

# Multiprotocol Sensor Network Based on Inertial MEMS Sensors Part II. Processing Algorithms and Experiments

Emil Ivanov Iontchev, Ilian Slavkov Damyanov,  
Petar Stefanov Kapanakov and Rossen Georgiev Miletiev

**Abstract** - The current paper represents the processing algorithms of the multiprotocol sensor network based on inertial MEMS sensors which is capable of measuring the accelerations of the given point of the network. Also some measurement results are shown to

**Keywords** – MEMS sensors, multiprotocol network

## I. INTRODUCTION

In our previous paper [1] we introduce the network topology and prepare an analysis of the multiprotocol sensor network. There we define the maximum bus nodes and bus speed. In this paper we analyze the sensor network processing principle and show some experimental results.

## II. SENSOR NETWORK PROCESSING PRINCIPLE

The sensor network processing algorithm is shown at Figure 1. The main problem is directed to the simultaneous sampling of the unit accelerations and conversion into digital words. The synchronization of the sensor sampling times is solved by sending of the given codeword to all sensors ('S'). When slave sensors receive this codeword they initialize the inertial sensor chip simultaneously and this guarantees that the sensors are synchronized. When the initialization procedure is completed than the master sensor scans the network nodes by sending the all available sensor number to the bus. When the recognition procedure is completed the master sensor sends the requests only to the active sensor numbers to save time and to increase the sampling frequency of the network.

When the sensor recognition procedure is accomplished, the microcontroller begins to scan the active inertial sensors via CAN or USART network.

Emil Iontchev is with the Higher School of Transport “T. Kableshev” 158 Geo Milev Street, Sofia 1574, Bulgaria, e-mail: e\_iontchev@yahoo.com

Ilian Damyanov is with the Faculty of Transportation at Technical University of Sofia, 8 Kl. Ohridski Blvd, Sofia 1000, Bulgaria, e-mail: idamyanov@tu-sofia.bg

Petar Kapanakov is with the ENCO Vending Ltd Company, Sofia 1000, Bulgaria, e-mail: kapanakov@enco-vending.bg

Rosen Miletiev is with the Faculty of Telecommunications at Technical University of Sofia, 8 Kl. Ohridski Blvd, Sofia 1000, Bulgaria. e-mail: miletiev@tu-sofia.bg

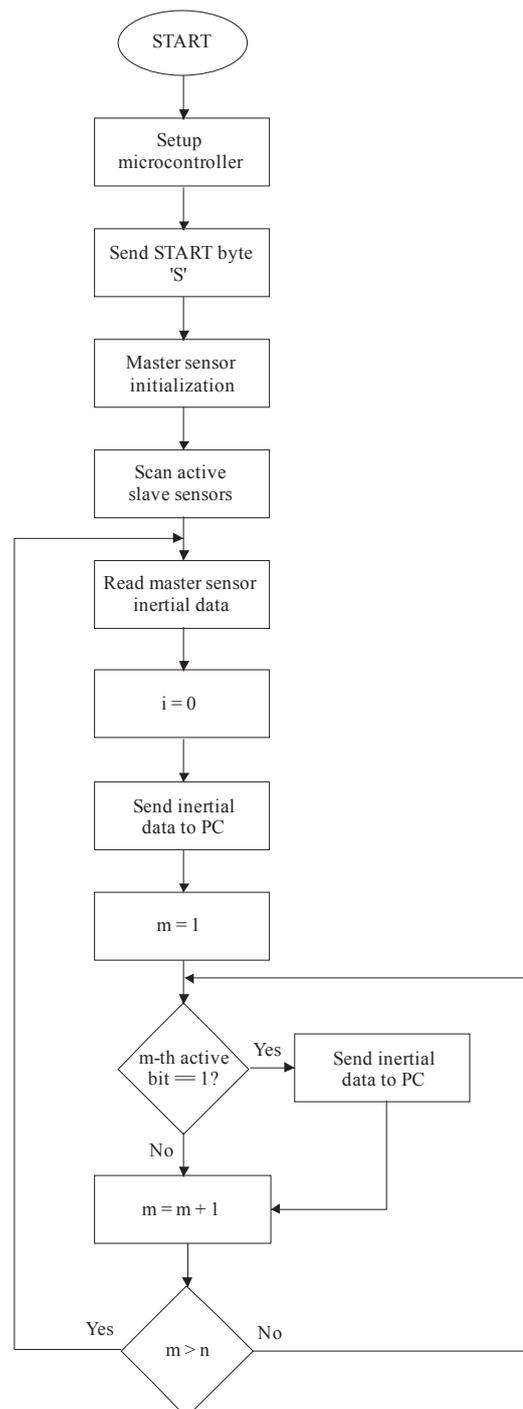


Figure 1. Master sensor processing procedure

The processing procedure of the master sensor is described at Figure 1. Initially the master sensor has to

decide which slave sensor is active or inactive. The scanning procedure is shown at Figure 2. If the slave sensor responded to the master sensor request, then the corresponding bit in the sensor matrix is set. Otherwise it is cleared.

