An RTOS based Wireless Robot for Rescue Operations in Military Applicatons

G. Siva Kumar
M. Tech, Assistant Professor
Department of ECE
ASIST, Paritala, A. P.

T. Sai Kumar
B. Tech, Student
Department of ECE
ASIST, Paritala, A.P.

Y. Srinath
B. Tech, Student
Department of ECE
ASIST, Paritala,A.P.

L. Ram Kumar
B. Tech, Student
Department of ECE
ASIST, Paritala, A. P.

Ch. Sri Pavan Kumar
B. Tech, Student
Department of ECE
ASIST, Paritala, A. P.

B. Mahesh Nayak
B. Tech, Student
Department of ECE
ASIST, Paritala, A. P

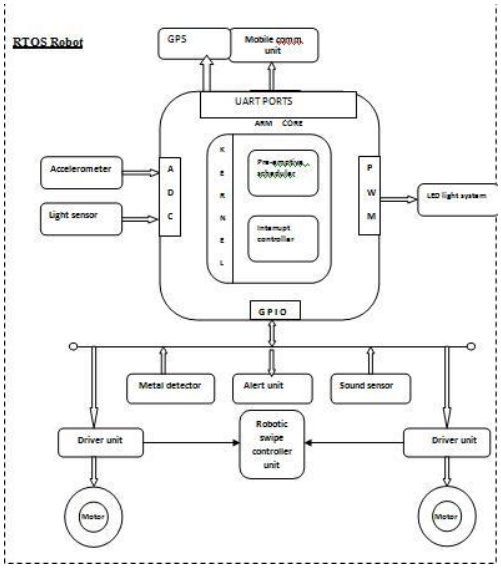
***Abstract*:- A critical time scheduling is a process in this we are assigning a predefined memory space for programming execution and an critical time for completing the particular event in that time for providing this we are used ucos-2 OS. RTOS is a Process which will be done between hardware and application. Here, scheduling is the one which is used to avoid the delay between one application with another, for providing time scheduling for separate tasks and achieving in time we are used ucos-2 based OS. It can able to meet the predefined tasks within the critical time (time previously defined). We are using in the mobile communication to receiving the condition of the border level. Using mobile communication and GPS modem. We are using in the mobile communication to receiving the condition of the border level. Using mobile communication and GSM technology we are giving the indication to the monitoring section. The semantic time scheduling is done all applications at a time without any time delay.**

***Keywords: ROBOTICS, RTOS, GSM, ARM*.**

I. INTRODUCTION

ucos-2 is a highly reliable framework specifically designed for task scheduling in industrial automation data management, controlling and robotics applications .The purpose of a real-time operating system (RTOS) is to schedule tasks in order to guarantee that inputs are acquired and outputs are produced according to timing constraints. In robotic applications, tasks periodically receive information about the environment through sensors or user interfaces, whereas commands to actuators and other outputs are sent at periodic intervals This sensor node is composed of a micro-processors, transceivers, displays and analog to digital converters. Sensor nodes are deployed for military process monitoring and control. The basic view of this technique is to reduce the damages to the human and gives the information about mine in the border section as well as crash detection. If the light intensity is reduced means based on the sensor the lighting system will on condition. Any sound will come due to mine explored it will detect by the sensor and through mobile communication and GPS location values detected through GPS modem for giving accurate location information to monitoring and controlling authorities. it will send information to military section. The project deals with the data receiving from sensor nodes without any delay. The data receiving time is increased with the mobile communication.

ARCHITECTURAL DESIGN:



II. ARM CORE:

.

a 16-bit/32-bit ARM7TDMI-S CPU with real-time emulation and embedded trace support, that combine microcontrollers with embedded high-speed flash memory ranging from 32 kB to 512 kB. A 128-bit wide memory interface and unique accelerator architecture enable 32-bit code execution at the maximum clock rate. For critical code size applications, the alternative 16-bit Thumb mode reduces code by more than 30% with minimal performance penalty. The meaning of LPC is Low Power Low Cost microcontroller. ARM7TDMI-S core is the synthesizable version of the ARM7TDMI core, available in both VERILOG and VHDL, ready for compilation into processes supported by in-house or commercially available synthesis libraries. Optimized for flexibility and featuring an identical feature set to the hard macro cell, it improves time-to-market by reducing development time while allowing for increased design flexibility, and enabling >>98% fault coverage. The ARM720T hard macro cell contains the ARM7TDMI core, 8kb unified cache, and a Memory Management Unit (MMU) that allows the use of protected execution spaces and virtual memory. This macro cell is compatible with leading operating systems including Windows CE, Linux, palm OS, and SYMBIAN OS.

