A WIRELESS SENSOR NODE FOR RIVER MONITORING USING MSP430® AND ENERGY HARVESTING

Leander B. Hörmann, Philipp M. Glatz, Christian Steger, and Reinhold Weiss

Institute for Technical Informatics, Graz University of Technology, Austria
email: {leander.hoermann, philipp.glatz, steger, rweiss}@tugraz.at
web: www.iti.tugraz.at

ABSTRACT

A microcontroller is the most important component of a wireless sensor node, also called mote. It has to perform the measurements using the integrated peripherals, to post process the measured data as well as to coordinate the data transport of the wireless sensor network (WSN). The lowest possible power consumption is also a key requirement. The MSP430™ microcontroller family from Texas Instruments® enables a high processing power at low power consumption. This work demonstrates the flexibility of the microcontroller that is needed to adapt a mote to a specific application area. In this case, the application area is river monitoring. One requirement is a perpetual operation using energy harvesting. This work presents the first water level monitoring platform only using ultracapacitors as energy storage elements to compensate the irregularity of environmental energy sources. It shows how perpetual operation can be enabled by versatile high-performance and low-power modes of the MSP430.

1. INTRODUCTION

The requirements on a WSN diversify very much, because they are used in many different application areas, e.g. precision agriculture, environmental monitoring, structural health monitoring, industrial applications [10]. A WSN typically consists of a high number of sensor nodes, also called motes, and of one or few gateway nodes. The gateway node is connected with a computer to access the data of the WSN. Each mote collects information about its environment using the attached sensors. This information is forwarded to gateway nodes in a multi-hop manner as shown in Figure 1. Usually, it is necessary to preprocess the measured data before it can be transmitted. A microcontroller can be used to perform this task. Such a microcontroller should also provide a wide range of peripherals that can be used for measurement or conversion tasks, e.g. analog-to-digital converter (ADC), timer and digital IO. This reduces the amount of needed components and with it the hardware costs.

A key requirement on motes is very low power consumption. Typically, the motes are powered by batteries, energy harvesting systems (EHSs) or both. Batteries have to be replaced after a certain time. Hence, the lifespan of a battery powered mote is limited. EHSs uses energy harvesting devices (EHDs), e.g. solar cells or thermogenerators, to harvest energy from the environment. If the EHS is well designed, it is possible to power the mote perpetually. However, both solutions need a low power consumption of the mote.

The needed processing power and the maximum allowed power consumption lead to a tradeoff between these requirements. However, the specific requirements depend on the application area. The mote should be adapted to that application area. Thus, this tradeoff can be optimized and the needed performance can be achieved by a minimum power consumption.

The MSP430™ ultra-low power microcontroller from Texas Instruments® supports all these requirements. There are many different models with different peripherals. It also supports different power modes which can be used to minimize the overall power consumption. This paper shows the development of a mote for river monitoring using an MSP430 microcontroller. The mote is supplied by solar cells and ultracapacitors are used to store the energy for the night. The hardware is adapted to the given application area in order to get the best system performance and to reduce costs.

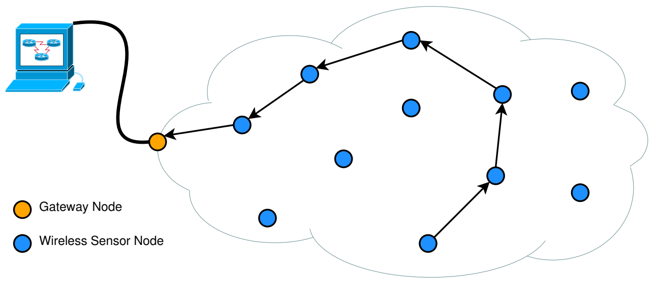

Figure 1: Typical structure of a wireless sensor network.

The paper is organized as follows. Section 2 explains the basic structure of a mote. It shows a list of typical microcontrollers that are used for motes and a short survey on energy harvesting. Section 3 explains the concept of a low power design. Section 4 shows details of the implementation and results of the measurement. Finally, Section 5 concludes the paper and outlines future work.

2. RELATED WORK

Typically, a mote of a WSN is composed of four main units which are connected to each other [7]. Figure 2 shows the interaction of these units and the following list describes the tasks of it.

Measurement Unit: It consists of the sensors to measure the relevant physical values (e.g. temperature, light, humidity, noise, etc.).

