# A RTOS Based Reconfigurable Architecture for IWSN Stack with Arm Cortex and EM250RF Processor Support

**L. Divya[1], V. Maheswari[2]**

[1]PG Scholar (EEE), Ganadipathy Tulis's Jain Engineering College, Kaniyambadi, Vellore, India

[2]Assistant professor, Ganadipathy Tulis's Jain Engineering College, Kaniyambadi, Vellore, India

**Abstract:** *Industrial wireless sensor networks are deployed in noisy environments for process measurement and control applications. The design of RTOS based industrial wireless sensor network (IWSN) consists of wireless sensor are installed on industrial equipment and monitor the parameters to each equipment's such as temperature, flow, level and pressure quality. The route to transfer the data to base station we concentrate on reconfiguring the path, when failure occurs. We propose a Round-robin (R) algorithm in which each task gets a small unit of time on the system, handling all tasks without priority. It offers benefits in terms of path independency, security, reliable communication, and performance scalability*

**Keywords:** IWSN, RTOS, ARM CORTEX, EM250, Round Robin

## 1. Introduction

Wireless process control has been increasingly recognized as an important technology in the field of industrial process management. In recent years three standards have been released Wireless HART [1], ISA100 [2] and WIAPA [3] for measurement and control application. The hardware challenges in Wireless HART were discussed [4] to provide timing integrity but more memory and power consumption. TDMA is chosen by Wireless HART to determine when and how a node transmits data, which provide better guarantee. Thus fully implemented Wireless HART communication stack requires more computational power and space is required.

The multi processor architecture challenges [5] achieve software flexibility, no of task executed concurrently. To achieve more efficiency we can easily replace a single CPU with less capable of interconnect devices used to control the data from different sensors. Clock flexibility is provided by the use of parallel execution of task for specific application. High performance chips are used for IWSN, but low cost. The open wireless standard of Wireless HART stack [6] was implemented to generate timing synchronization, routing and reliable frame work. Each and every sensor nodes are maintained at every second and the collection of tables are maintained necessarily. Network manager is responsible for maintain updating information about the routing table and communication schedule. Two routing protocol are used, Graph routing and source routing.

Control and maintenance of routing table is somewhat difficult. To improve the productivity of industrial system IWSN provides highly reliable and predictable framework [7] to address the challenges of reducing noise and interference in coverage area by using radio technologies. But it has only limited battery life and difficult to implement the cross layer implementation. Reliable communication and timing integrity is Performed and achieved [8] by graph routing algorithm. Timing integrity is achieved by using least laxity fit algorithm. In this timeslots are shared to

multiple devices. But they are only focused more on the communication architecture rather than the hardware architecture implementation.
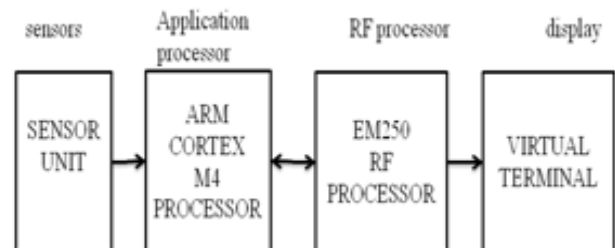
## 2. Hardware Design



**Figure 1:** Block Diagram

The design of IWSN stack consist of sensors like temperature, pressure, flow and level sensor, these sensors outputs are scheduled by the processor and transmitted through the EM250 processor and displayed by the display unit.ARM CORTEX M4 (Stellaris Microcontroller) LM4F120 family of ultra-low-power microcontrollers consists of several devices featuring different sets of peripherals targeted for various applications. The architecture, combined with five low-power modes, is optimized to achieve extended battery life in portable measurement applications. The device features a powerful 32-bit RISC CPU, 32-bit registers, and constant generators that contribute to maximum code efficiency. The digitally controlled oscillator allows wake-up from low-power modes to active mode in less than 1μs. The LM4F120 series are ultra-low-power digital signal microcontrollers with built-in 32- bit timers, up to 24 I/O touch-sense-enabled pins, a versatile analogy comparator, and built-in communication capability using the universal serial communication interface. The Ember EM250 system-on-chip (SoC) combines a 2.4 GHz IEEE 802.15.4 compliant radio transceiver with a 16-bit microprocessor. The radio provides outstanding RF performance with excellent sensitivity and transmits power for long range, and 802.11 immunity.