The ARM7EJ-S processor is a synthesizable core that provides all the benefits of the ARM7TDMI – low power consumption, small size, and the thumb instruction set – while also incorporating ARM's latest DSP extensions and Jazelle technology, enabling acceleration of java-based applications. Compatible with the

ARM9™, ARM9E™, and ARM10™ families,

and Strong-Arm® architecture software written for the ARM7TDMI processor is 100% binary-compatible with other members of the ARM7 family and forwards-compatible with the ARM9, ARM9E, and ARM10 families, as well as products in Intel's Strong ARM and xscale architectures. This gives designers a choice of software-compatible processors with strong price-performance points. Support for the ARM architecture today includes:

LPC214x micro controller review:

The internal state of the ARM7TDMI is examined through a JTAG-style serial interface. This allows instructions to be serially inserted into the pipeline of the core without using the external data bus. For example, when in debug state, a Store-Multiple (STM) instruction can be inserted into the pipeline. This exports the contents of the ARM7TDMI registers. This data can be serially shifted out without affecting the rest of the and the instruction set and related decode mechanism are much simpler than those of micro programmed Complex Instruction Set Computers (CISC). This simplicity results in a high instruction throughput and impressive real-time interrupt response from a small and cost-effective processor core.

Pipeline techniques are employed so that all parts of the processing and memory systems can operate continuously. Typically, while one instruction is being executed, its successor is being decoded, and a third instruction is being fetched from memory. The ARM7TDMI-S processor also employs a unique architectural strategy known as Thumb, which makes it ideally suited to high-volume applications with memory restrictions, or applications where code density is an issue.

The key idea behind Thumb is that of a super-

reduced instruction set. Essentially, the ARM7TDMI-S processor has two instruction sets:

- The standard 32-bit ARM set.
- A 16-bit Thumb set.

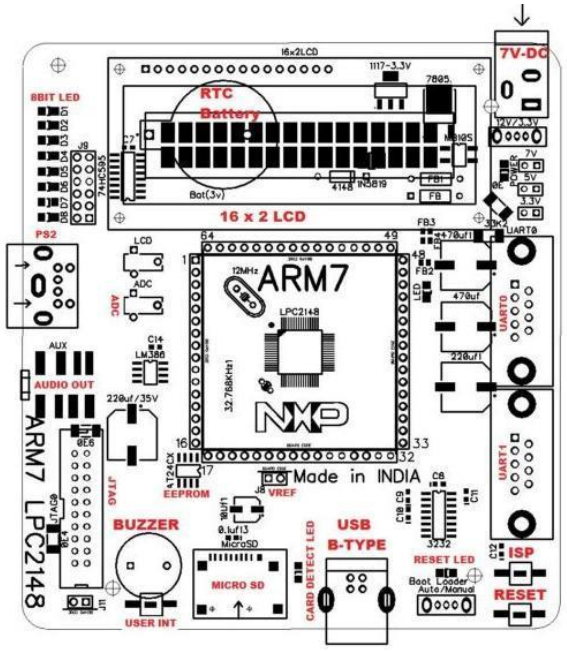


Fig: ARM7TDMI evaluation board architecture

III. GSM MOBILE COMMUNICATION:

A GSM modem is a wireless modem that works with a GSM wireless network. A wireless modem behaves like a dial-up modem A GSM modem can be an external device or a PC Card / PCMCIA Card. Typically, an external GSM modem is connected to a computer through a serial cable or a USB cable. A GSM modem in the form of a PC Card / PCMCIA Card is designed for use with a laptop computer. It should be inserted into one of the PC Card / PCMCIA Card slots of a laptop computer. Like a GSM mobile phone, a GSM modem requires a SIM card from a wireless carrier in order to operate In addition to the standard AT commands, GSM modems support an extended set of AT commands. These extended AT commands are defined in the GSM standards. With the extended AT commands, you can do things like:

- Reading, writing and deleting SMS messages.
- Sending SMS messages.
- Monitoring the signal strength.
- Monitoring the charging status and charge level of the battery.
- Reading, writing and searching phone book entries.

