

Tomáš PAWLENKA\*

## DESIGN OF CAN BASED SMART MOTION SENSOR

### NÁVRH INTELIGENTNÍHO POHYBOVÉHO SNÍMAČE PRO CAN SBĚRNICI

#### Abstract

The work deals with the design and implementation of smart motion sensor based on MEMS technology for use on high-speed CAN bus. The MEMS MPU-9250 chip with a three-axis accelerometer, gyroscope and magnetometer was chosen as the sensor. The main goal is to create a plug-and-play smart sensor unit solution with its own measurement and configuration software and with possibility of basic configuration via CAN bus.

#### Abstrakt

Práce se zabývá návrhem a realizací inteligentního pohybového snímače na bázi MEMS určeným pro nasazení na vysokorychlostní sběrnici CAN. Jako snímač byl zvolen modul s MEMS čipem MPU-9250, který je vybaven tříosým akcelerometrem, gyroskopem a magnetometrem. Cílem je vytvoření „plug-and-play“ řešení s vlastním měřicím a konfiguračním softwarem a s možností základní konfigurace prostřednictvím CAN sběrnice.

#### Keywords

MEMS, accelerometer, CAN bus, motion sensor

## 1 INTRODUCTION

Smart sensors are the basis of many systems, and they are almost everywhere. It is a device that has a sensing element and a processing unit implemented in one compact solution. The processing unit processes the signals from the sensing element and provides the processed data in some normalized range or through some type of digital transmission [1].

In the automotive, systems for measuring vehicle dynamic behavior are often used. That are usually MEMS-based systems that have whole measuring and control chain implemented on a single chip. One of these systems is also subject of the design of the smart motion sensor unit, which this article deals with [2].

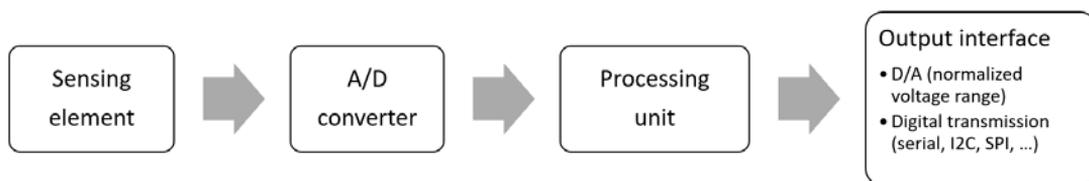


Fig. 1 Smart sensor structure

---

\* Ing., Department of Control Systems and Instrumentation, Faculty of mechanical engineering, Technical University of Ostrava, 17. listopadu 2172/15, 70800 Ostrava, Tomas.Pawlenka@vsb.cz

Currently, in automotive is usually used CAN bus (Controller Area Network) for data transmission. It is a high-speed communication bus designed by Bosch especially for use in cars **Chyba! Nenalezen zdroj odkazů..**

The main motivation of this work is creation of the plug-and-play device for deployment on CAN bus with possibility to measurement and configuration through own interfaces and applications. Other motivation is to find the best frequency of data providing.

## 2 DESCRIPTION OF SELECTED COMPONENTS

The basis of proposed device is MEMS unit based on MPU-9250 chip and it is shown in the Fig. 2. It consists of the triple-axis gyroscopes, accelerometer and magnetometer. Gyroscopes include user-programmable full-scale ranges of  $\pm 250$ ,  $\pm 500$ ,  $\pm 1000$ , and  $\pm 2000^\circ/\text{sec}$ , 16-bit analog-digital converters and digitally programmable low-pass filter. Accelerometer includes programmable full-scale ranges of  $\pm 2g$ ,  $\pm 4g$ ,  $\pm 8g$ ,  $\pm 16g$  and 16-bit analog-digital converters as well and magnetometer is based on hall-effect magnetic sensor with magnetic concentrator. Resolution of output data is 14-bit and full-scale range is  $\pm 4800$  uT. Communication speed with MPU registers via I<sup>2</sup>C is about 400 kHz [4].



