Journal of Information Systems and Communication
ISSN: 0976-8742 & E-ISSN: 0976-8750, Volume 3, Issue 1, 2012, pp.-182-186.
Available online at http://www.bioinfo.in/contents.php?id=45

DESIGN OF LOW COST LOW POWER MOTES FOR WIRELESS SENSOR NETWORKS USING MSP430

KULKARNI N.S.[1], BHARGAVA S.[1], RAKESH R.[2], SHANTANU SINGH BUNDELA[3] AND RAJESHWARI HEGDE[4]

[1]AGC Networks Ltd, Bangalore, India
[2]Tecnotree Convergence Ltd, Bangalore, India
[3]Robert Bosch Engg and Business Solutions Ltd, Bangalore, India
[4]Dept of Telecommunication Engineering, BMS College of Engineering, Bangalore, India
*Corresponding Author: Email- nikhilkulkarni_88@yahoo.com, rajeshwari.hegde@gmail.com

Received: January 12, 2012; Accepted: February 15, 2012

Abstract- Wireless Sensor Networks (WSNs) consist of a network of wireless nodes that have the capability to sense a parameter of interest. The sensed parameter is relayed to a base station through the network formed amongst these nodes. The devices used are typically characterized by low cost, low power and are rugged in operation. They are commonly referred to as motes in the WSN domain. The mote integrates programming, computation, communication, and sensing onto a single system and provides an easy user interface for operating and deploying it. The paper presents such a design which minimizes cost, size and current consumption, thus enhancing the life time of the mote.
Keywords- Motes, WSN, Zigbee, MSP430, CC2420

Citation: Kulkarni N.S., et al (2012) Design of Low Cost Low Power Motes for Wireless Sensor Networks Using MSP430. Journal of Information Systems and Communication, ISSN: 0976-8742 & E-ISSN: 0976-8750, Volume 3, Issue 1,pp.-182-186.

Introduction

The advances in the hardware and wireless technologies have resulted in inexpensive low power communication devices that can be deployed throughout a physical space, providing dense sensing close to physical phenomena, processing and communicating this information, and coordinating actions with other nodes. Such a deployment can be termed as a Wireless Sensor Network (WSN). To realize such a network, we must address a new collection of challenges. The individual devices in a WSN are inherently resource constrained: they have limited processing speed, storage capacity, battery capacity, and communication bandwidth. These devices have substantial processing capability in the aggregate, but not individually. These individual devices are referred to as motes in WSN domain.

A 'mote', is a node in a wireless sensor network that is capable of gathering information, processing and communicating with other connected nodes in the network. Typically the mote may contain one or more sensors that can monitor the surroundings for specific parameters. Some of these sensors commonly used are to sense temperature, light, sound, position, acceleration, vibration, stress, weight, pressure, humidity, etc. The microcontroller performs all the data processing tasks and controls the functionality of other components in the sensor node. The sensors measure data of the area to be monitored. The continual analog signal sensed by the sensors is digitized by an Analog-to-digital converter and sent to controllers for further processing. The motes also contain the 'Zigbee' module which provides communication over wireless medium using transceivers. Motes can be powered by using batteries.

A large number of motes hence communicate over wireless channel using the ZigBee module and form an ad-hoc network. All the information can eventually be transmitted to a gateway node. By forming an ad-hoc network they can function for a long time without any human intervention since they consume limited power.

In this paper we aim to design motes that consume low power and have a lower overall cost when compared to commercially available ones. They should also be able communicate over the wireless medium using the ZigBee protocol. We have evaluated and selected components which fit our basic requirements of low power consumption, low cost, small size and processing speed. The remainder of this paper is organized as follows: Section II deals with the related work. Session III deals with the overview of wireless sensor network. Section IV, presents various aspects of Zigbee. Section V gives the system design. Section VI explains the implementation. Section VII deals with results and discussion. The paper is concluded in section VIII.

Related Work

The usage of motes was conceived and implemented by the Smart Dust project[1]. The Smart Dust project explored the possibility of using motes to form a massive distributed network. The main goal of the project was to explore micro fabrication technology's limitations and study the feasibility of such a platform. Because of the mote's discrete size, substantial functionality, connectivity, and anticipated low cost, Smart Dust was envisaged to facilitate innovative methods of interacting with the environment, and provide more information from physical regions less intrusively.

The intended applications for smart dust were:

- Deploy sensor networks rapidly by unmanned aerial vehicles or artillery in battlefield scenarios and track enemy movements.
- Tracking the movements of birds, small animals, and insects and understand their behavioral patterns
- Monitor health of rotating machinery and understand the reasons for high cycle fatigue.