### 2.1 EM250

A flexible RF interface minimizes external components in configurations. Dedicated high Performance chips for IWSN are rare, but low cost also high performance IEEE802.15.4 system-on-chip (SoCs) is very common in commercial markets. It is attractive if we can use these low cost commercial chips combining with high performance industrial processors, so called "*low-high combination*". So, the multi-processor support of IWSN stacks is a realistic requirement for the best trade-off between performance and cost. This modules are designed with low to medium transmit power and for high reliability wireless network. It requires minimal power and provides reliable delivery of data between devices. The interface provided with the module help to directly fit into many industrial application. The module operate within the ISM 2.4-2.4835 GHz frequency band with IEEE 802.15.4 baseband. The EM250 device is optimized for designs requiring long battery life, low external component count, and a reliable, proven, industry-standard networking solution. Innovative on-chip debugging via the packet trace port provides developers with the most advanced views into their application and network available in the industry.

## 3. Software Design

The rapid development of DSP and high performance embedded microprocessor does not make the protection function more perfect and reliable, but also can gather and deal with more abundant data. The system needs precise and fast inner communicating function for transmission of vast data, and it also needs the network. It is difficult to achieve the requirement of the high-speed dealing with the data from DSP, including using parallel communication, in traditional monitor system and programming technique, so that the choice of the more intelligent embedded system with RTOS is needed imminently. It is the trend of developing monitor system with high-performance embedded system. And it is very easy to increase the speed CPU, the capability of memory and especially the design of linear program.

The traditional monitor system brings us the bad configuration of the program, low efficiency on developing, and so on, so that the embedded system of RTOS is chosen for the new research and development platform of monitor system. Separating each function as an individual task, while the system should distribute resource to ever task according to their Priority levels, which could ensure the real time response of multi-task? It is better to realize abundant manage functions in the monitor system than traditional one.

### 3.1 SYS/BIOS

SYS/BIOS are a scalable real-time kernel. It is designed to be used by applications that require real-time scheduling and synchronization or real-time instrumentation. SYS/BIOS is designed to minimize memory and CPU requirements on the target. SYS/BIOS is an advanced real-time operating system from Texas Instruments for use in a wide range of DSPs, ARMs, and microcontrollers. It is designed for use in embedded applications that need real-time scheduling, synchronization, and instrumentation. SYS/BIOS Provide a wide range of system services such a Pre-emptive,

deterministic multi-tasking, Hardware abstraction, Memory management Configuration tools and Real-time analysis. SYS/BIOS provide many benefits that make it a great operating system for use in embedded applications running on TI DSPs, ARMs, and MCUs. Used powerful real-time C code. Energia is software created by Texas Instrument (TI) for board implementation of stellaris microcontroller, Within Code Composer Studio (CCS).Writing normal C code that calls SYS/BIOS APIs for threading, synchronization, memory management, and error handling. Many embedded applications need to perform various functions at the same time but at different frequencies. SYS/BIOS provide a wide range of system services to an embedded application such as pre-emptive, memory management and real-time analysis. And it was designed to be highly configurable.