**Fig. 2** Smart motion unit MPU-9250

As a main control unit, the Arduino NANO board was chosen, and it is shown in the Fig. 3. It is interfaced with ATmega328P microcontroller powered by 16 MHz clock source. ATmega328P is a 8-bit microcontroller with reduced instruction set (RISC) based on Harvard architecture and belongs to AVR family of microcontrollers [5][6].



**Fig. 3** Arduino NANO control unit

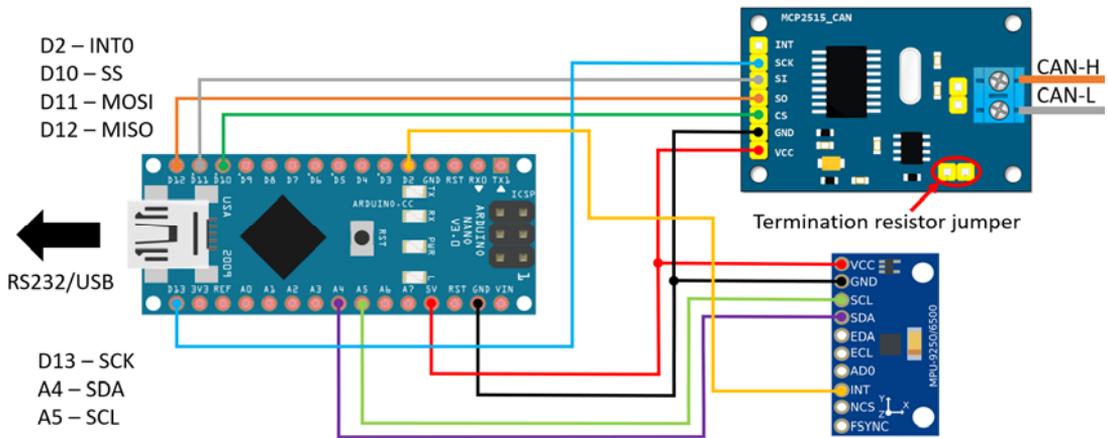
The Fig. 4 shows the selected CAN bus interface based on the MCP2515 chip with high-speed CAN transceiver TJA1050. It is powered by 8 MHz clock and it can communicate with Arduino via SPI interface. It has optional 120  $\Omega$  termination resistor, which can be activated by jumper. Maximum CAN bitrate is 1 Mbit/s. Also, it supports CAN V2.0B protocol specification [6].



**Fig. 4** CAN bus interface based on MCP2515 chip

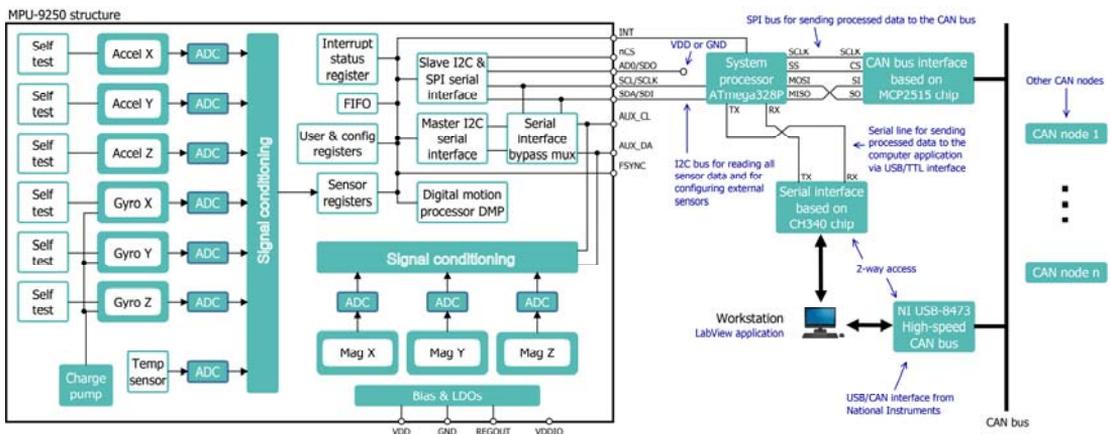
### 3 DESIGN AND IMPLEMENTATION

In the next step, the method of linking the individual components was selected and a wiring diagram, which is shown in the Fig. 5, was created. CAN bus interface is connected to the Arduino unit with SPI interface and MPU-9250 unit via I<sup>2</sup>C bus.



