International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 – 5161

Available at http://inpressco.com/category/ijcet

Research Article

Design and Implementation of Real-Time Task's Scheduling on ARM processor

Boppani Krishna Kanth[†*] and G. Bhaskar Phani Ram[†]

[†]Department of ECE, Vardhaman Engineering College, Hyderabad,Telangana state, India

Accepted 31 July 2015, Available online 03 Aug 2015, **Vol.5, No.4 (Aug 2015)**

Abstract

This paper is an RTOS based architecture designed for the purpose of mine detection. 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. We are using in the mobile communication to receiving the condition of the border level. Using mobile communication 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.*

1. Introduction

The job of the RTOS is to manage the allocation of these resources to users in an orderly and controlled manner. 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. 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 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.

2. Hardware Design

The ARM7 family includes the ARM7TDMI, ARM7TDMI-S, ARM720T, and ARM7EJ-S processors. The ARM7TDMI core is the industry's most widely used.

ARM7TDMI solution provides the low power consumption, small size, and high performance needed in portable, embedded applications. The 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.

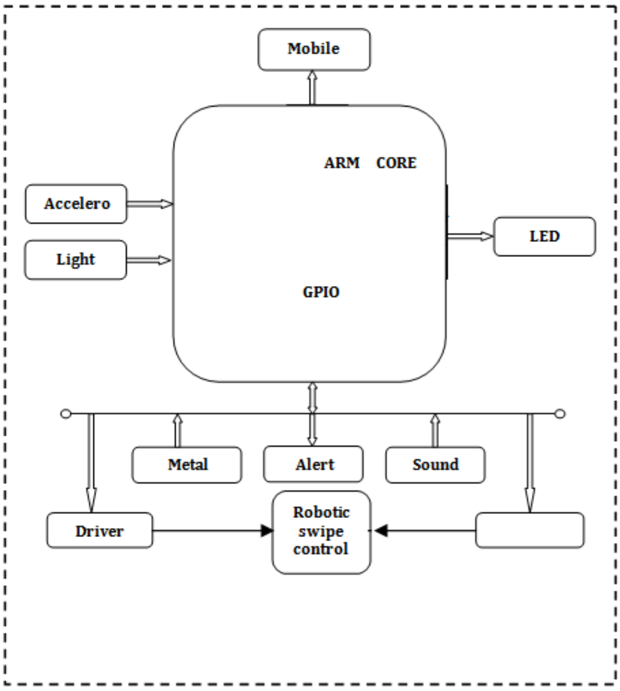

Fig 1: Block Diagram

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

*Corresponding author: **Boppani Krishna Kanth; G. Bhaskar Phani Ram** is working as Assistant Professor

macro cell is compatible with leading operating systems including Windows CE, Linux, palm OS, and SYMBIAN OS.

LPC2148 Processor

LPC2148 Microcontroller Architecture. The ARM7TDMI-S is a general purpose 32-bit microprocessor, which offers high performance and very low power consumption. The ARM architecture is based on Reduced Instruction Set Computer (RISC) principles, 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 corePipeline techniques are employed so that all parts of the processing and memory systems can operate continuously.

Fig: 2 ARM7TDMI PCB board

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.

The Thumb set's 16-bit instruction length allows it to approach twice the density of standard ARM code while retaining most of the ARM's performance advantage over a traditional 16-bit processor using 16-bit registers. This is possible because Thumb code operates on the same 32-bit register set as ARM code. Thumb code is able to provide up to 65% of the code size of ARM, and 160% of the performance of an equivalent ARM processor connected to a 16-bit memory system.

GSM

Global System for Mobile Communications or GSM (originally from Groupe Special Mobile) is the world's most popular standard for mobile telephone systems. The GSM Association estimates that 80% of the global mobile market uses the standard. GSM is used by over 1.5 billion people across more than 212 countries and territories. This ubiquity means that subscribers can use their phones throughout the world, enabled by international roaming arrangements between mobile network operators. GSM differs from its predecessor technologies in that both signaling and speech channels are digital, and thus GSM is considered a second generation (2G) mobile phone system. The GSM standard has been an advantage to both consumers, who may benefit from the ability to roam and switch carriers without replacing phones, and also to network operators, who can choose equipment from many GSM equipment vendors.

SMS Commands

¬AT+CIMI
Note: scan IMSI
¬AT+CMGS=+919704040791
¬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, the dialing command is ATD<number>; ATD3314629080; here the dialing command ends with semicolon. The AT commands are given to the GSM modem with the help of PC or controller. The GSM modem is serially interfaced with the controller with the help of MAX 232.

Global usage

Originally GSM had been planned as a European system. However the first indication that the success of GSM was spreading further afield occurred when the Australian network provider, Telstra signed the GSM Memorandum of Understanding.