Table 1- [2] shows the comparison of the existing motes

Table1- Comparison of existing motes

Mote Type	WeC	Rene	Rene2	Dot	Mica	Mica2Dot	Mica2	Telos
Processor	AT90LS8535		ATMega163		ATMega128			MSP430
Radio Chip	TR1000					CC1000		CC2420
Flash	32 KB				512 KB			128 KB
Sensors	No on-board sensors			Yes	No on-board sensors			Yes
Frequency of Operation	916.5 Mhz					868 Mhz		2.4 Ghz
Transmit Power	0 dBm							
Max Data Rate	10 Kbps				40 Kbps	38.4 Kbps		250 Kbps
Power Consumption	36 mW					42 mW		35 mW
Sleep Power (μW)	45				75			6
wakeup Time (μS)	1000		36		180			6
Interface	IEEE 1284 and RS232							USB

There have been a number of platforms which were developed on the lines of smart dust motes. The initial devices incorporated small microcontrollers (8 bit, 4 KB flash) and a simple radio (4 Kbps data rate) and their life time was up to a maximum of 2 years. As can be seen from the table, the main focus in the design of the motes is low power of operation. Almost all the microcontrollers used in various designs operate at low voltages of 2.7 volts to 3.6 volts. Another important factor for the choice of microcontroller is the current drawn during its sleep and wake up time. Low sleep current and low wake up times minimize the duty cycle of a mote and therefore extends the life of a mote and consequent network life time. Telos design uses MSP430 microcontroller which has lowest sleep and wake times as compared to the other microcontrollers operating at low voltages. On-chip RAM also plays a vital role in the processing capability of the mote and MSP430 offers the maximum on-chip RAM in comparison. Another salient factor in the mote's evolution is their usage of radio chips for communication. With the introduction of802.15.4 standard for wireless sensor networks, the motes too started incorporating chips that use these standards for interoperability of platforms.

Overview Of Wireless sensor Network

Sensor networks are applied to various fields ranging from special application fields such as wild environment monitoring, industrial machine measurement and military purpose measurement to daily application fields such as fire monitoring and pollution monitoring. [3]. A wireless sensor network is a wire and wireless network, which consists of several sensor nodes deployed in a certain field. A sensor node should have computation, sensing and wireless communication functions. A wireless sensor network limits the radio frequency channel, due to, that is to say, unstable links, limit of physical protection of each sensor node, actual of each nodes connection, variation topology in addition dangerousness about routing security is high by activity spite nodes. In addition, restrictions of the hardware of the sensor node itself makes it difficult guarantee the maintenance of security because of vulnerability.[4] [5].

A wireless sensor network (WSN) consists of spatially distributed autonomous sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants. A sensor network normally constitutes a wireless ad-hoc network, meaning that each sensor supports a multi-hop routing algorithm. The network does not rely on a pre-existing infrastructure, such as routers in wired networks or access points in managed (infrastructure) wireless networks. Instead, each node participates in routing by forwarding data for other nodes, and so the determination of which nodes forward data is made dynamically based on the network connectivity.

The base stations are one or more distinguished components of the WSN with much more computational, energy and communication resources. They act as a gateway between sensor nodes and the end user.

Why ZigBee

The name "ZigBee" is derived from the erratic zigzag patterns many bees make between flowers when collecting pollen. This is suggestive of the invisible webs of connections existing in a fully wireless environment, similar to the way packets would move through a mesh network.

During the last few years, the ZigBee Alliance has made significant modification and improvement on the ZigBee standard IEEE 804.15.4, making it more applicable to the increasing demand on Personal Area Network (PAN) service. However, the ZigBee standard specified that the maximum data rate of a ZigBee link be 250kbps. This data rate faces many difficulties dealing with the increasing data transmission pressure in many applications [6]. ZigBee is a low-cost, low-power wireless mesh networking proprietary standard. The low cost allows the technology to be widely deployed in wireless control and monitoring applications, the low power-usage allows longer life with smaller batteries and the mesh networking provides high reliability and larger range. The technology defined by the ZigBee is intended to be simpler and less expensive than other WPANs, such as Bluetooth. ZigBee wireless sensor network has great advantages in terms of low power consumption, high fault tolerance, flexibility, and autonomy.