Processing Unit: The microcontroller is the main element of this unit. It gathers the measurement values using digital IO or ADCs and it processes and stores the data. It is used to coordinate the communication in the WSN.

Communication Unit: It provides connectivity to other motes and consists of an wireless transceiver and an antenna.

Power Unit: The power unit supplies the other units with energy. To enable a continuous mode of operation, it is necessary to store energy, because *energy from the environment is generally unpredictable, discontinuous, and unstable* [2].

2.1 Microcontrollers

The key element of each mote is its microcontroller. Therefore, it is necessary to select the most suitable one. Table 1 shows a list of state-of-the-art microcontrollers which are typically used for wireless sensor network hardware. It can be seen that the ARM9 microcontroller has a very high processing power and a lot of peripherals but also the highest power consumption. It is very difficult to integrate such a microcontroller in a power-aware application. However, an ARM9 core microcontroller is used for the Java™ Sun SPOT [9]. Java needs the processing power to adapt the software to a wide range of application areas.

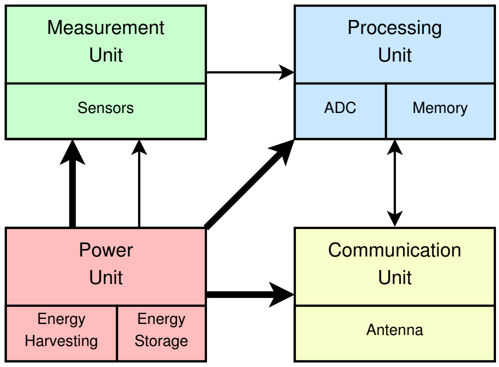

Figure 2: Generic structure of a mote adapted from [7]. Thick lines represent the energy flow and thin lines the information flow.

Micro-controller	MSP430 F1611	ATmega 128L	ARM9 core AT91SAM9R64
Architecture	16 bit	8 bit	32-bit
Supply [V]	1.8-3.6	2.7 - 5.5	1.2 1.8 and 3.3
Stand-by [mA]	0.0011	0.015	0.175
Active [mA]	1.32 (2.2V, 4MHz)	5 (3V, 4MHz)	47 (220MHz)
ADC	12 bit	10 bit	10 bit
USARTs	2	2	4
Wake-up* [μs]	6	-	-
RAM [kByte]	10	4	64
Flash [kByte]	48	128	-

Table 1: Comparison of microcontrollers. *Wake-up time from stand-by mode after an interrupt.

The MSP430 microcontroller has a better performance and lower power consumption than the Atmega128L microcontroller. Only the on-board flash memory is smaller. The Atmega128L microcontroller is used for the Mica2 and MicaZ motes from Crossbow Technology® [10]. The MSP430F1611 is integrated in the TelosB mote [6]. In [1] a WSN platform for monitoring coral reefs is introduced. They use an MSP430 microcontroller, too. The hardware is placed into a watertight housing and it is placed at the ocean surface level to allow communication. They also mentioned a low power hardware design using P-channel MOSFETs to power down the sensors during sleep state.

2.2 Energy Harvesting

In [3] an environmentally powered sensor network is introduced which is powered by solar cells. It combines ultracapacitors and a Lithium Ion battery to exploit the positive attributes of both energy storage elements. They predict a system operation time of 4 years at 10% load.

There are mainly two possibilities to store the electrical energy. First, the energy can be stored in a rechargeable battery and second, it can be stored in an ultracapacitor. Batteries have much more capacity compared to the ultracapacitors. The disadvantages of the batteries are reduced lifetime and temperature dependencies. An ultracapacitor has typically 500000 full charge-discharge cycles [4]. A rechargeable battery has a much shorter cycle lifetime. For example, a NiMH battery has only a lifetime of 500 full charge-discharge cycles [8].

3. LOW POWER DESIGN USING MSP430 MICROCONTROLLER

The design of a mote is often driven by the tradeoff between low power consumption and high processing power. The key challenge is to select a microcontroller which meets the given requirements. The first step is to analyze the application area of the WSN. Then, the sensors can be selected to measure the physical quantities which have to be monitored. The power consumption, the signal type and the sampling rate of the sensors lead to first microcontroller requirements. Thereafter, the most suitable microcontroller can be selected.