**Fig. 5** Wiring diagram of selected components

The Fig. 6 shows the inner structure of MPU and communication links between all components. The MPU includes integrated Digital motion processor, which acquires data from accelerometers, gyroscopes and magnetometers and processes them.



**Fig. 6** Structure and communication diagram

The resulting data can be read from DMP registers by host processor - in that case ATmega328P via I<sup>2</sup>C. Data are then sent to the serial line via serial interface based on CH340 chip or

to the CAN bus via MCP2515 based CAN bus interface. The measurement and configuration software running on the workstation can use two-way access to the sensor data. One way is the connection with USB cable directly to the Arduino unit. The other way is the connection to the CAN bus with use of supported USB/CAN interface from National Instruments [4].

For the connection of all components together, the connection board was designed in CADSOFT Eagle software and then manufactured. Board was interfaced with D-SUB connector to be more robust against vibration. The final smart motion sensor design is shown in the Fig. 7. The package was designed and implemented especially for this solution by 3D printing technology.

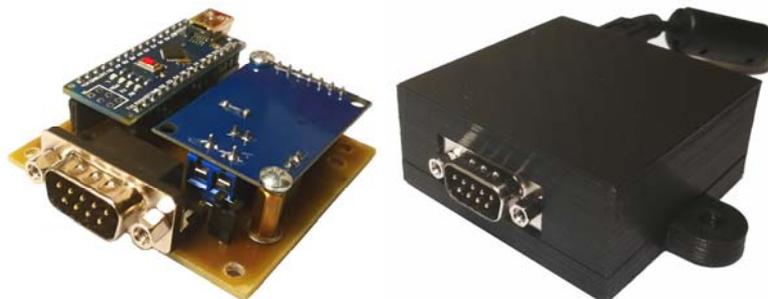


Fig. 7 Connection board with all components

#### 4 CONTROL UNIT PROGRAMMING

The Arduino IDE was used as a development environment for programming the controller and it is shown in the Fig. 8. The environment was developed in Java and is used to program supported platforms. The number of these platforms is large and still growing. There are also options for installing official and custom libraries and boards. The environment also includes many examples that make programming easier [5].

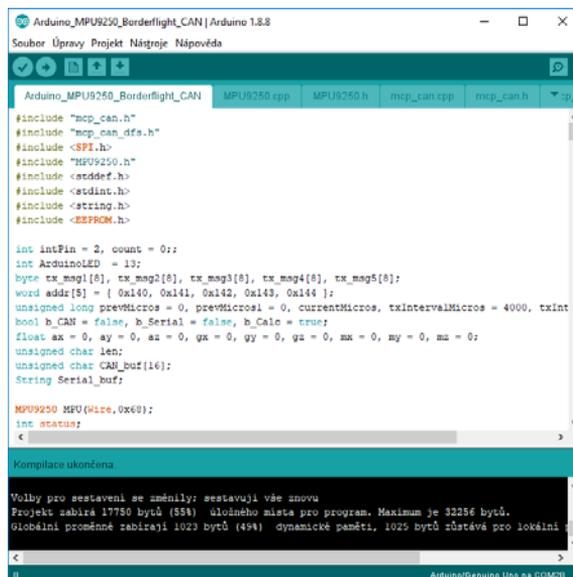
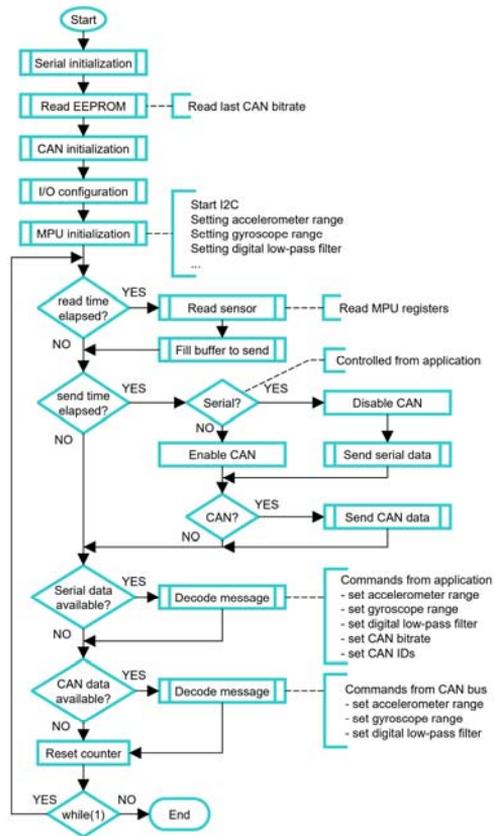


