

Low power strain gage measurement with wireless network access

Ioan Turcin¹, Manfred Pauritsch¹ and Petra Weber¹

¹ CAMPUS 02 University of Applied Science/Degree Programs in Automation technology, Graz, Austria

Abstract-This paper describes a measurement system for industry applications where the weight of containers or cable rolls has to be monitored in a flexible way. The measurement system can be used to build several kinds of weight cell applications in which the low power sensor is battery powered and able to communicate with a master unit through a wireless network. The system requirements are described and all blocks derived from these requirements are presented. The implementation of the wireless network includes a peer-to-peer connection between multiple measurement units with the master unit by using just one communication channel.

I. INTRODUCTION

The goal of this paper is to present a flexible measurement unit that can be used to make high accuracy measurements with bridge sensors like strain gages. The sensor that can be built with this unit should be flexible in terms of application and low power so that it can operate for at least six months on one battery pack. In order to apply multiple sensors, a wireless network has to be established. The administration of this network should be as basic as possible so a simple microcontroller of a PC can be used as a master unit to collect the measurement data. The measurement interval is relatively flexible since these sensors are typically used as weight cells. These weight cells are distributed and monitor the weight of production material that slowly decreases (f.ex. items in a stock) or increases (f.ex. items that are produced).

II. SYSTEM ARCHITECTURE

A. Analog front end

The typical sensor to be connected to the measurement system is a simple strain gage with four resistors or weight cells whose strain gages are distributed around the corners and consist of sixteen resistors. In order to achieve at least 10^7 load cycles a cyclic strain level of $\epsilon = \pm 1000 \mu\text{m/m}$ is assumed [1]. Applying 3V excitation voltage and assuming a bridge sensitivity of 2mV/V, the expected maximum input voltage is 6mV at maximum load.

In order to achieve a resolution of 0.01% the required analog-to-digital converter resolution is:

$$N = \log_2 \left(\frac{V_{\text{excitation}} / V_{\text{in max}}}{\text{resolution} * \text{PGA gain}} \right) = 16.2 \text{ bit} \quad (1)$$

The thermal noise and the quantization noise of the analog-to-digital converter will cause a certain rms noise which should be lower than the resolution. To prevent code flicker on the output due to rms noise the resolution has to be increased by a factor of 6.6 which corresponds to the relation of the peak to peak voltage to the rms value of Gaussian noise. So 99.99% of the outputs codes occur within $6.6 \times \text{rms}$ noise. [2] In order to get a flicker free code output the resolution of the analog-to-digital converter has to be increased by

$$N = \log_2(6.6) = 2.7 \text{ bit} \quad (2)$$

This leads to a final solution of 19bit. The equivalent input noise is $89 \text{ nV}_{\text{rms}}$. To keep the additional noise of the source resistance equal to the analog-to-digital converter a maximum bridge resistance of

$$R_{\text{max}} = \frac{V_{\text{noise}}^2}{4 * k * T * B} = 4780 \Omega \quad (3)$$

at room temperature and with a bandwidth $B = 100 \text{ Hz}$ is required.

The selected ADC is a precision 24-bit analog-to-digital converter (ADC) with an onboard, low-noise programmable gain amplifier (PGA), precision delta-sigma ADC internal oscillator and provides a complete front-end solution for bridge sensor applications including weigh scales, strain gauges and pressure sensors.