As mentioned in the introduction, the application area of the developed mote is river monitoring, especially the water level of the river. There are many different sensors, which can be used to measure the river level, e.g. floaters, pressure transducers, radar and ultrasonic devices. Three different measurement principles have been envisioned:

Pressure measurement: A pressure transducer at the river bed measures the hydrostatic head.

Ultrasonic measurement: An ultrasonic wave is transmitted from the surface of the river and the reflected ultrasonic wave can be detected again at the surface. The time in between corresponds to the water depth.

GPS heighting: GPS receivers can also be used to measure the water level of a river. One receiver is used as a reference station at a fixed point and the other receiver is placed at the surface of the river. The height difference can be used to determine the level of the river if the position of the reference station is known. [5]

The novel integration of three different sensors for water level measurements ensures a high reliability and accuracy of the system. A disadvantage of the use of three different sensors is a higher power consumption than using only one sensor. However, with the right techniques it is possible to reduce the average power consumption to a minimum.

3.1 Sensor Power Estimation

One technique to reduce the power consumption is to activate the sensors only during the measurement process. Therefore, energy can be conserved and the average power consumption of the mote decreases. The measurement interval is set to $T_{intervall} = 15$ *min*. Equation 1 shows the calculation of the duty cycle D of each sensor.

$$D_i = \frac{t_{a,i}}{T_{intervall}} \tag{1}$$

The average power consumption of a sensor can be calculated according to Equation 2.

$$P_{avg,i} = P_i \cdot D_i \tag{2}$$

Note that Equation 2 is only correct if all sensors are completely powered off during the sleep state. This saves a lot of energy, because the power consumption is nearly zero. However, it must be supported by the hardware and the software. Therefore, it can be seen as a functional requirement. Table 2 shows the power estimation of the three sensors. It can be seen that the average power consumption is high for a mote, which should be powered environmentally. Therefore, the microcontroller should be as efficient as

Sensor	t_a [s]	D	P_a [mW]	P_{avg} [mW]
GPS	30	0.033	120	4.0
Pressure	1	0.001	250	0.28
Ultrasonic	1	0.001	10	0.01
$P_{tot,sensors}$				4.29

Table 2: Power estimation of the different sensors. An activation interval of 15 *min* is assumed.

possible. It should also support low power sleep states to ensure minimum power consumption between the measurement intervals.

3.2 Microcontroller Selection and System Overview

The MSP430F1611 microcontroller has been chosen due to the following advantages:

- The standard power consumption of the microcontroller is very low (4.36 mW @ 6 MHz and 2.2 V supply voltage).
- It supports several low power states, which can be used to safe energy (e.g. low power mode 3: 2.86 μW @ 2.2 V, 32kHz timer active). The wake-up process triggered by an interrupt is very fast with less than 6 μs.
- It has a lot of integrated peripherals. They are needed to ensure an easy integration of the sensors into the mote.
- The 16 bit architecture of the microcontroller enables a high performance processing of the measured values.
- It has also a lot of flash memory and RAM on board (48 KB and 10 KB respectively). Therefore, no external memory is needed for a standard application.

Figure 3 shows the system overview of the mote. It can be seen that the GPS receiver is not connected directly to the MSP430-F1611 microcontroller, because both serial communication interfaces, also called USARTs (universal synchronous/asynchronous receiver/transmitter), are already occupied. However, it is no problem to integrate the GPS receiver using an SPI compatible UART (universal asynchronous receiver/transmitter). The UART is connected to the SPI bus of the mote, which is controlled by one USART of the MSP430F1611. Also the wireless transceiver module is connected to the SPI bus. The ultrasonic module is connected directly to the microcontroller. This enables the control (power on/off, ultrasonic wave generation) and the measurement (reflected ultrasonic waves). The USB interface is used to connect the mote to a PC. It is designed to communicate with the microcontroller and also to program it via a serial interface. This is possible, because the MSP430F1611 provides a bootstrap loader (BSL). The energy harvesting module provides signals which have to be measured from the microcontroller. A short overview of the energy harvesting module is given in following section.

