New Design of Low Power Consumption Mote in Wireless Sensor Network

Saidi Nabiha, Khaled Zaatouri, Walid Fajraoui And Tahar Ezzeddine

***Abstract* -----**The market of Wireless Sensor Network WSN has a great potential and development opportunities. Researchers are focusing on optimization in many fields like efficient deployment and routing protocols. In this article, we will concentrate on energy efficiency for WSN because WSN nodes are habitually deployed in severe No Man's Land with batteries are not rechargeable, so reducing energy consumption represents an important challenge to extend the life of the network. We will present the design of new WSN mote based on ultra low power STM32L microcontrollers and the ZIGBEE transceiver CC2520. We will compare it to existent motes and we will conclude that our mote is promising in energy consumption.

***Keywords* ---** component; WSN mote; power consumption, STM32L, sensors and CC2520

I. Introduction

WSN is a network of many tiny low power devices, called motes or nodes. These motes are spatially distributed in areas to be supervised then it relay data to the sink node. The nodes are collaborating together in order to perform measurement of the sensed environment.

WSN is receiving significant attention due to their unlimited potential. However, many challenges exist like the deployment of nodes to cover the larger areas, security of data, fidelity and energy efficiency [1]. The power consumption challenge remains a major challenge.

II. Causes of energy consumption in WSN

The causes of energy consumption are multiple in WSN. In this paragraph, we will discuss the energy consumption in the level of the mote itself and among nodes when are employed.

Errors in the communication among nodes can increase the energy consumption. We presented some errors like collision and overhead and solutions to resolve those problems. First, in the case where two packages are transmitted at the same time, they become not exploitable and must be given up or when a node receives simultaneously two packages which were not necessarily emitted at the same time or which were emitted by two nodes out of reach one of the other, we speak about collisions in WSN.

These researches and innovation are performed in the device MOBIDOC funded by the European Union under the PASRI program administered by the ANPR.

Saidi Nabiha, Khaled Zaatouri, Walid Fajraoui And Tahar Ezzeddine
SysCom Laboratory, National Engineering School of Tunis, University of Tunis El Manar 1002, le belvedere, Tunis, Tunisia. Email id: saidinabihaetn@gmail.com, khaled.zaatouri@gmail.com, fajraoui.walid@gmail.com, taharezz @gmail.com

Second, when a node source transmits data and that the node recipient is not ready to receive, we speak about Overmitting. It occurs when the radio is listening to the channel to receive possible data.

Third, the reception and the listening of the packages of control consume energy thus reducing the effective flow. We speak about the overhead of the possible packages of control.

The energy cost of overhead, collision and overmitting is particularly high. Researchers tried to resolve these problems by efficient protocols of communication like TEEN routing protocol and different deployments of motes [2]_[3]_[4].

The data processing is also an important source of energy consumption. It depends primarily on sleep mode time of the microprocessor, its tension of operation and its technology of manufacture, the transceiver characteristics and functionalities to achieve and the operating mode because in different applications, motes operate frequently like in the bridge control to avoid any danger and in other applications motes operates seldom [5].

III. Related Works: Existing Motes

In fact, microcontrollers perform the main task in the sensor nodes i.e. processing the data and controlling the functions of the other components. Most known motes are built by MSP430, STM32F and Atmel processors (Tab1).

- Motes are based on ATMEL 8-bit microcontroller are MicaZ, Rene and Iris Motes… The ATmega128L's main features are its six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby [6]. It has low power consumption in sleep mode (5 µA).
- Motes are based on Texas Instruments MSP430 16-bit microcontroller like Eyes, TelosB, SenseNode, Zolertia Z1, Tinynode, Shimmer …etc. Indeed MSP 430 has an important EEPROM and low power consumption in run mode (5.1µA).
- Researchers have exploited the STM32F processors in WSN. In [7], Yu-Jia and al designed a mote using an STM32F103RE, and in [8], Benjamin and al used STM32F4. STM32F series are rather advanced and powerful in sleep mode (10 µA).

We present in (TABLE I) the proprieties of these microprocessors.

TABLE I
PROPERTIES MICROPROCESSORS SPECIFICATIONS (*LPMC: LOWEST POWER MODE CURRENT,*RMC: RUN MODE CURRENT)

Part Number	FLASH (KB)	EEPROM (KB)	RAM(KB)	LPMC (µA)	RMC (µA)
Atmel	128	4	64	5	17 000
MSP430	48	1MB	10	1800	5.1
STM32F	256	196	64	10	1640
STM32L1	128	4	16	0.3	177

The ST Microelectronics has designed new STM32 Ultra-Low-Power MCU with record breaking performances, STM32L. This microcontroller has very low power consumption in sleep mode compared to MSP430.