SMS Commands:

AT+CIMI Note:
scan IMSI

AT+CMGS=‖+91818785004‖

AT+CMGR=1

AT+CMGD=1,4

Note: Delete it Note: Message

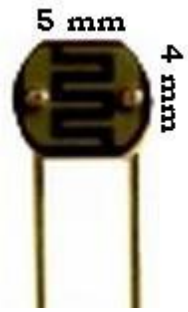
Global system for mobile communication (GSM) is A globally accepted standard for digital cellular Communication.GSM is the name of standardization Group established in 1982 to create a common European mobile telephone standard that would formulate specifications for a pan-European mobile cellular radio system operating at 900 MHz A GSM modem is a wireless modem that works with a GSM wireless network. A wireless modem behaves like a dial-up modem. The main difference between them is that a dial-up modem sends and receives data through a fixed telephone line while a wireless modem sends and receives data through radio waves. The working of GSM modem is based on

commands, the commands always start with AT means Attention) and finish with a <CR>character. For example,

IV. LDR SENSOR:

Although the M1 has a Sunrise / Sunset clock built in that will determine when the sunrises and sets, hence if it is dark or light outside, often inside light is a totally different subject. The system needs to know what the light level is in a particular room so when automating internal lighting it needs to know if the lights should be activated or not. Otherwise it defeats the purpose of energy saving by automating the lights for cost savings.

One way of doing this is with a $5.00 item from Ness with our Ness-LDR. This LDR wires directly into a M1 Zone Input (Any Zone). The Zone need to be programmed as a Analog Zone. The more light the LDR sensor has on it the lower the voltage the zone will read and the lower the light level, the higher. The zone voltage. The following table will provide a summary of the type of voltages v's light (Lux) you



Could expect to read. As the Ness LDR is very small (approx. 5 mm x 4mm x 2 mm) it can be installed anywhere. Although it can be installed on a PIR detector consideration must be given as to the amount of light near the ceiling in a corner compared to lower near the floor. As a suggestion you could mount it on a blank electrical plate attached to the wall near the floor / power point level where the light is more even. This would change from site to site; room by room The LDR Sensor is wired directly to any Zone input. (Even the Keypad Zone input (where a good location for the LDR could be on the keypad!)) It does not need power.

Frequencies: Originally it had been intended that GSM would operate on frequencies in the 900 MHz cellular band. In September 1993, the British operator Mercury One-to-One launched a network. Termed DCS 1800 it operated at frequencies in a new 1800 MHz band. By adopting new frequencies new operators and further competition was introduced into the market apart from allowing additional spectrum to be used and further increasing the overall capacity.

V. GSM NETWORK:

GSM provides recommendations, not requirements. The GSM specifications define the functions and interface requirements in detail but do not address the hardware. The reason for this is to limit the designers as little as possible but still to make it possible for the operators to buy equipment from different suppliers. The GSM network is divided into three major systems: the switching system (SS), the base station system (BSS), and the operation and support system (OSS).

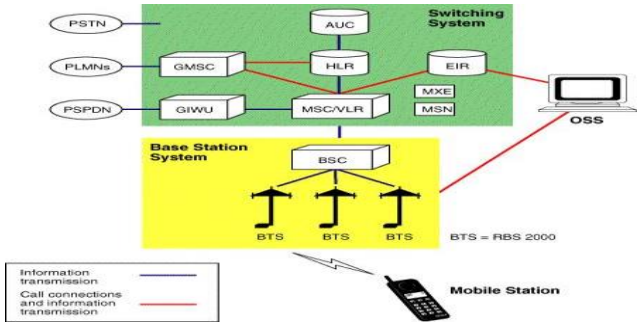


Fig:GSM Network Elements

***RTOS*:**

RTOS is an operating system which is used to perform a task with in a particular time

interval i.e. within the specific allocated time. It is a real time operating system. A real-time OS that can usually or generally meet a deadline is a soft real-time OS, but if it can meet a deadline deterministically it is a hard real-time OS. Compared with OS and RTOS, RTOS only supports the multitasking operations and time scheduling tasks. Real-time OS is the level of its consistency concerning the amount of time it takes to accept and complete an application's task. If we are implementing any task without RTOS, it is less accuracy and time delay of the specified time and normally it can possible to perform only one task at a time. So in normal operations systems perform a task one by one. So we are implementing our project using real time operating system.