### 3.2 TI-RTOS

TI-RTOS is a real-time operating system for TI (Texas instrument) microcontrollers. TI-RTOS enables very faster development by eliminating the need for developers to write and easy to control maintain system software such as schedulers, protocol stacks and drivers. It combines a real-time multitasking kernel with additional middleware components including TCP/IP and USB stacks, a FAT fi le system, and device drivers, enabling developers to focus on differentiating their application. TI-RTOS Kernel formerly known as SYS/BIOS is an advanced, real-time kernel for use in a wide range of DSPs, ARMs, and microcontrollers. It provides scheduling, multitasking, hardware abstraction, and memory management. TI-RTOS Kernel is at the core of TI-RTOS, a full-featured real-time operating system including drivers, networking and USB stacks.

### 3.3 Real Time Scheduling Algorithms

Some of the well known real-time scheduling algorithms are described as follows. **Clock Driven Scheduling** usually precomputed which task would run when and store this schedule in a table at the system is designed and configured. Rather than Cyclic scheduling is static – computed offline and stored in a table. **Rate Monotonic Algorithm (RM)** is a fixed priority scheduling algorithm which consists of assigning the highest priority to the highest frequency tasks in the system, and lowest priority to the lowest frequency tasks. At any time, the scheduler chooses to execute the task with the highest priority. **Earliest-Deadline-First Algorithm (EDF)** uses the deadline of a task as its priority. The task with the earliest deadline has the highest priority, while the task with the latest deadline has the lowest priority.

### 3.4 Round Robin Scheduling

In order to schedule processes fairly, a round-robin scheduling generally employs starvation free, interactive time-sharing and maximize the throughput by providing each process gets a small unit of time on CPU (time quantum or time slice).The job is resumed next time a time slot is assigned to that process. In the absence of time-sharing, or if the quanta were large relative to the sizes of the jobs, a process that produced large jobs would be favoured over other processes.

**Table 1:** Input Components for the Processor

| Task | Burst time (milliseconds) | Priority |
|------|---------------------------|----------|
| 1 | 4 | 1 |
| 2 | 5 | 2 |
| 3 | 20 | 1 |
| 4 | 20 | 3 |

The above four task has been scheduled using simple Round Robin architecture. The time slice of four milliseconds has been used. In round robin algorithm no task is allocated for more than one time slice in a row. If the task exceeds one time slice, the concern task will be pre-empted and put into the ready queue. The task is pre-empted after the first time quantum and the CPU is given to the next process which is in the ready queue (task P2), similarly schedules all the process and completes the first cycle In the second cycle task P2 requires one millisecond time slice (doesn't require four milliseconds time slice), so it quits before its time quantum expires. The CPU is given to next taskP3. In the same way all other processes in the system are scheduled .The process time slicing in simple Round Robin architecture is shown in fig-2.

| P1 | P2 | P3 | P4 | P2 | P3 | P4 |
0 4 8 12 16 17 21 25 1 6 17 21 25

| P3 | P4 | P3 | P4 | P3 | P4 |
25 29 33 37 41 45 49
**Figure 2:** Time Slicing in Round Robin Architecture

**Table 2:** Comparison of Clock driven &Round robin

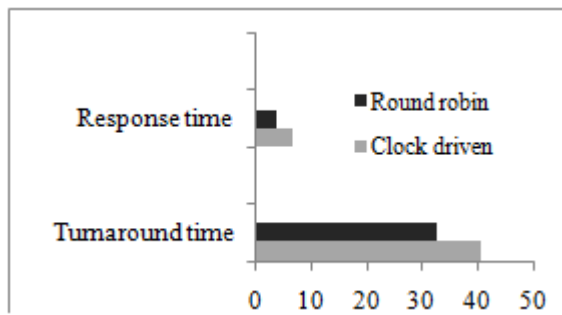| Algorithm | Average turnaround time(ms) | Average response time(ms) |
|-----------|------------------------------|---------------------------|
| Clock driven | 40.3 | 6.3 |
| Round robin | 32.3 | 2.5 |