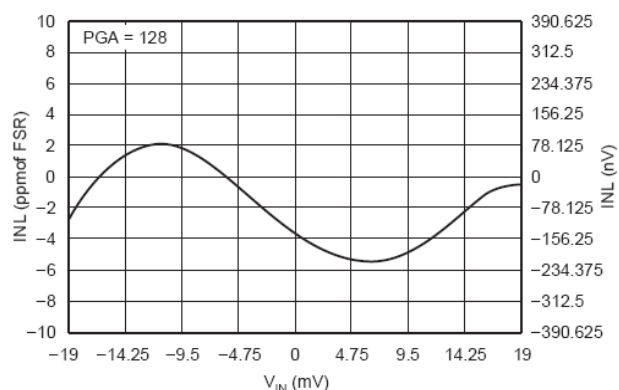


Figure 1. Integral Nonlinearity vs Input Signal

The internal programmable amplifier in the analog-to-digital converter has a gain of $PGA_{gain}=64$ supporting a full-scale differential input of $\pm 39mV$. The delta-sigma ADC has 23.5-bit effective resolution and is comprised of a 3rd-order modulator and 4th-order digital filter.

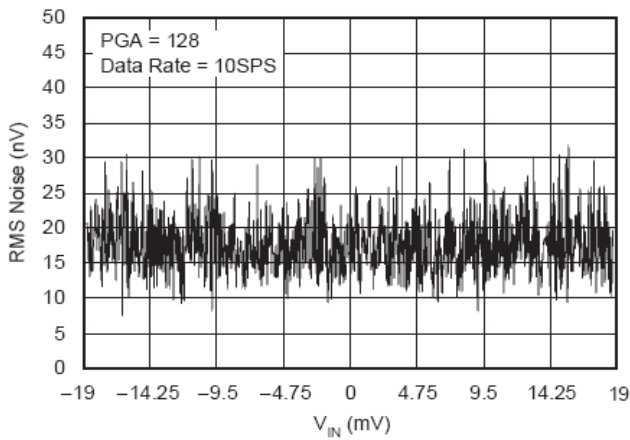


Figure 2. Noise vs Input Signal

Two data rates are supported: 10SPS (with both 50Hz and 60Hz rejection) and 80SPS. The ADC can be clocked externally using an oscillator or a crystal. There is also an internal oscillator available that requires no external components. Offset calibration is performed on-demand and the ADC can be put in a low-power standby mode or shut off completely in power-down mode.

All of the features of the ADC are operated through simple pin-driven control. There are no digital registers to program in order to simplify software development. Data are output over an easily-isolated serial interface that connects directly to the microcontroller. [3]

A more explicit block diagram of the Slave circuit is given below in Figure 4 and the Schematic diagram in Figure 3:

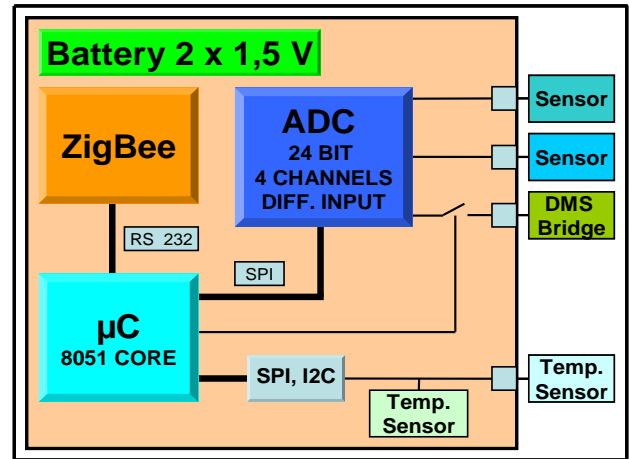


Figure 4. Slave circuit block diagram

The main parts the Slave circuit consists of are the following:

- The ADC (analog-digital converter)
- The microcontroller
- The ZigBee module
- Sensors connectivity
- Interfaces (RS232, SPI, I2C)