In addition, WSN motes spend the often times in sleep mode which would extend the lifetime of motes based on STM32L MCU. In run mode, STM32L has an important consumption compared to MSP430, but it consumes less than Atmel and STM32F. The STM32L supports dynamic voltage scaling to optimize its power consumption in run mode when it consumes the important current.

Standby modes are even used to achieve the lowest power consumption. STM32L has an important memory with up 128KB Flash, 16KB SRAM , 4KB and its frequency range is from 32 kHz up to 32 MHz max with 8 peripheral interface and 10 timers [9]. We will present in the next paragraph, the performance of the new WSN mote design based on the STM32L MCU and we will compare it with the mote ZOLERTIA Z1.

IV. PROPOSED ARCHITECTURE OF THE NEW MOTE

The new mote is composed essentially of a microcontroller STM32L [9] with a CC2520 transceiver [10], IEEE 802.15.4 wireless system-on-chip, EEPROM, battery and sensors with low power consumptions components (fig1).

The sensors have the role to convert analogical signals (temperature, humidity …) to digital signals readable by the microcontroller. Our mote contains temperature, humidity, pressure and accelerometer sensors. The HTS221 of ST Microelectronics is the most powerful (TABLE II) in power consumption and in precision.

TABLE II
TEMPERATURE AND HUMIDITY SENSORS SPECIFICATIONS

	SHT21	HTS221	TMP102	SILICON
ADC (bits)	12	16	12	12/14
Sleep mode supply current(µA)	0.15	0.5	0.5	60nA
measuring supply current (µA)	300	2	15	150
Manufacturer	sensirion	ST	TI	Silicon

The interfaces (UART, SPI, I2C and USB) give the possibilities to connect the mote to another host like PC, other sensors or other machines. Buttons and LEDs give us an idea about the state of the mote.

The battery is the power supply for the mote. We improved the control of the power supply by a regulator of voltage because there are different voltages to supply the different components.

The Flash memory is a region for non-volatile program and data. M25P16 has the largest memory (16MB) and the lowest energy consumption. We will adapt it in our new design for low power consumption (TABLE III).

TABLE III
FLASH MEMORY SPECIFICATIONS

Flash IC	M48Z35	M25P16	bq2204A
Memory	256 Kb	16 Mb	256 Kb
Manufacturer	ST	nuumonyx	TI
Max voltage (V)	5.5	3.6	5.5
supply current (mA)	20	15	3
Power-down mode	xx	1 µA	xx

The Transceiver CC2520 is a new series of ZIGBEE transceiver. The CC2520 provides extensive hardware support for frame handling, data buffering, burst transmissions, data encryption, data authentication, clear channel assessment, Link quality indication and frame timing information, and it supports the energy detection / RSSI and Link Quality Indication (LQI).

The radio transceiver has 250 kbps data rate with a configurable frequency from to 16 MHZ.

CC2520 is typically controlled by STM32L connected to the SPI. Two principal power modes are supported by the transceiver (sleep and active mode) to optimize power consumption [10].

The AtmelRF230 transceiver [11] consumes lower power than CC2520 transceiver but it has not hardware encryption support, neither packet sniffing.

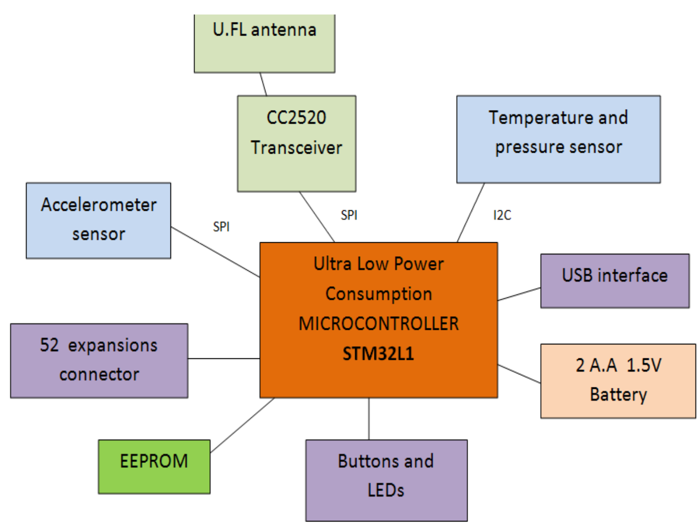

Figure 1. Block diagram of the proposed new mote (fig1)

The CC2520 transceiver offers an interrupt and address filtering system. It also should be more robust against wideband interference and coexisting systems because it has a much tighter band-pass filter, as indicated by the channel rejection data. The CC2420 transceiver, predecessor of the CC2520, has less power consumption compared to the CC2520 but it is not recommended for future designs.