**Figure 3:** Comparison of Performance of Algorithms
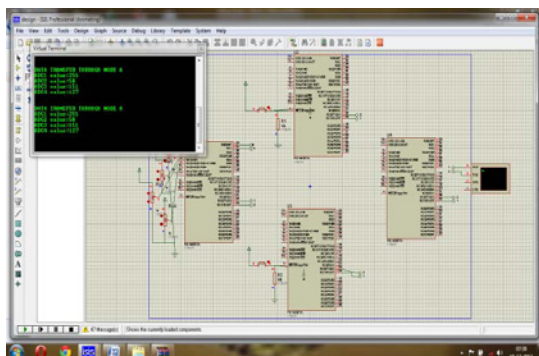
## 4. Simulation


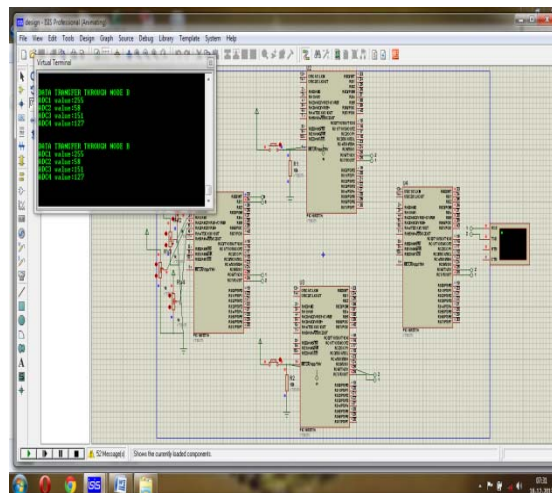**Figure 4:** Data transmit to node A


**Figure 5:** Data transmit to node B

All sensors outputs scheduled by application processor using RTOS concept from the sensors and transmitted to the base station through the radio processor. Path for data transmission can reconfigured automatically if any path failure occur. Fig shows for transmission of data from node A and node B.

## 5. Conclusion

We have observed the RTOS based architecture is used to transmit the huge amount of information to the base station. All the sensor values are monitored and scheduled by Sterllaris application processor. The transceiver of radio processor used to transmit the data and reduces the power consumption and increases the battery life of the system. UART communication is performed for data transmission. Reconfiguration of the path can be when the node failure occurs**.** We can get the data more easily, which also more accurate than normal system by the RTOS scheduling.

## 6. Future Work

A comparison of clock driven and time sliced round robin architecture is made. It is concluded that the architecture is superior as it has less turnaround time, response times, usually less preemption and saving of memory space. Future work can be based on this architecture modified and implemented for hard real time system where hard deadline systems require event driven scheduling algorithm.

## References

[1] Wireless HART standard, HART Communication Foundation. www.hartcomm.org. [2] ISA100 standard, International Society of Automation.www.isa.org/isa100.
[2] WIA-PA standard, Chinese Industrial Wireless Alliance.www.industrialwireless.cn/en/.
[3] Xiuming Zhu; Song Han; Mok, A.; Deji Chen;Nixon, M. Hardware challenges and their resolution in advancing WirelessHART. IEEE Int. Con. on Industrial Informatics (INDIN), 2011, 416-421.
[4] Utz Roedig, Sarah Rutlidge, James Brown, Andrew Scott, "Towards Multiprocessor Sensor Nodes". ACM Workshop on Hot Topics in Embedded Networked Sensors (HotEmNets '10).

[5] Jianping Song; Song Han; Mok, A.K.; Deji Chen; Lucas, M.; Nixon, M..WirelessHART: Applying Wireless Technology in Real-Time Industrial Process Control. IEEE Real-Time and Embedded Technology and Applications Symposium, 2008.

[6] Vehbi C. Gungor, Member, IEEE, and Gerhard P. Hancke, Senior Member, IEEE Industrial Wireless Sensor Networks: Challenges, Design Principles, and Technical Approaches.

[7] Song Han, Xiuming Zhu, Aloysius K. Mok;Deji Chen, Mark Nixon Reliable and Real-time Communication in Industrial Wireless Mesh Networks.