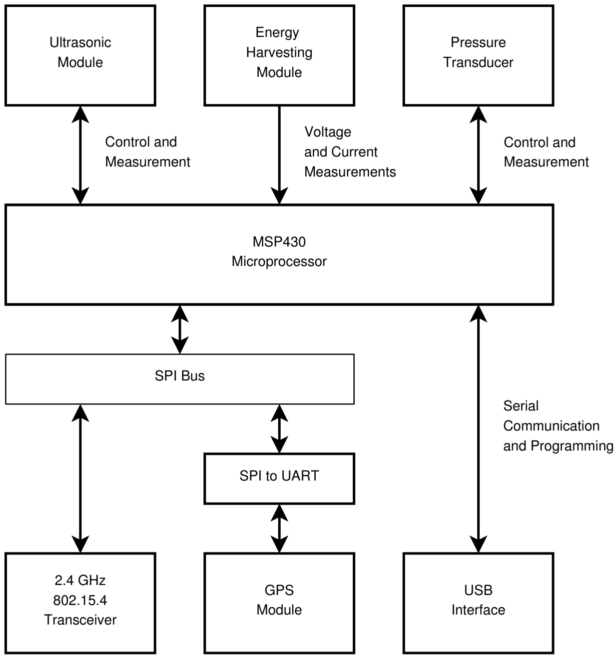

Figure 3: Mote system design.

3.3 Energy Harvesting and Storage Concept

The EHS is used to power the mote from an environmental energy source. It consists of an EHD, an energy storage element and a control and measurement circuit. Only a very short overview is provided here, because a detailed explanation would go beyond the scope of this paper. A solar cell is used to convert solar radiation energy into electrical energy. This energy is stored in two ultracapacitors from Maxwell Technologies, which are connected in series. Previous advances in ultracapacitor technologies enables high capacity cells with more than 300F (about C size battery body). This high capacity and the very long lifetime makes it suitable for this application.

Connecting two ore more capacitors in series, the overall amount of energy stored can be enhanced. The amount of energy stored in a single capacitor can be calculated according to Equation 3.

$$E_{C,single} = \frac{1}{2} \cdot C \cdot U^2 \tag{3}$$

Connecting two or more capacitors in series doubles the nominal voltage, but halves the capacity. The resulting formula is shown in Equation 4.

$$E_{C,series} = \frac{1}{2} \cdot \frac{C}{n} \cdot (U_C \cdot n)^2 = \frac{1}{2} \cdot C \cdot n \cdot U_C^2 \tag{4}$$

C is the capacity of a single capacitor, n is the number of capacitors connected in series and U_C is the nominal voltage of a single capacitor. The reduced form of Equation 4 shows the direct influence of the number of capacitors connected in series on the amount of storable energy.

A disadvantage of connecting capacitors in series is the need of a balancing circuit. This is needed, because the individual voltages of the capacitors have to be balanced. Otherwise they can be damaged due to overvoltage. The unbalancing is caused by small differences of the capacity and unequal leakage currents.

A voltage conditioning unit is used to provide a constant voltage supply of the mote. The input current, the input voltage, the output current and the capacitor voltage can be measured with special circuits. The measurement lines are directly connected to the MSP430F1611 microcontroller. Figure 4 shows the concept of the EHS.

3.4 Low Power Measurement Module Integration

It is very important to consume as low power as possible during the sleep state. Therefore, the integration of the measurement modules should be designed to be able to switch off the complete module if it is not needed. Figure 5 shows that integration. The MSP430F1611 microcontroller is connected with a P-channel enhancement MOSFET $T1$. The enable line to the MOSFET is inverted. This means the measurement module will be powered, if the enable line is low. After a reset of the microcontroller, the IO pins of it are configured as input with a pull-up resistor. Therefore, the default level of the enable line is high and the measurement module is not supplied and thus power can be saved during the initialization.

4. WATER LEVEL MEASUREMENT - IMPLEMENTATION AND RESULTS

This section describes the GPS and the ultrasonic measurement principle. The pressure transducer was not implemented. However, an interface for this sensor is implemented and it can be connected easily. Finally, the perpetual operation is demonstrated.