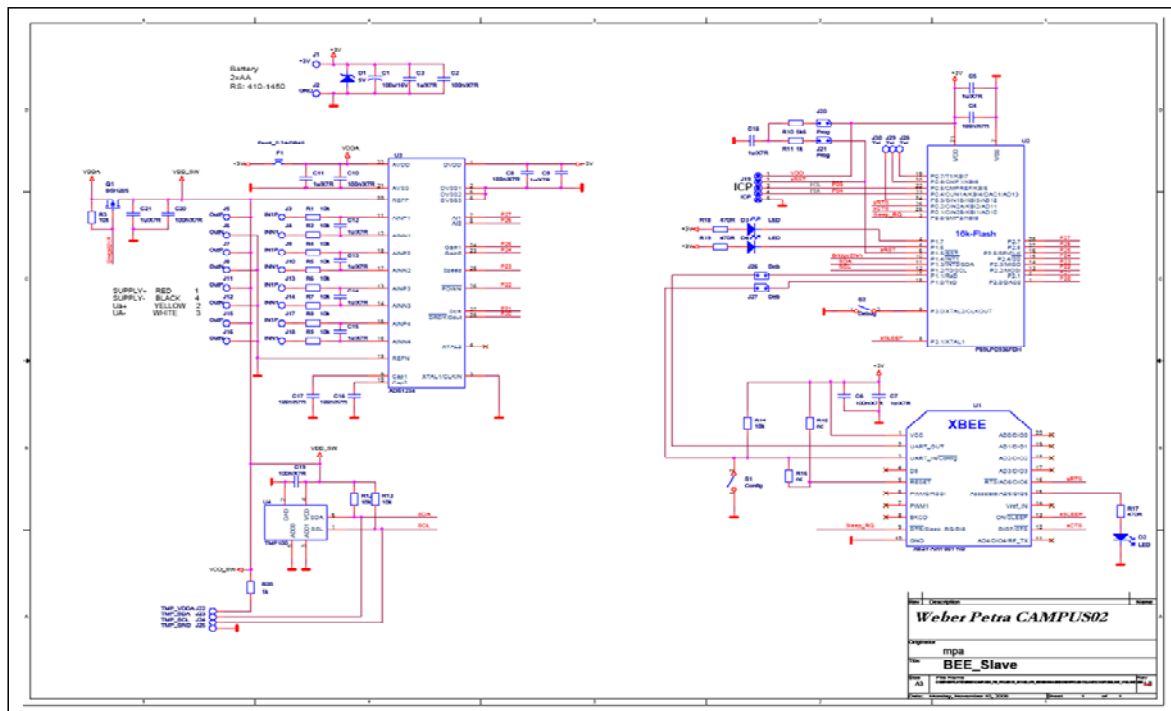


Figure 3. Schematic diagram of the Slave circuit

The Slave circuit is controlled by a simple housekeeping microcontroller. Since critical operations are constantly performed by external devices an 8 bit low speed microcontroller with simple periphery can be used. Since the controller runs on a built-in RC-oscillator frequency there is no need for an external crystal. The purpose of the controller



Figure 5. Slave circuit

is to power down all external devices and external measurement bridges and stay in sleep mode for ten minutes. After ten minutes the RF-transceiver is turned on which starts network synchronization by polling nodes.

Although RF-transceivers can switch to sleep mode, this feature cannot be used because the sleep time of the transceiver is limited to one minute. In order to enable network synchronization the transceiver module is configured via the UART at every wakeup. The configuration can be done by entering the AT command mode of the transceiver. After entering this mode the main network parameters are changed. The 802.15.4 protocol channel numbers can be set between 11 and 26 which correspond to the frequency range from 2.405GHz to 2.480GHz in the ISM band [4]. The personal area network ID is set to 5000.

To enable peer-to-peer transmission the networking addresses have to be dedicated numbers. The sensor 16 bit source address starts at 001 and increases with every sensor. The targeted master transceiver is Source address plus 100. Since the master processor could be busy receiving data from a different sensor a network scanning duration of 40 seconds has to be selected. If a connection can be established during the scanning time the microcontroller waits an additional ten seconds to receive a command byte from the master processor. After receiving this command byte a measurement cycle is started. In measurement mode the power supply for the external bridge circuits is enabled and the current consumption of the whole sensor increases to 60mA for two

seconds. During that time all analog-to-digital conversations are performed and the temperature of the module and all external temperature sensors are read via SPI interface.