TABLE IV
SPECIFICATIONS OF TRANCSEIVES

Part Number	CC2420	CC2520	AtmelRF230
Supply Voltage (V)	3.6	3.6	3.6
Supply Current (Lowest power mode) (µA)	.25	.3	0.02
Supply Current (Run mode per MHz) (µA)	87	177	15500
TX current consumption (mA at 0dbm)	17.4	25.8	16.5 (at 3 dbm)
RX current consumption (mA)	18.8	18.5	15.5
Packet sniffing	NO	YES	NO
Encryption support	AES-128	AES-128	NO
Receive sensivity (dbm)	-95	-98	-101

In the following figure we presented the schematic of CC2520 IC linked to the microprocessor. In reality, the CC2520 is the responsible for communication between motes. It receives signals via the RF inputs (FR-P and RF-N). The RF core (DAC, mixers, amplifiers of signals) supports and control the analog radio modules.

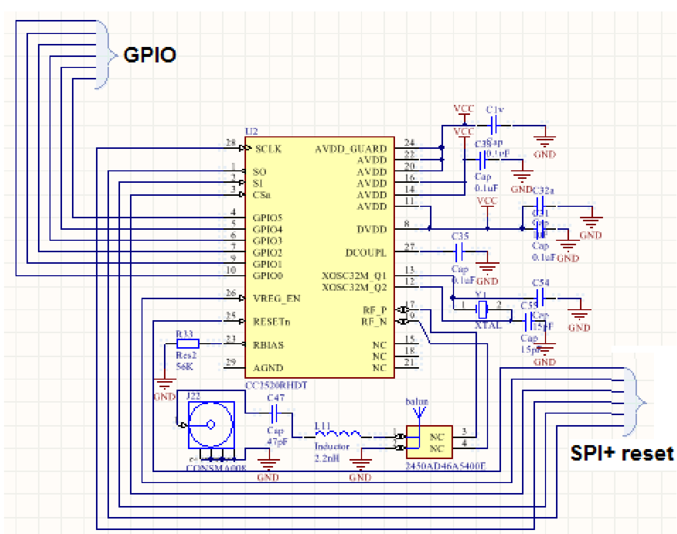

Figure 2. CC2520 electronic scematic (fig 2)

Signals are detected by the antenna. The external 50/100 Balun as an RF component provides an unbalanced to differential 50-to-100 Ohms conversion to present the STM32L with its optimal load. A single inductor makes up the matching network and optimizes RF performance.

V. PERFORMANCE AND EVALUATION OF OUR NEW MOTE

The energy consumption is challenging and divergent field of research in WSN. In this paper, we are interested to minimize the mote's power consumption. We presented in the table below a comparison between the mote ZOLERTIA Z1 and our mote.

Our mote has an important RAM, Flash and EEPROM memory.

The current consumption of the microcontroller of our mote is close to the Z1 microcontroller current consumption but there is a big difference in current consumption in TX mode.

TABLE V
COMPARISON BETWEEN ZOLERTIA AND OUR MOTE

	ZOLERTIA Z1	Our mote	CM5000
EEPROM (Kb)	2048	4096	xx
Flash	16 Mb	128 Kb	1024 Kb
RAM (Kb)	8	32	xx
Supply Voltage (min) (V)	1.65	1.65	1.65
Supply Voltage (max) (V)	3.6	3.6	3.6
Supply Current (Lowest power mode) (µA)	0.25	0.3	0.2
Supply Current (Run mode per MHz) (µA)	87	177	330
TX current consumption (mA at 0dbm)	17.4	25.8	17.4
RX current consumption (mA)	18.8	18.5	18.8

In fact, in TX mode, there is greedy power consumption; it causes a life challenge for the motes in WSN. But in our mote, the TX mode has a low power consumption compared to the Z1 which extends our mote's life.

VI. DESCRIPTION OF THE SOFTWARE OF THE SMART NODE

In this paragraph we present a description of the algorithms of the mote (fig3 and fig4).

After powering-up the mote, the entire devices would be initialized. The initialization includes the initialization of clocks, I/Os, serial interfaces and the transceiver C2520, it is an initialization of the hardware, it also includes scheduling, and memory management.