ZigBee operates in the industrial, scientific and medical (ISM) radio bands; 868 MHz in Europe, 915 MHz in the USA and Australia, and 2.4 GHz in most jurisdictions worldwide. The ZigBee specifications are available free of cost for all non-commercial purposes. ZigBee can go from sleep to active mode in 15 msec or less, thus the latency can be very low and devices can be very responsive particularly compared to Bluetooth wake-up delays, which are typically around three seconds. Because ZigBee can sleep most of the time, average power consumption can be very low, resulting in long battery life. It is a typical wireless communication technology, which is widely used in wireless sensor network. ZigBee wireless sensor network has great advantages in terms of low power consumption, high fault tolerance, flexibility, and autonomy [7]. Using Zigbee techniques as a back bone to develop ubiquitous applications has been warming up while current information technology evolution moving from electrification to mobilization. However, most successful business cases still rely on mobile tools, such as PDA, WIFI, RFID, and GPS, to realize the concept of ubiquitous. The main challenges ahead for making real U-applications on the market are not only the definitions of Zigbee specifications and protocols, but the related optimal database build-up and interface design methods [8].

System Design

Some key design issues that were addressed are mentioned below:

- Provision of external power supply – Since the mote cannot always be connected to the PC, we need to use an external supply. We decided to use the TPS60211 voltage regulator and a 3V Panasonic CR series Lithium coin cell.
- Sensors – Each mote was required to have one sensor on board. We chose to have a temperature sensor and a motion detector (PIR sensor). Interfacing the sensors to the MSP requires the use of GPIO pins. For this purpose we need to provide a port on the PCB.
- Low power consumption – Keeping in mind the requirement for low power consumption, we had to use FET switches to control the current flow through the sensors in order to prevent them from draining the power supply when not required.
- Gateway node – One of the three motes was to be used as the gateway node which could connect to a PC to relay the information from the sensors to the user.
- Extension for serial communication – Since we need a mote to communicate to the PC while being used in a sensor network, we need to provide a serial interface. For this reason we provide an external header for the TX and RX pins along with Vcc and ground to implement serial communication using USART if needed.
- Network – The two sensor motes are required to sense and transmit this data at regular time intervals (in case of temperature sensor) or when interrupted (in case of PIR sensor).

In the design of a mote, various components were considered to optimize the performance. Texas Instrument's MSP430 microcontrollers were chosen along with Chipcon's CC2420 RF transceiver. MSP430 family of microcontrollers [9]consumes very less amounts of power and can handle both analog and digital signals. CC2420 also consumes less power and produces good output power for transmission of packet in the 2.4GHz frequency spectrum. Voltage regulator TPS 60211 [12]was used to regulate power supply and provide a constant supply to various peripherals. Sensors were used on two motes, one was the Panasonic [13] motion sensor with digital output and the other was a temperature sensor. This sensor requires a low voltage of about 3V for operation. The other one was the temperature sensor[14] by Dallas Semiconductor. This sensor communicates with the microcontroller using a 1-wire interface and can draw power from the controller itself. Three motes were designed, one of which was used as a gateway for the user to monitor the network.

The block diagram of the mote is shown in the fig. 1.

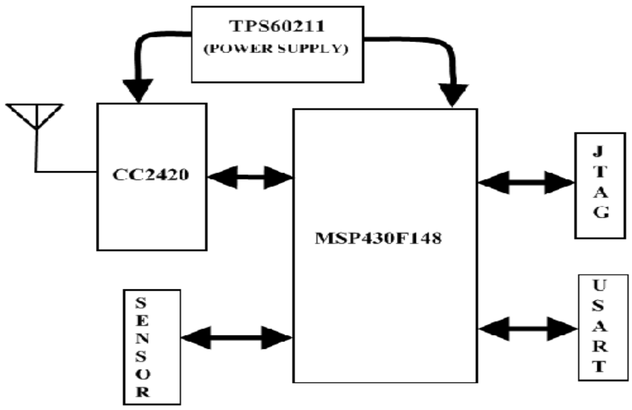

Fig. 1- Block Diagram of Mote

Implementation

A. MSP430 with RF Transceiver CC2420

The MSP430 is interfaced with the CC2420 using the Serial Peripheral Interface (SPI) [10]. SPI allows high-speed synchronous data transfer between the MSP430 and CC2420. The MSP430 has two SPI ports which include a three-wire full-duplex synchronous data transfer and can operate as a master or a slave. Programmable bit rates allow the interface to operate at the user specified rate. The MSP430 microcontroller uses 4 I/O pins for the SPI Configuration Interface. These are connected as per figure 2 and communication takes place like any standard SPI. The microcontroller is interfaced to receive and transmit FIFOs using the FIFO and FIFOP status pins. It is interfaced to the CCA pin for clear channel assessment and SFD for timing information.