B. Wireless network connection

To be able to connect to a wireless network a ZigBee module is used.

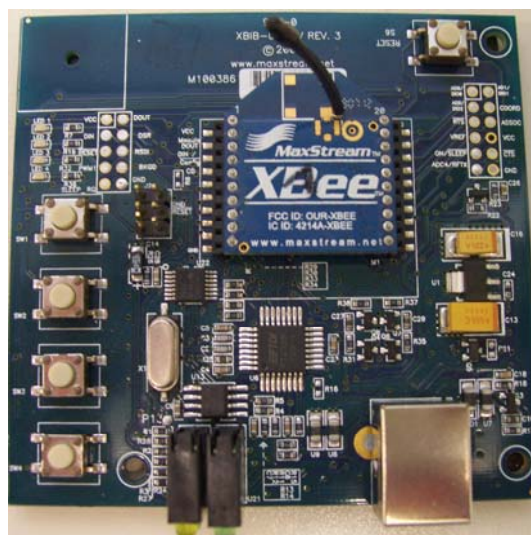


Figure 6. Master Circuit

The Master and Slave circuit used in our application contains the XBee Series 2 OEM RF Modules. MaxStream's XBee family of radios can be set up to operate in a point-to-point, point-to-multipoint or a peer-to-peer configuration. While standard 802.15.4 always requires a coordinator, the MaxStream radios are set up so that a coordinator is not required. These modules are engineered to operate within the ZigBee protocol and support the unique needs of low-cost, low-power wireless sensor networks. They require minimal power and provide reliable delivery of data between remote devices and operate within the ISM 2.4 GHz frequency band. ZigBee has been developed to meet the growing demand for capable wireless networking between numerous low power devices. In industry ZigBee is being used for next generation automated manufacturing, with small transmitters in every device on the floor, allowing for communication between devices to a central computer. This new level of communication permits finely-tuned remote monitoring and manipulation. [5]

A ZigBee module with an output power of 1mW was chosen and is a tradeoff between a reasonable current consumption of 50mA in transmitting mode and an indoor range of 30meter. Once the communication is established the transceivers can be used by the microcontroller to communicate through a TTL-level UART. The connected RF-modules operate in transparent mode and as a serial line replacement. The RF-modules work in unicast mode because this mode supports retries. In this mode the receiving module sends an acknowledgement when receiving a packet. If the

transmitter does not receive this acknowledgement it will resend the packet up to three times. [6]

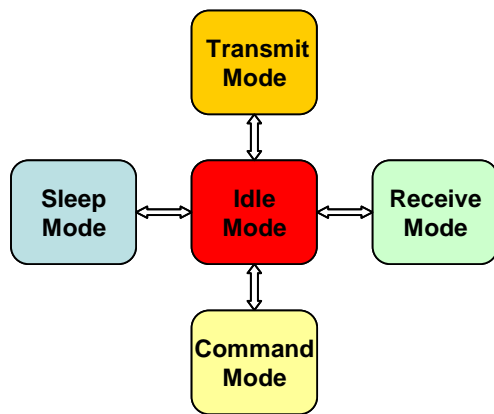


Figure 7. Modes of operation

The RF-modules can operate in four modes. Most of the time the microcontroller forces the RF-module into sleep mode. Every ten minutes the sleep line is left and the module changes to idle mode and then to receive mode. The module searches the network and tries to find a connection to the master board which is defined by the destination address bytes in the EEPROM.

C. Network topology

The module is able to communicate with the 802.15.4 protocol configured for non-beacon communication.