Activating sensors: the host system (MCU) sends commands to the sensors to be activated. The sensors in the WSN mote may be an accelerometer sensor, a pressure and temperature senor, selected in order to be low power consumption, correct, stable and can be used in different environmental application. The host system (MCU) initializes the sensor hardware and performs the measurements in the sensor in order to measure real parameters then it waits for the digital data.

Processing data: it includes the interrupt handling, read/write to memory, read …etc. others algorithms could take place in the level of the microprocessor for example choice of the best neighbor via Link Quality Indicator LQI calculation, determining the average of controlled parameters like temperature, humidity or pressure by processing data locally and minimize the number of communication via the transceiver and the encryption algorithm for the WSN security.

The MCU turns the RF field ON and transmitting the first protocol command of transmission and reception through the FIFO buffer and all protocol data processing (fig4).

The AtmelRF230 transceiver [11] consumes lower power than CC2520 transceiver but it has not hardware encryption support, neither packet sniffing. The CC2520 transceiver offers an interrupt and address filtering system. It also should be more robust against wideband interference and coexisting systems because it has a much tighter band-pass filter, as indicated by the channel rejection data. The CC2420 transceiver, predecessor of the CC2520, has less power consumption compared to the CC2520 but it is not recommended for future designs.

The transceiver filters if the frame filtering bit is enabled (which is the default setting). The CC2520 will only accept frames that fulfill all of the requirements (length byte, destination and frame type) for more details please see [10].

The reception of frames starts with start of frame (SFD) detection, followed by the byte length to let the transceiver when the reception is achieved. Then an automatic FCS checking of frames takes place and it is indicated by the CRC_OK bit if the FCS is correct.

After successful frame reception, an automatic acknowledgment (ACK) transmission with the correct values (like timing based on the results from source address matching and FCS checking) is sent to the sender via the CC2520 hardware for acknowledgment transmission.

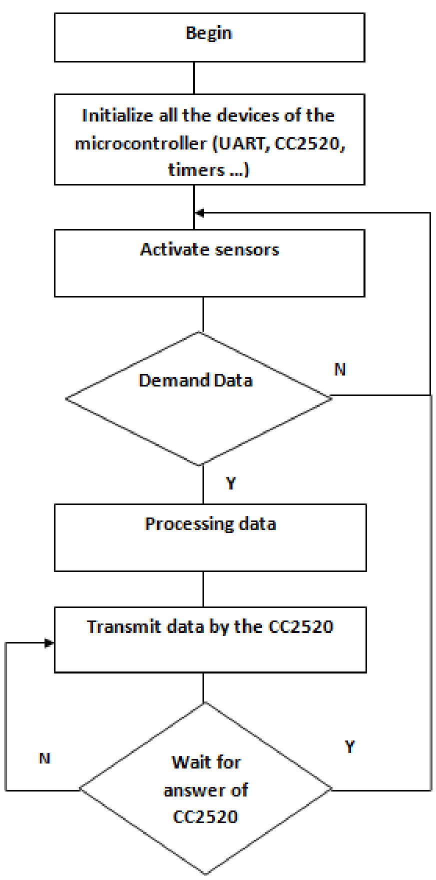

Figure 3. The program flowchart of the microcontroller (fig 3)

We can resume these algorithms and its relations with different layers of ZIGBEE software like Contiki or TinyOS. In the figure below (fig5), we included, hardware layer, where we found microprocessors, sensors and the transceiver.

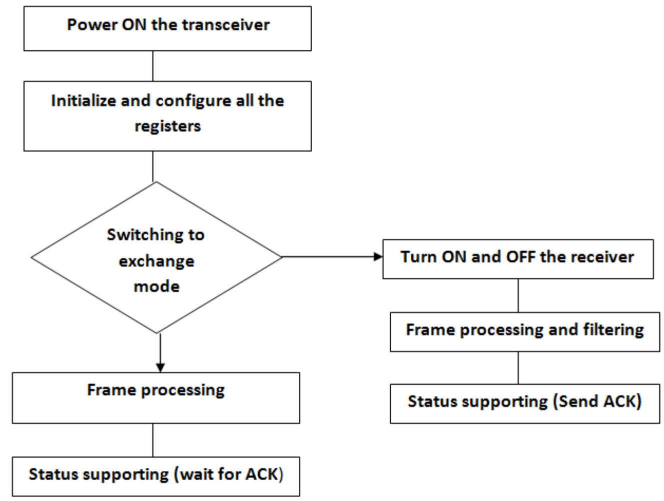

Figure 4. The program flowchart of the CC2520 (fig4)

Then the hardware management composed of APIs and drivers which take on a special role in the Linux kernel. Rubini and Corben call drivers the “black boxes” that make a particular piece of hardware respond to a well-defined internal programming interface [15]. It includes sensor driver, memory driver…etc.