4.1 GPS Heighting Results

The GPS heighting is based on measuring the height difference using two GPS receivers. One receiver is placed at a fix position (e.g. river bank) and the other receiver is placed at the surface of the river (e.g. buoy). The first implementation uses cheap GPS receiver with an integrated patch antenna. It is very small and can be integrated into a buoy easily. The receiver uses a serial interface to

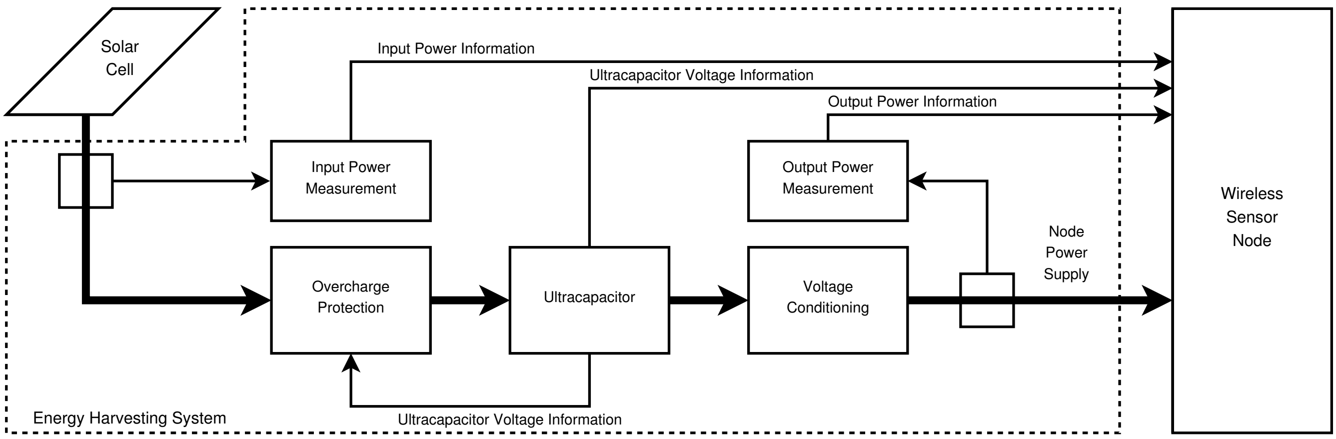

Figure 4: Energy harvesting system design. Thick lines represent the energy flow and thin lines the information flow.

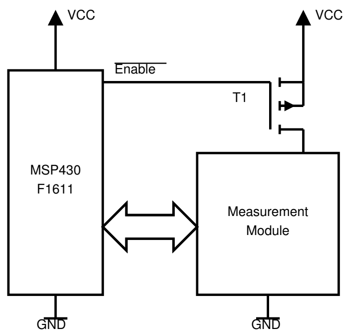

Figure 5: Low power integration of a measurement module.

transmit the position information to the SPI compatible UART. The UART generates an interrupt after receiving a byte and the microcontroller reads it from the UART. The protocol is called NMEA and it is text based. One line represents a sentence and it looks like the following: The main interesting information is the height

```
$GPGGA,HHMMSS.ss,BBBB.BBBB,b,LLLLL.LLLL,
   l,Q,NN,D.D,H.H,h,G.G,g,A.A,RRRR*PP
```

Listing 1: GGA sentence of the NMEA protocol.

of the position (`H.H`). This information has to be extracted from the microcontroller. It parses the sentence and converts the text characters into numbers. Figure 6 shows a sample trace of two GPS receivers (red and blue). They are at the same level at the beginning and at the end of the measurement. In the middle, the height of the red receiver is reduced by about 10 meters. The simultaneous jump of both receivers indicates a correction data update of both GPS receivers. Unfortunately, it can be seen that the accuracy of the measurement is not sufficient to measure the water level of a river. However, it shows the possibility of the integration of GPS receiver into an MSP430 based application.

4.2 Ultrasonic Measurement Results

The ultrasonic measurement is based on the time-of-arrival (TOA) principle. An ultrasonic transceiver is used to emit an ultrasonic wave into the water at the surface of the river. This wave is reflected at the bed of the river. The reflected wave can be detected at the surface of the river. The time in between is proportional to the depth of the water.