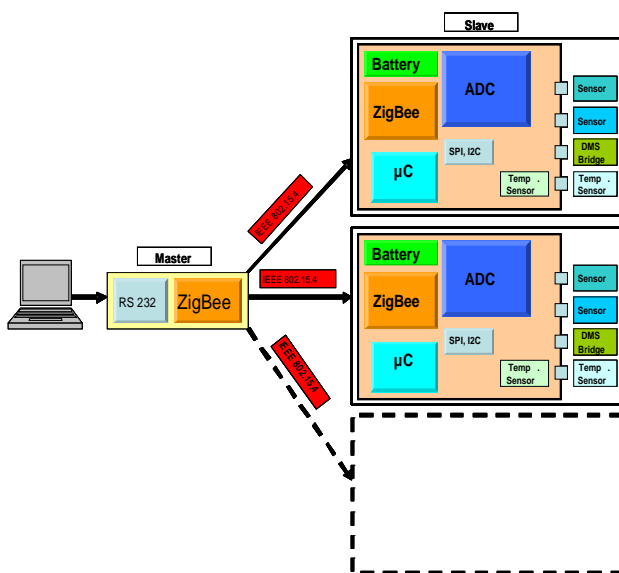


Figure 8. Network topology

The 802.15.4 is a standard for wireless communication put out by the IEEE (Institute for Electrical and Electronics Engineers) and was developed with lower data rate, simple

connectivity and battery application in mind. ZigBee protocol uses the 802.15.4 standard as a baseline and adds additional routing and networking functionality. Generally, a non-beacon network includes two types of nodes. One type of them is a Reduced Function Device (RFD) which mainly performs a function as a sensor and the other type is an (FFD) which performs a function as a path for communication data. In the non-beacon network, several tens or several hundreds of RFDs are connected to a single FFD so as to constitute a single network. In the constituted network, the function of such RFD is almost simple. The RFD is in an active state to perform communication only at the moment when the communication is required (chiefly at the moment when a predetermined signal is generated from a sensor embedded in a terminal), and the RFD is in a sleep state during the remaining time, thereby minimizing power consumption. [7]

In a NonBeacon (w/ Coordinator) system, the Coordinator can be configured to use direct or indirect transmissions. If the SP (Cyclic Sleep Period) parameter is set to '0', the Coordinator will send data immediately. Otherwise, the SP parameter determines the length of time the Coordinator will retain the data before discarding it. Generally, SP (Cyclic Sleep Period) and ST (Time before Sleep) parameters should be set to match the SP and ST settings of the End Devices.

A Peer-to-Peer communication is established with two receivers which do not depend on Master/Slave relationships. This is important because the Master program in the PC can only support one serial communication interface.

In the current implementation the wireless network only consists of end devices and there is no coordinator present. In that mode the end devices do not attempt to associate to a coordinator on power up but will only operate as specified by the source-address, the destination-address and channel parameters defined in the EEPROM.

III. APPLICATION

A. Hardware

The first application is also used for the field tests and is employed as external weight measurer and as a temperature sensor. For this a conventional weighing machine with the possibility to measure the weight of beehives as you can see below in Figure 9 is adapted. The temperature range for these electronic components lies between -30°C to $+70^{\circ}\text{C}$. The weight is measured with strain gages, which are implemented in the machine.

The weight cells are placed at random within a distance of two hundred meters and can weigh beehives up to one hundred kilogram. Also, the temperature inside the beehive is monitored using an integrated circuit temperature sensor with an SPI bus. With this system it is possible to monitor the weight for the whole summer season with one battery pack. The Master circuit is connected to a standard notebook running Windows XP.