The multitasking is a process to perform a more than one application or task at concurrently, it means possible to perform a so many operations at the same time. In the normal operating systems are not supported this type of multitasking. So in this project we are implementing RTOS concepts. As The main advantage of RTOS is multitasking and time scheduling and rescheduling etc. In RTOS due to the internal minimum time delay of the time scheduling process it will give the output with in the specified time.

However, due to the lack of uniform programming model and system components for these different teams, the migrations costs of a function model from software to hardware are high. But these actions are necessary in the hardware-software partitioning of embedded systems, especially in the prototype designs. To cope with this problem, we adopt a uniform multi-task model and implement UCOS II RTOS (Red- Time Operating System).

MEMS Sensor

MEMS accelerometers are one of the simplest but also most applicable micro-electromechanical systems. They became indispensable in automobile industry, computer and audio-video technology. This seminar presents MEMS technology as a highly developing industry. Special attention is given to the capacitor accelerometers, how do they work and their applications. The seminar closes with quite extensively described MEMS fabrication.

An accelerometer is an electromechanical device that measures acceleration forces. These forces may be static, like the constant force of gravity pulling at our feet, or they could be dynamic - caused by moving or vibrating the accelerometer. There are many types of accelerometers developed and reported in the literature. The vast majority is based on piezoelectric crystals, but they are too big and to clumsy. People tried to develop something smaller, that could increase applicability and started searching in the field of microelectronics.

They developed MEMS (micro electromechanical systems) accelerometers. The

first micro machined accelerometer was designed in 1979 at Stanford University, but it took over 15 years before such devices became accepted mainstream products for large volume applications [1]. In the 1990s MEMS accelerometers revolutionized the automotive-airbag system industry. Since then they have enabled unique features and applications ranging from hard-disk protection on laptops to game controllers. More recently, the same sensor-core technology has become available in fully integrated, full-featured devices suitable for industrial applications. Micro machined accelerometers are a highly enabling technology with a huge commercial potential. They provide lower power, compact and robust sensing. Multiple sensors are often combined to provide multi-axis sensing and more accurate data.

CONCLUSION

Finally from whole this concept we discussed about a strong time scheduling critical task execution systems for industrial automation and rescue searching activities through UART a framework designed to deal with time period programming and reconfiguration of task sets depending on the present context and on the

―semantic content of tasks.‖ this is often a haul that's typically left within the background by researchers within the field of intelligent robotic systems. Here, the matter has been formally outlined, the answer Implemented by UART has been delineate intimately, and its theoretical properties are mentioned.

REFERENCES:

[1] M. Piaggio, A. Sgorbissa, and R. Zaccaria, ―Pre-emptive versus non-preemptive real time scheduling in intelligentmobile robotics,‖ *J. Exp. Theory. Artif. Intell.*, vol. 12, no. 2, pp. 235–245, 2000.

[2] H. Bruyninckx, ―Open robot control software: The OROCOS project,‖ in *Proc. IEEE Int. Conf. Robot. Autom.*, Seoul, Korea, May 2001, vol. 3, pp. 2523–2528.

[3] L. B. Becker and C. E. Pereira, ―SIMOO-RT—An object-oriented framework for the development of real-time industrial automation systems,‖ *IEEE Trans. Robot.*, vol. 18, no. 4, pp. 421–430, Aug. 2002.

[4] R. Brennan, M. Fletcher, and D. Norrie, ―An agent-based approach to reconfiguration of real-time distributed control systems,‖ *IEEE Trans. Robot. Autom.*, vol. 18, no. 4, pp. 444–451, Aug. 2002.

[5] I. A. D. Nesnas, A. Wright, M. Bajracharya, R. Simmons, and T. Estlin,―CLARAty and challenges of developing interoperable robotic software,‖ in *Proc. 2003 IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2003, pp. 2428– 2435.

[6] A. Brooks, T. Kaupp, A. Makarenko, S. Williams, and A. Oreback, ―Towards component-based robotics,‖ in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2005, pp. 163–168.

[7] Y. hsin Kuo and B. MacDonald, ―A distributed real-time software framework for robotic applications,‖ in *Proc. IEEE Int. Conf. Robot. Autom.*, 2005, pp. 1964–1969.