPS, Rtos, Wireless robot, salvo, revitalization, CPU resources

--------------------------------------------------------------------***--------------------------------------------------------------------

## 1. INTRODUCTION

Cyber-physical systems are expected to play a major role in the design and development of future engineering systems with new capabilities that far exceed today's levels of autonomy, functionality, usability, reliability, and cyber security[2]. Embedded systems have always been held to a higher reliability and predictability standard than general-purpose computing. Consumers do not expect their TV to crash and reboot. They have come to count on highly reliable cars, where in fact the use of computer controller has dramatically improved both the reliability and efficiency of the cars [3]. In the transition to CPS, this expectation of reliability will only increase. In fact, without improved reliability and predictability, CPS will not be deployed into such applications as traffic control, automotive safety, and health care. The physical world, however, is not entirely predictable. Cyber physical systems will not be operating in a controlled environment, and must be robust to unexpected conditions and adaptable to subsystem failures[1]. Cyber-physical systems research mainly aims at safety, security, efficiency and to develop high confidence systems in which cyber and physical designs are compatible, synergistic, and all operates at real time. Hence for such systems the processor should be highly calculable and should possess high end capabilities to meet up with unpredictable conditions.[4] This results in the necessity of revitalizing a processor to remain stable ,perform efficiently by reducing the jitter, delay, congestion between tasks and manages energy and power effectively even when maximum resource is utilized.

The Current Approaches are used to develop enormously complex and expensive monolithic humanoid robots. But CPS approaches results in simple, cheap, networked robots. So in this paper we are going to examine a simple, wireless robot which is designed to be used in defense field .Here, the Robot is a computational device which interacts with the physical world through sensors and navigates according to the instructions from the server and completes all tasks within the critical time.

## 2. ARCHITECTURE OVERVIEW

Embedded Based Multifunction Mobile Robot with real time movement mechanical model is developed, which can replace huge man power for various applications in real time environment. The existing systems will perform only the specific tasks like rescue operation or firing, but the proposed model performs various operations like detecting the obstacle, border security, audio video recording, Firing, Obstacle detection, Vibration detection, alarming. DC motors are used for all type of movements like moving forward, reverse, tuning left and right and 360°rotation. Each and every actions of the Robot is controlled from thebase station. The control unit navigates the robot according to the visual pictures captured by the camera..As the Robot is designed to work in the war field ,all the task assigned to it should be completed within critical time otherwise it may result in dangerous instabilities. Therefore all the tasks are designed to be scheduled within the critical time and the complete functionality of the Robot is grouped under various modules.

They are Microcontroller module, RF Module, Sensor module, Control and monitor module, Zigbee module
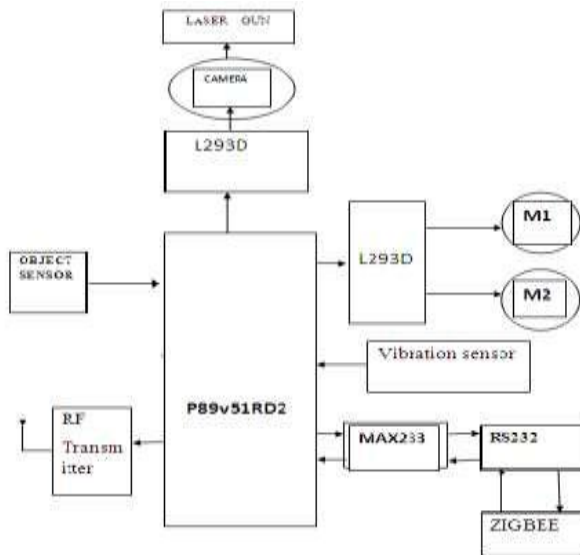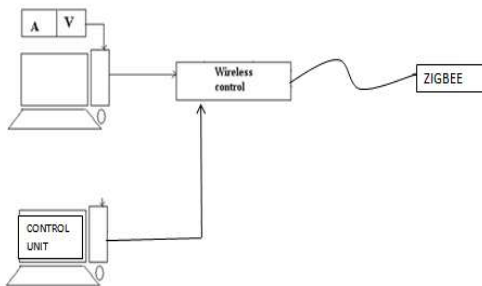


**Fig 1:** Basic Block Diagram of the Robot



**Fig 2**: Base Station

## 3. SOFTWARE IMPLEMENTATION

As discussed earlier, the basic requirement and challenges in CPS, we need software that makes these systems to work effectively in unpredictable conditions. One way to decrease the complexity of application is to use a thread-oriented design and divide a project into more manageable pieces or threads. Each thread is then responsible for some part of the application.. In Such a case when a Professional Rtos is employed it ensures

- That time-critical parts of the code execute within their real-time constraints.
- Make complex applications easier to develop and maintain
- Distribute different parts of the application among several developers.

Selection of a particular RTOS for an application depends on the processor and development tools. Therefore ,Here for a 8051 controller 'Salvo'– 'the RTOS that runs in tiny places' is chosen.Salvo is a commercially available RTOS specially intended for the small embedded system, with a version which works with the Microchip C18compiler.

### 3.1 Salvo Features

Salvo can run multiple prioritised tasks and works with a cooperative scheduler.

- The number of tasks in the fully featured version is limited only by available RAM, while 16 priority levels are available. Tasks can also share priority levels.
- Salvo supports a range of different 'events', including binary and counting semaphores, messages and message queues. Salvo is supplied as a very extensive set of files – source, header, library and others.
- One of Salvo's principal features is that it is highly reconfigurable.

Different libraries are available for use, each one with a different set of features. For a given application ,the most suitable library is chosen. Once this is done, the library services are fixed. The user source code then makes calls to the Salvo functions contained in the chosen library. Apart from the library files, certain other source and header files are also incorporated.

### 4. CONCLUSIONS

A scalable mechanical model that executes Rtos code was implemented and it was made to function efficiently when maximum resource is utilized. This Robot is made to work reliable, stable at unpredictable conditions, thus fulfilling the major requirements of future engineer systems. In the Robot the electronic boards , DC motors, sensors and battery are sensibly integrated .The wheels are fixed to the base plate, which are driven by the two DC motors. Battery is placed over the base plate. The centre plate holds the electronics and the top plate acts as a cover with a camera. Any objects weighing below 25kg ca n be placed over it. The real time mechanical model is fabricated using mechanical grade Teflon sheets.



**Fig 3:** Final Structure

## REFERENCES

[1]. Kathleen Baynes, Chris Collins, Eric Fiterman, , rinda Ganesh, Paul Kohout,**The Performance and Energy Consumption of Embedded Real-Time Operating Systems ,** IEEE transactions on computers, vol. 52, no. 11, November 2003 .

[2]. OS Product Manager Texas Instruments Incorporated Nick Lethaby, **Why Use a Real-Time Operating System in MCU Applications.**

[3]. M. Proctor and W. P. Shackleford, "Real-time operating system timing jitter and its impact on motor control," in SPIE Proceedings of Sensors and Control for Intelligent Manufacturing Conference, 2001.

[4]. Marcio f. s. oliveira, henning zabel, Wolfgang Mueller **assertion-based verification of rtos properties** clab, University of Paderborn, germany, 2005

## BIOGRAPHIE

V. Shanmuga priya M.E. K.C.E.T, Cuddalore (vspriiya@gmail.com) received B.E in Electronics and Communication from Anna University, India in 2009