Figure 9. Weight cell application rear image

B. Software

A comprehensive software flowchart of the Slave circuit is given below in Figure 10:

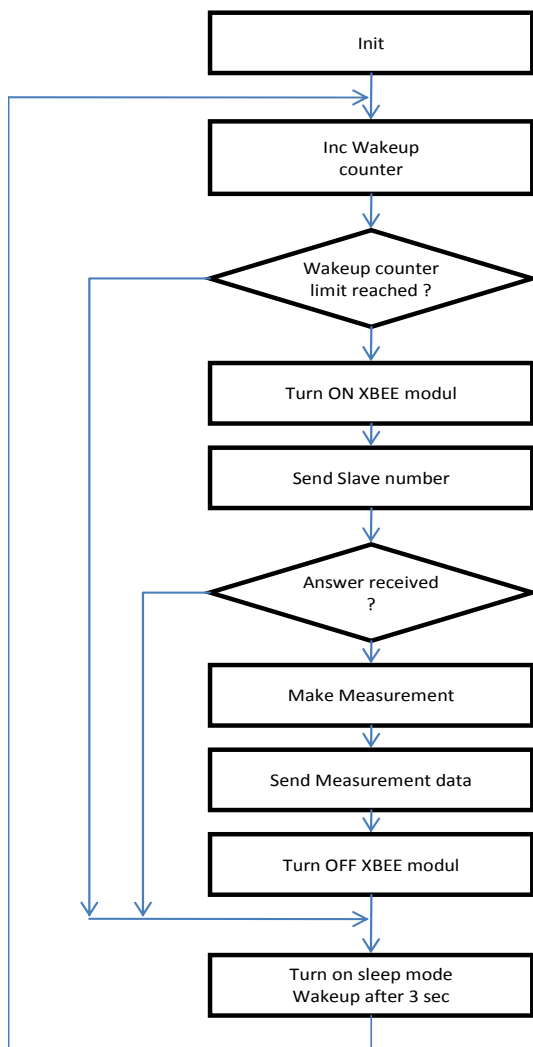


Figure 10. Software flowchart of the Slave circuit

The firmware on the slave modes is programmed to wake up every ten minutes and send the measured data on demand. The master board periodically changes its own source address according to the destination address of the slave module. This enables simple peer-to-peer communication via a standard UART with any application that supports only one communication port.

More precisely, the software of the Slave circuit starts with the initialization and the increase of the wakeup counter.

This wakeup counter has to reach a defined limit, because the wakeup time is limited for this type of microcontroller. If the counter does not reach the limit, the microcontroller changes into the sleep mode and after three seconds the process will start again. If the counter has reached the limit the XBEE module will turn on. The next step is that the Slave transmits the ATU number to the master circuit. Now there are two opportunities: if no answer is received, the microcontroller changes into sleep mode; otherwise the measurement will start. After that the measurement data is sent. Then the XBEE module is turned off, the microcontroller changes into the sleep mode and the process will start again.

A comprehensive software flowchart of the Master circuit is given below in Figure 11:

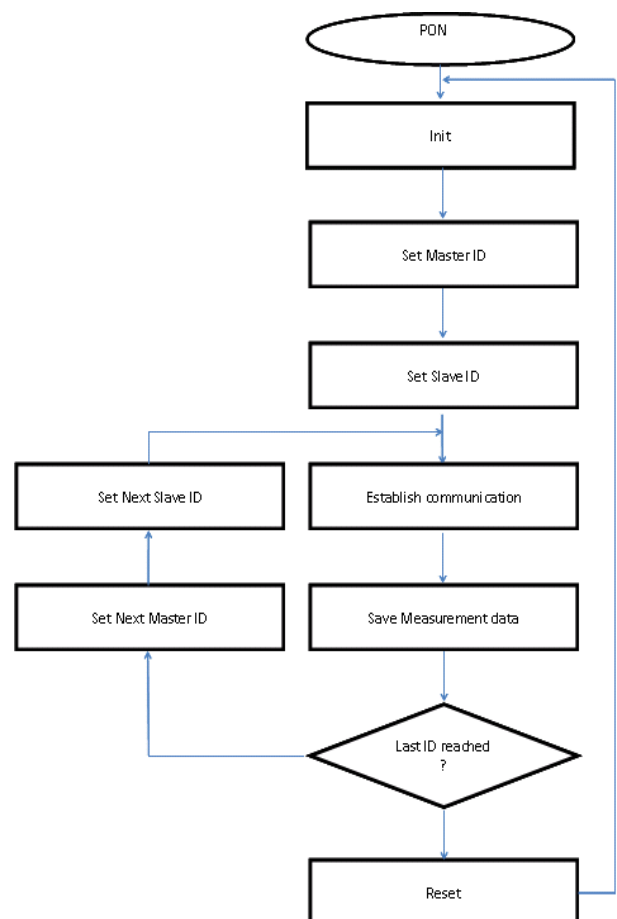


Figure 11. Software flowchart of the Master circuit

The Master circuit software starts with power on and the initialization follows. After this process the Master module has to set the Master identification and the suitable Slave identification. Then the communication between the Master and Slave system is established. After that the measurement data is saved and it is monitored whether all Master and Slave identifications have been measured. If the last ID does not reach its address the program continues increasing the IDs, otherwise the whole system is reset and the flow starts again.

On the one hand the software is realized with μ Vision wedge (program control for the Microcontroller P89LPC9336) and on the other hand visualization, representation and/or the control element for the user with LabView.

In the actual implementation a LabView program is written which monitors ten weight cells in an agricultural application for continuous control of the weight of beehives in order to check the optimal time for harvesting.

The data can be processed by export function easily in Excel. The software function plots the received data information which includes the actual weight, the ambient temperature and also the internal temperature of the beehives. A decrease of the weight is able to be represented relatively rapidly and clearly. Due to this information the apiculturist can choose the most suitable moment to open the beehives and to centrifuge the honeycombs. Another advantage is that the apiculturist can see from the charts if something is wrong in the beehives: for example, if the weight curve does not increase it is possible that a parasite, like the varroa mite, is in the beehives.

IV. CONCLUSION

The measurement system described in this paper offers high accuracy monitoring at low costs. Due to this cost reduction it is possible to monitor significantly more beehives at the same costs than with other solutions on the market.

In this paper only one application was presented. Yet, due to its high resolution it is possible to connect a variety of different sensors.

Therefore the following alternative fields of application are possible: Industrial engineering and automation technology: system control, (quality) monitoring of goods, transmission sensor data; Medical Engineering: transmission of patient data; Home and Building Automation; Consumer Electronics; Agriculture; Computer Peripherals.

V. REFERENCES

- [1] VISHAY, "Strain Gage Selection: Criteria, Procedures, Recommendations", *Tech Note TN-505-4*
- [2] Mary Mc Carthy, "Peak-to-Peak Resolution Versus Effective Resolution", *Analog Devices, Application note AN615*
- [3] Texas Instruments Incorporated, *Production Data Information ADS 1232/1234*
- [4] Chulho Won, Jong-Hoon Youn, Hesham Ali, "Adaptive Radio Channel Allocation for Supporting Coexistence of 802.15.4 and 802.11b", *2005 IEEE 62nd Vehicular Technology Conference (VTC2005-Fall)*

- [5] Dr.S.S.Riaz Ahamed, The Role of ZigBee Technology in Future Data Communication System, www.jafit.org/volumes/research-papers/Vol5No2/5Vol5No2.pdf
- [6] MaxStream, "Product Manual v1.xAx-802.15.4 Protocol"
- [7] Tae-Joon Kang Chan-Hee Jun, *Method of Minimizing Electric Power Consumption in Non-Beacon Networks*, www.faaqs.org/patents/app/20090172435

VI. AUTHORS

Ioan Turcin - CAMPUS 02 University of Applied Science, Körblergasse 126, A - 8021 Graz (e-mail: ioan.turcin@campus02.at).

Manfred Pauritsch - CAMPUS 02 University of Applied Science, Körblergasse 126, A - 8021 Graz, (e-mail: manfred.pauritsch@campus02.at).

Petra Weber - CAMPUS 02 University of Applied Science, Körblergasse 126, A - 8021 Graz (e-mail: petra.weber@campus02.at).