Fig. 8 Arduino development environment

Next picture Fig. 9 shows description of control unit operating algorithm by flowcharts. The program starts by initializing serial and CAN communication modules, reading the last device configuration from EEPROM, configuring the I/O unit and initializing the MPU.



**Fig. 9** Operating algorithm of control unit

This is followed by the main program loop. The communication between the control unit and the MPU is Master-Slave. In this case, the controller reads data from the MPU at predefined time intervals and then sends it to a serial line or CAN bus. In the next step, incoming data from both communication interfaces are monitored. These data can represent commands from applications such as requirements for setting accelerometer and gyroscope ranges, setting a digitally programmable lowpass filter bandwidth, setting the CAN baud rate, or changing CAN IDs. The program also includes communication control. By default, data are sent via CAN bus. However, if serial transmission is enabled from the application, the CAN module is disabled to increase computing power. As soon as the serial transmission is terminated, the data are sent on the CAN bus again.

## 5 COMPUTER SOFTWARE DEVELOPMENT

For the measurement, configuration and data acquisition, the two applications were developed in LabVIEW software from National Instruments. It is advanced software that uses block programming, which makes this development very easy and fast [7].

The program of both applications consists of two loops. The first loop serves for reading and writing data on the appropriate communication interface, and the second loop processes the data, visualizes them and optionally archives them into the TDMS file. The Fig. 10 shows the main loop design of CAN based application in LabVIEW.

The first application “MPU-USB” is focused on the communication with MPU through the USB. For successful connection it is necessary to select number of communication port and baud rate and click on the "Start serial" button. This application has options for setting MPU, changing CAN

baud rate and CAN IDs and for archiving measurement to TDMS file. Design of this application is shown in the Fig. 11.

The second "MPU-CAN" application is developed to communicate with the MPU directly via the CAN bus using the National Instruments CAN bus interface supported by LabVIEW. NI USB-8473 was used as this interface and it is shown in the Fig. 12. For programming CAN bus communication in LabVIEW, it was necessary to install additional CAN bus library. For successful communication, it is important to select the correct CAN baud rate and enter the IDs of CAN messages to be read. MPU configuration or data archiving is also possible in this application. The final design of "MPU-CAN" application can be seen in the Fig. 13.