The ultrasonic measurement module consists of a transmitter

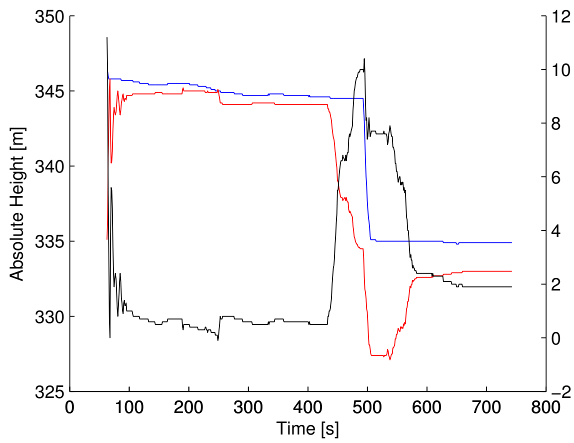

Figure 6: Differential GPS measurement with uncoupled receivers. The blue and the red trace show the absolute height of the receivers. The black one shows the difference.

circuit that is controlled by the MSP430. The receiver circuit consists of an amplifier and a low pass filter. The first idea was a simple threshold detector using a comparator. After reaching a specified value, the microcontroller can determine the time and calculate the water level.

Figure 7 shows the measured voltage of the ultrasonic measurement module. The blue trace shows the output of the low pass filter. There is an initial jump of the output voltage at time t1. This is caused by electromagnetic interferences during the generation of the ultrasonic waves. Therefore, it is not easy to specify a threshold for the comparator that is correct all the time. The solution is a continuous sampling of the trace with the integrated ADC of the MSP430. The magenta trace indicates active conversions of the ADC. A general IO pin of the MSP430 is used as debug output for a better visualization only. Figure 8 shows the converted voltage of the ADC. The ADC is configured in continuous mode. Every 18 μs a conversion is completed. The trace shows 100 samples. The converted values are post processed. A moving average is used to smooth the trace. This curve can be analyzed by the microcontroller. The implemented solution is a simple relative threshold. The reference point of the threshold is the lowest detected ADC value before the current sample. The reflected wave is detected if the difference of current sample and the minimum value is higher than this threshold. The time can be easily calculated by counting the completed conversions of the ADC. The conversion can be stopped after the detection. Thus, valuable energy can be saved. However, the fig-

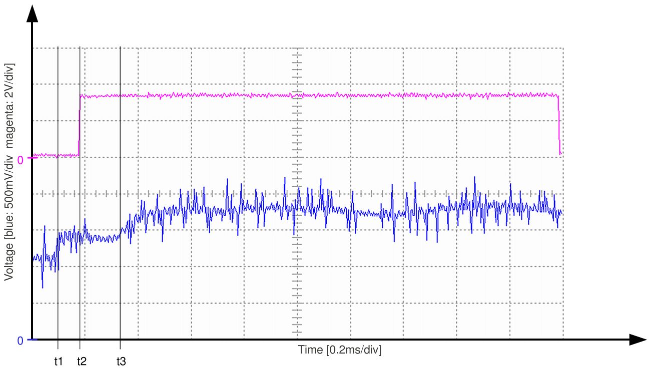

Figure 7: Output of the ultrasonic measurement module (blue trace). The magenta trace indicates active conversion of the ADC.

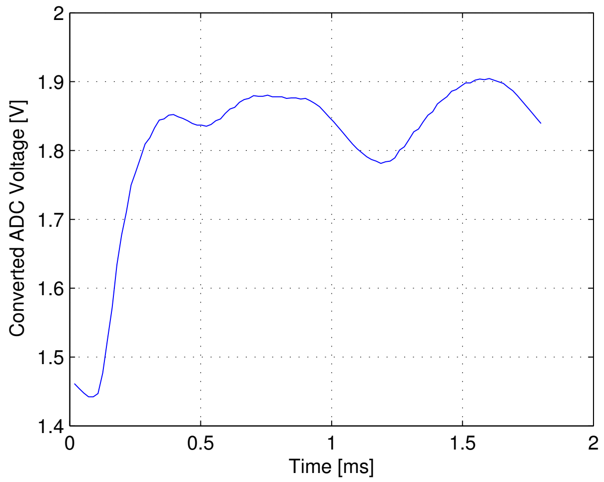

Figure 8: Converted voltage of the ADC.

ure shows the complete trace to analyze the detection algorithm. A comparison of Figure 7 and 8 shows the correct functionality. Time t2 at Figure 7 corresponds to time $t = 0$ at Figure 8. The effect of the reflected wave can be seen arround time t3 which is equivalent with $t = 0.2\ ms$ at Figure 8.

The ultrasonic measurement demonstrates the flexibility of the MSP430 microcontroller to solve emerging problems. It also shows the ability of the microcontroller to measure and process a high amount of data.