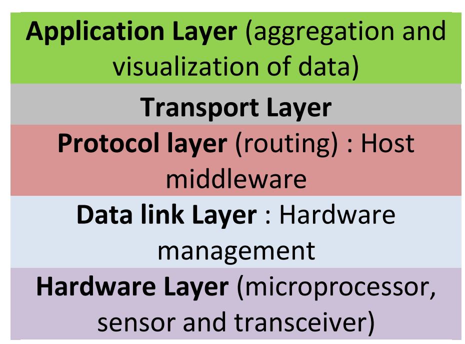

Figure 5. The layers of ZIGBEE software (fig5)

The middleware management includes algorithms of encryption engine, multiplexing, MAC, system timers and radio transmission process to provide services like collection environment data and packaging of the data by the microprocessor and then transmitting them by the transceiver and virtual machine.

The protocol layer and transport layer are to provide routing and collision avoidance. The application layer is responsible for the data management.

VII. Conclusion and Perspectives

In this paper, we discussed the new architecture of WSN mote in order to enhance the mote’s performance and lifetime. The new design is based on low power consumption microcontroller STM32L and the ZIGBEE transceiver CC2520 and low power consumption sensors and memory.

We presented a comparison between the components of mote, microprocessors, transceiver memory and sensors. We were concentrating on low power consumption components in order to let our mote to have good performance compared with other motes.

REFERENCES

[1] Intelligent optimization of wireless sensor networks through bio-inspired computing: survey and future directions. Jabbar, S.; Iram, R.; Minhas, A.A.; Shafi, I.; Khalid, S.; Ahmad, M , Int. J. Distr. Sens. Netw.2013, 1–13.

[2] Efficient deployment of wireless sensor networks targeting environment monitoring applications, FM Al-Turjman, HS Hassanein, MA Ibnkahla - Computer Communications, 2013.

[3] TEEN: A Routing Protocol for Enhanced Efficiency in Wireless Sensor Networks, Arati Manjeshwar and Dharma P. Agrawal , in Proc. IPDPS 2001 Workshops, 2001.

[4] Protocoles auto-adaptatifs énergie-trafic pour les réseaux de capteurs sans fil. Ye-Qiong Song. Ph.Roose et N. Couture. 8èmes journées francophones Mobilité et Ubiquité (Ubimob), Anglet, France. Cépaduès Editions (ISBN: 978.2.36493.018.6), Jun 2012.

[5] Energy-Efficient Communication Protocol for Wireless Microsensor Networks. Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan. In *Proceedings of the 33rd Hawaii International Conference on System Sciences-Volume 8 - Volume 8* (HICSS '00), Vol. 8. IEEE Computer Society, Washington, DC, USA, 8020, 2000.

[6] MicaZ wireless measurement systems, http://www.openautomation.net/uploadsproductos/micaz_datasheet.pdf

[7] Design of a Wireless Sensor Network node based on STM32 , UN Yu-jia, WANG Xiao-ming,JIA Fang-xiu, YU Ji-yan , Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering (ICCSEE 2013).

[8] The datasheet of the CC2520 IC, http://www.ti.com/lit/ds/symlink/cc2520.pdf.

[9] Datasheet of the ultra low power microprocessor STM32L15 http://www.st.com/web/en/resource/technical/document/datasheet/CD00277537.pdf.

[10] The datasheet of the zigBee transceiver CC2520 IC http://www.ti.com/lit/ds/symlink/cc2520.pdf.

[11] At86rf230 low power 2.4 ghz transceiver for zigbee, ieee 802.15.4, 6lowpan, rf4ce and ism applications, http://www.atmel.com/dyn/resources/prod_documents/doc5131.pdf.

[12] Benjamin, A Contiki port for my custom cc2520+stm32f4-boards, http://vedder.se/2013/04/cc2520-and-stm32-rf-boards/

[13] List of WSN nodes , http://en.wikipedia.org/wiki/List_of_wireless_sensor_nodes.

[14] Vini Madan and SRN Reddy, Review of Wireless Sensor Mote Platforms VSRD - IJEECE, Vol. 2 (2), 2012, 50-55.

[15] Linux device drivers 2nd Edition, Alessandro Rubini, Junathan Corben , O'Reilly Media, ISBN 10: 0-596-00008-1, (june 2011).