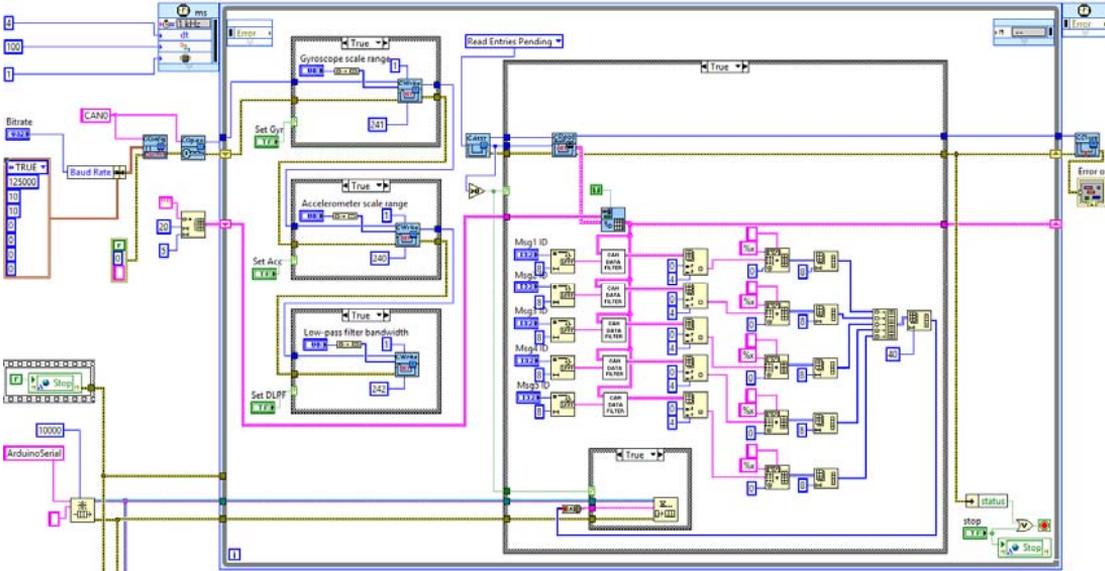


Fig. 10 Main loop of MPU-CAN application

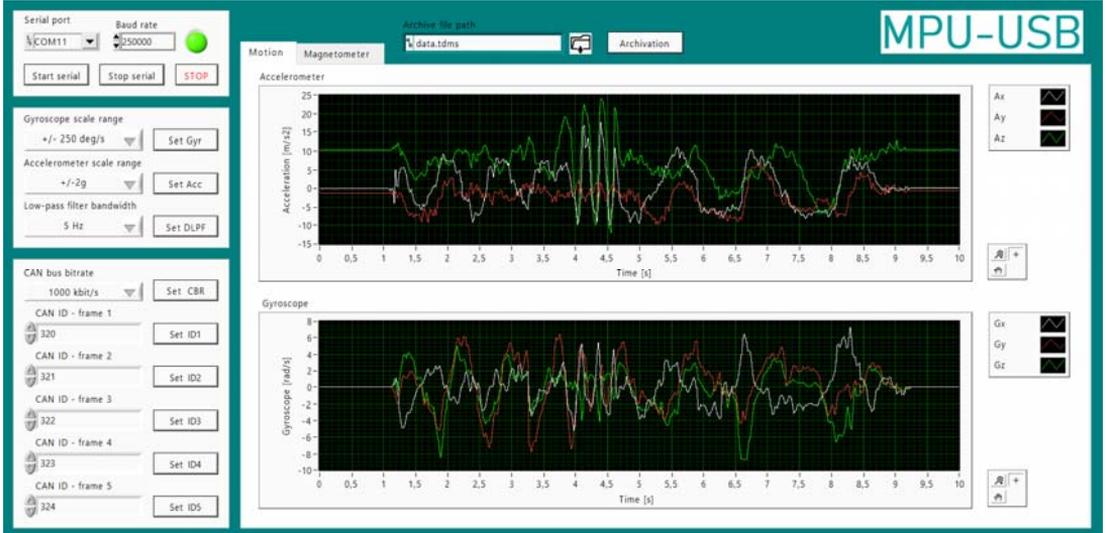
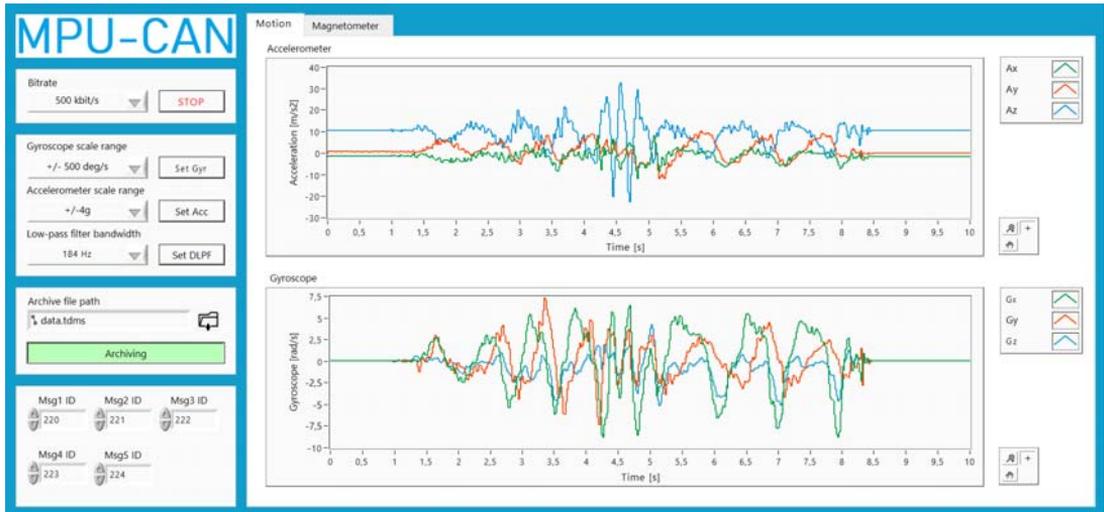