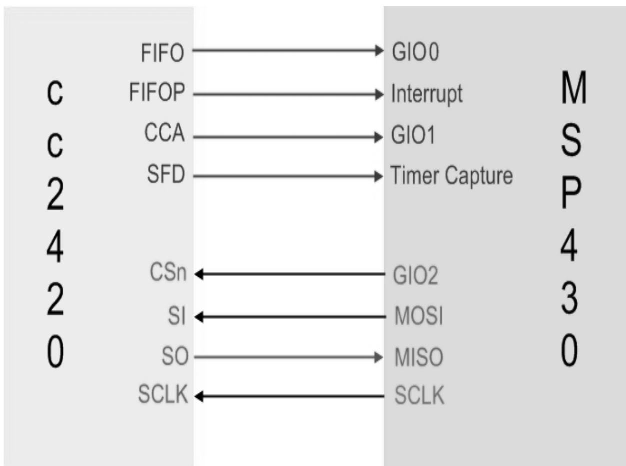

Fig. 2- Interfacing MSP430 with CC2420

B. JTAG Interface

In our design we gave direct access to the four JTAG pins and also the RST/NMI pins. Texas Instruments provides the FET430UIF flash emulation tool which uses a JTAG interface to program the flash memory and also provide debugging facility. This module connects to the PC using a USB interface. The FET430UIF requires a 14-pin connector and for this purpose we had to create an intermediate connector of our own.

C. Master Routine of MSP430

We have developed a Master routine that the microcontroller runs when the mote is powered on. As the main controller of the whole system, the MSP430's major responsibilities are initializing the system, receiving and executing the orders and memorizing these. The system initialization includes initializing the real-time clock, resetting the watchdog timer, initializing the serial ports and interrupt flags. The flow of the main programs is illustrated in figure 4. [15]

The operation of the main program of processor can be divided into five parts: (1) Setup the system, including initializing the clock, RTC, Serial Port; setup the ZigBee module and switch off. (2) Processor goes into low-power-consumption mode and waits for the switch-off from the serial port. (3) The data input at the serial port will interrogate its breaking off and wake up the processor to resume normal working; it can also identify and operate the data at the serial port. (4) Decide whether the data received at the serial port is useful. If not, the processor shall return the low-power-consumption mode and keep on waiting for the serial port data; if useful, the processor shall decode and identify them and decide the content of the order. (5) As per the content of the order, by controlling the peripheral equipment, the processor sets up the time, measures the parameters at a certain time. After the operation, the processor returns to the low power-consumption mode and waits for the data from the serial port [16]. We have implemented several commands, such as Set Time, Collect parameters and Upload Data. The Texas Instruments Z-stack was used to implement the transmission/reception functionality [17].

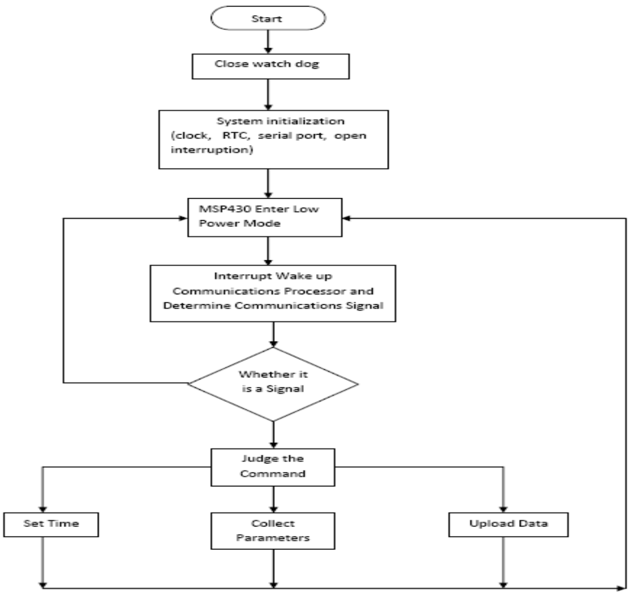

Fig. 3- Master Routine of MSP430

Results and Discussion

We tested the mote for various scenarios and measured the current consumption levels for each case. The current levels were found to comply with our requirements. The current levels measured for various scenarios are shown in table2.

Table2- Current levels for different cases

Measurement scenario	Measured Current
MSP430 in standby mode	6.5μA
MSP in an infinite loop	1.8mA
MSP + CC2420 in transmit mode	19.3mA
MSP + CC2420 in receive mode	22.4mA

Figure 4 shows the size comparison of our designed mote with 2 rupee coin.