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. This trend was followed in many countries, and soon the term DCS 1800 was dropped in favor of calling it GSM as it was purely the same cellular technology but operating on a different frequency band. In view of the higher frequency used the distances the signals travelled was slightly shorter but this was compensated for by additional base stations. In the USA as well a portion of spectrum at 1900 MHz was allocated for cellular usage in 1994. The licensing body, the FCC, did not legislate which technology should be used, and accordingly this enabled GSM to gain a foothold in the US market. This system was known as PCS 1900 (Personal Communication System)

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).

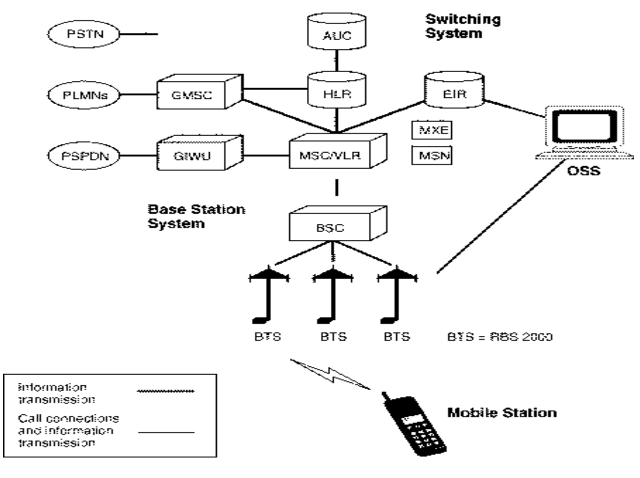

Fig: 3 GSM Network Elements

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. a 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. In the 1990s MEMS accelerometers revolutionised 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.

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.

Fig.4: LDR Sensor

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.

3. Results

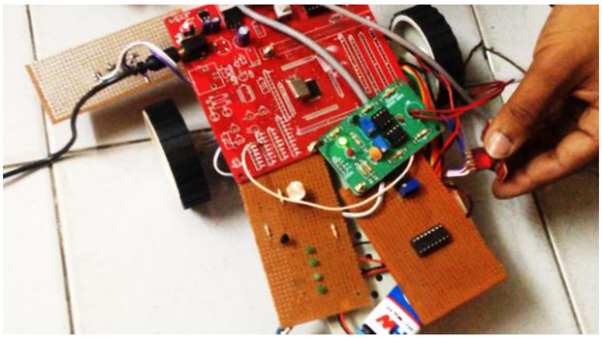

Fig.5: Kit hardware

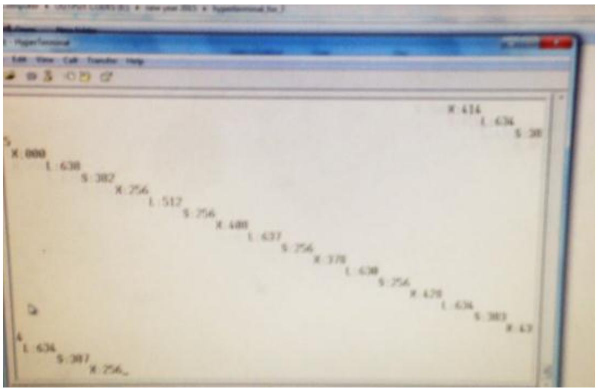

Fig.6 Values for Mems Sensor

Conclusion

This paper introduced UART a framework designed to deal with time period programming and reconfiguration of task sets depending on the 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

M. Piaggio, A. Sgorbissa, and R. Zaccaria (2001), Pre-emptive versus non-preemptive real time scheduling in intelligentmobile robotics, *J. Exp. Theor. Artif. Intell.*, vol. 12, no. 2, pp. 235–245

H. Bruyninckx (May 2001), Open robot control software: The OROCOS project, in *Proc. IEEE Int. Conf. Robot. Autom.*, Seoul, Korea, vol. 3, pp. 2523–25

L. B. Becker and C. E. Pereira (Aug. 2002), 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

R. Brennan, M. Fletcher, and D. Norrie (Aug. 2001), An agent-based approach to reconfiguration of real-time distributed control systems, *IEEE Trans. Robot. Autom.*, vol. 18, no. 4, pp. 444–451, I.

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

A. Brooks, T. Kaupp, A. Makarenko, S. Williams, and A. Oreback (2005), Towards component-based robotics, in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, pp. 163–16

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

Mr. Boppani Krishna Kanth is working towards a Master of Technology in Embedded systems in prestigious Vardhaman College of Engineering, *R.R.Dist*, Telangana state, India. He obtained B.Tech from Joginpally B R Engineering College (Electronics and Communication Engineering in 2012

Mr.G.Bhaskar Phani Ram is presently working as an Assistant Professor of Electronics Communication Engineering in prestigious Vardhaman College of Engineering, R.R.Dist, Telangana state, India. He obtained M.Tech Embedded Systems in JNTU University, Hyderabad in 2011