Fig. 11 Application for measurement and configuration via RS232



**Fig. 12** CAN bus interface NI USB-8473 from National Instruments



**Fig. 13** Application for measurement via CAN bus with NI USB-8473

### 3 CONCLUSIONS

In this work, the configurable smart motion sensor unit for CAN bus deployment was designed and implemented. The best achieved frequency of reliable data providing is 100 Hz via RS232 and 250 Hz via CAN bus. As an improvement, the use of an ATmega328P chip with a higher oscillator frequency may be proposed or it may be replaced by more powerful unit, for example 32-bit SAMD21 microcontroller with 48 MHz frequency. In case of using SAMD21, the speed of data providing could be increased approximately three-times. For the measurement, configuration and data acquisition, the LabVIEW applications were developed. One application is for access via USB/RS232 and the other is for access via CAN bus with use of specialized CAN bus interface from National Instruments.

### REFERENCES

- [1] KIRIANAKI, N. V. Data acquisition and signal processing for smart sensors. New York: J. Wiley, 2002. ISBN 0-470-84317-9.
- [2] JURÁNEK, M. Měření na vozidlech a jejich diagnostika v reálném čase. Ostrava: VŠB – Technická univerzita Ostrava, 2013. ISBN 978-80-248-3007-0.
- [3] ŠKUTA, J. & KULHÁNEK J. Controll of car LED lights by CAN/LIN bus. In: *Proceedings of the 2015 16th International Carpathian Control Conference (ICCC)*. Szilvasvarad, Hungary, May 27-30, 2015, s. 486-489. ISBN 978-1-4799-7370-5. DOI: 10.1109/CarpathianCC.2015.7145128.

- [4] TDK, c2019. [Online]. Available from: <https://www.invensense.com/>. [Accessed: 17-May-2019].
- [5] ARDUINO, c2019. [Online]. Available from: <https://www.arduino.cc/>. [Accessed: 17-May-2019].
- [6] MICROCHIP TECHNOLOGY, c1998-2019. [Online]. Available from: [www.microchip.com](http://www.microchip.com). [Accessed: 22-Jan.-2019].
- [7] NATIONAL INSTRUMENTS: *Automated Test and Automated Measurement Systems*, c2019. [Online]. Available from: <http://www.ni.com/>. [Accessed: 17-May-2019].