4.3 Energy Harvesting and Perpetual Operation

Figure 9 shows the continuous operation of the system for 5 days. The green trace represents the voltage of the ultracapacitors. The weather conditions at the first 2 days were very bad (rainy and foggy day in December). However, the minimum voltage does not drop below $2.8\ V$. This ensures the ability of perpetual operation.

5. CONCLUSION AND FUTURE WORK

This work demonstrated the adaptability of a mote to a specific application area using an MSP430 microcontroller. It showed low power design techniques. It also demonstrates the possibilty to supply the mote continuously using ultracapacitors as energy storage elements. The river level measurement is implemented using two different measurement principles. Future activities will target an enhancement of the GPS and ultrasonic measurement and an integration of the pressure measurement.

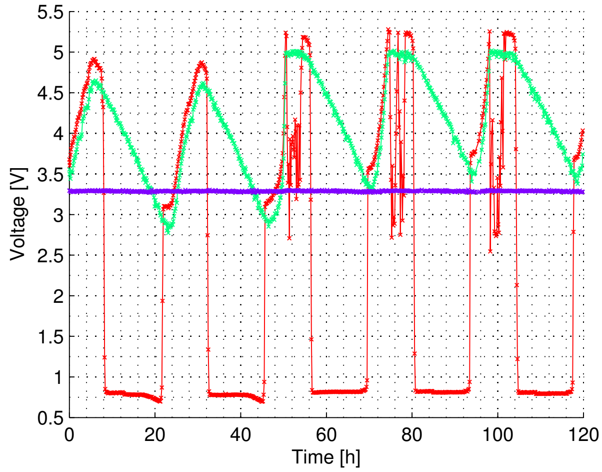

Figure 9: Perpetual operation of the mote. The red trace shows the solar voltage, the purple trace shows the constant supply voltage of the mote and the green trace shows the voltage of the ultracapacitor.

REFERENCES

[1] M. Bromage, K. Obraczka, and D. Potts. Sea-labs: a wireless sensor network for sustained monitoring of coral reefs. In *NETWORKING'07: Proceedings of the 6th international IFIP-TC6 conference on Ad Hoc and sensor networks, wireless networks, next generation internet*, pages 1132–1135, Berlin, Heidelberg, 2007. Springer-Verlag.

[2] A. Janek, C. Trummer, C. Steger, R. Weiss, J. Preishuber-Pfluegl, and M. Pistauer. Simulation based verification of energy storage architectures for higher class tags supported by energy harvesting devices. *Microprocessors and Microsystems*, 32(5-6):330 – 339, 2008. Dependability and Testing of Modern Digital Systems.

[3] X. Jiang, J. Polastre, and D. Culler. Perpetual environmentally powered sensor networks. In *IPSN '05: Proceedings of the 4th international symposium on Information processing in sensor networks*, page 65. IEEE Press, 2005.

[4] MaxwellTechnologies. Datasheet bcap0310 p250. http://www.maxwell.com, Document number: 1009473.5, November 2009.

[5] T. Moore, K. Zhang, G. Close, and R. Moore. Real-time river level monitoring using gps heighting. *GPS Solutions*, 4:63–67, 2000.

[6] J. Polastre, R. Szewczyk, and D. Culler. Telos: enabling ultra-low power wireless research. In *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*, pages 364 – 369, 15 2005.

[7] D. Puccinelli and M. Haenggi. Wireless sensor networks: applications and challenges of ubiquitous sensing. *Circuits and Systems Magazine, IEEE*, 5(3):19–31, 2005.

[8] V. Raghunathan, A. Kansal, J. Hsu, J. Friedman, and M. Srivastava. Design considerations for solar energy harvesting wireless embedded systems. In *Proceedings of the 4th international symposium on Information processing in sensor networks*, page 64. IEEE Press, 2005.

[9] D. Simon, C. Cifuentes, D. Cleal, J. Daniels, and D. White. Java on the bare metal of wireless sensor devices: the squawk java virtual machine. In *VEE '06: Proceedings of the 2nd international conference on Virtual execution environments*, pages 78–88, New York, NY, USA, 2006. ACM.

[10] J. Yick, B. Mukherjee, and D. Ghosal. Wireless sensor network survey. *Computer Networks*, 52(12):2292 – 2330, 2008.