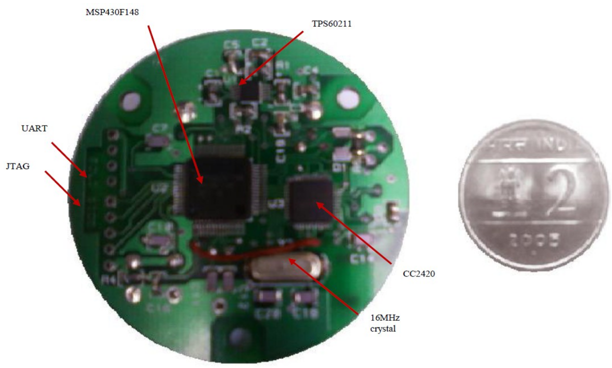

Fig. 4- Comparison of designed mote with two Rupee coin

Conclusion

In this paper we have presented the system architecture for a wireless sensor node that is capable of addressing the strict requirements of wireless sensor networks. By utilizing a single shared controller that is augmented by a collection of specialized hardware accelerators, the architecture is able to support flexible, application-specific communications protocol such as Zigbee without sacrificing efficiency. This architecture has been validated through the development of three hardware platforms and a software developing IDE. We developed the software using IAR workbench which provides the fine-grained concurrency mechanisms required to implement wireless sensor network protocols and applications. We also have presented an architecture that addresses key issues that arise when building a wireless sensor network device. They include: flexibility, low power consumption and small size. Some of the future steps that can be taken in order to minimize the size and increase efficiency of the mote are

- Explore feasibility of using a 32-bit microcontroller: Currently we are using the 16-Bit RISC Architecture, of the MSP430F148. By using a 32 bit microprocessor like the Atmel AVR32 UC3 family we can achieve faster and better performance.
- Port a tiny version of Linux: By porting a tiny version of Linux we can increase the flexibility of the mote. Other operating systems meant for such applications like TinyOS and RTOS can also be considered.
- Increase output power: By using an efficient power amplifier circuit we can boost the output power and hence increase the range of our mote.

- On board directional antenna: By incorporating an on board directional antenna we can increase the range and also minimize the matching circuit losses.
- Integrated microcontroller-radio chip: We can greatly reduce the size of the mote incorporating an integrated microcontroller-radio chip like the C2530.

Acknowledgements
Authors would like to thank Texas Instruments for providing the required hardware and software tools to carry out the work.

References

[1] Warneke B., Last M., Liebowitz B. and Pister K.S.J. (2001) Smartdust: *Communicating with a cubic-millimeter computer.* 34(1), 44-51.
[2] Naveen M. (2007) *KMote- Design and Implementation of Wireless Sensor Networks*.
[3] Yong-Sik Choi, Young-Jun Jeon, Sang-Hyun Park (2010) 12*th International Conference on Advanced Communication Technology*, 574-579.
[4] Ben Ezovski G.M., Watkins S.E. (2007) *IEEE International Conference*, 15-22.
[5] Akyiliz I.F., Su W., Sankarasubramaniam Y. and Cayirci E. (2002) *IEEE Communication Magazine*, 102-114.
[6] Shaosheng Li, Wenze Li, Jianxun Zhu (2009) *IC-NIDC, IEEE*.
[7] Liu Yanfei, Wang Cheng, Qiao Xiaojun, Zhang Yunhe, Yu chengbo,Liu Yanfei (2009) *2nd IEEE International Conference on Computer Science and Information Technology*, 515-518.
[8] Chia-Ping Huang (2009) *Eighth International Conference on Machine Learning and Cybernetics,* 12-15.
[9] MSP430F148 datasheet- *http://focus.ti.com/docs/prod/folders/print/msp430f148.html*
[10] CC2420 datasheet - http*://focus.ti.com/docs/prod/folders/print/cc2420.html*
[11] MSP430x1xx Family User's Guide - *http://www.ti.com/litv/pdf/slau049f*
[12] TPS60211 Datasheet- *http://focus.ti.com/docs/prod/folders/print/tps60211.html*
[13] Motion sensor AMN44122 Datasheet- *http://pewa.panasonic.com/pcsd/product/sens/pdf_cat/amn.pdf*
[14] DS18B20 Datasheet- *http://datasheets.maxim-ic.com/en/ds/DS18B20.pdf*
[15] Peng Jiang, Hongbo Xia, Zhiye He and Zheming Wang (2009) *Design of a Water Environment Monitoring System based on Wireless Sensor Networks*, *Sensors* ISSN 1424-8220.
[16] Nikhil S. Kulkarni, Rakesh R., Sarthak Bhargava, Shantanu Singh Bundela, Rajeshwari Hegde (2010) *IET International Conference on Next Generation Networks*.
[17] Designing a low-cost, low-power indigenous mote using MSP430 and CC2420.(2010) *UniTI Newsletter,* 3(4).