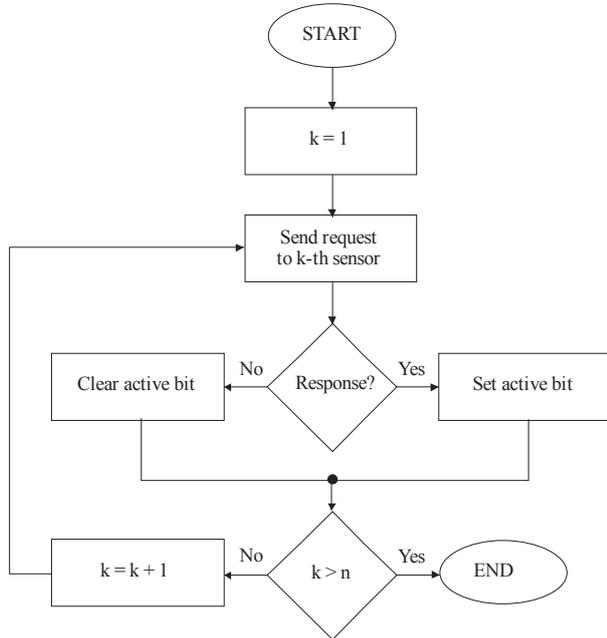


Figure 2. Master sensor initial scanning procedure

When the recognition procedure is accomplished, then the master sensor sends a request to all active sensors consecutively. When the slave sensor receives the request it checks the number of the requested sensor and if this number does not coincide with the slave sensor number, it repeats the request to the next slave sensor. If the both numbers are the same, the slave sensor reads the data from the inertial sensor and sends them to the master sensor.

When the data reached the master sensor, it forms a data packet (frame) which is (m+1) or (m+2) bytes long. The frame consists of the following fields:

- Synchronization byte
- Slave sensor number (1 byte) - optional
- Inertial data (m bytes)

These data are transmitted to PC according to the procedure, which is shown at Figure 3. This algorithm has a special feature, that the next request to the slave sensor is send before the termination of the transmission procedure to PC. As is clearly visible at Figure 3, the next request is send at the second byte, which is send to PC. This algorithm is chosen to decrease the data transmission time and to achieve higher communication transfer, because when the master sensor sends data to PC, the slave sensor request is transmitted through the sensor bus to the corresponding slave sensor.

As the transfer to the PC is started, the master sensor may receive the response of the previously scanned sensor due to the full duplex operation of the USART interface. This also decreases the slave sensor scanning time.

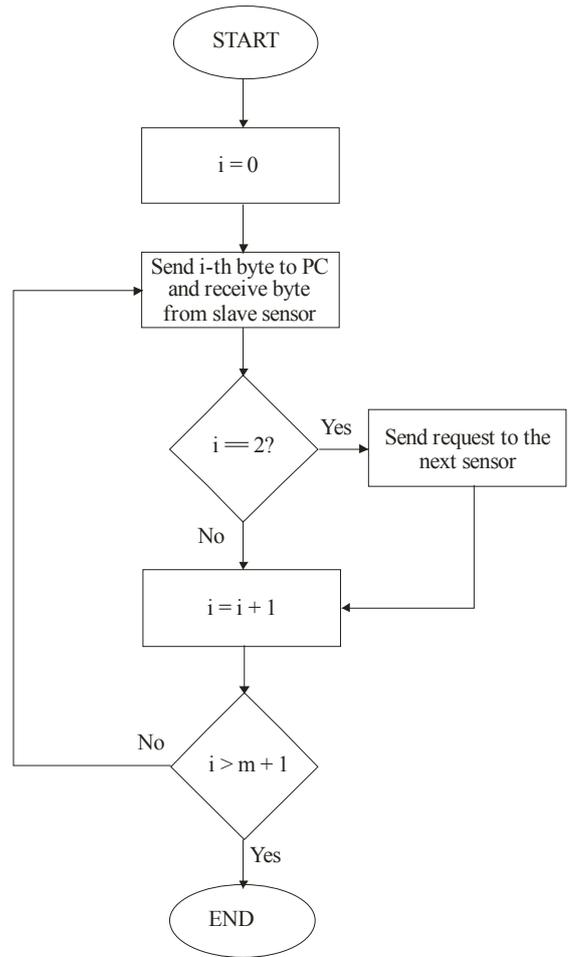


Figure 3. Master sensor data transmission procedure

### III. EXPERIMENTS

The selected inertial sensor network is tested on a passenger car to determine the relative movement of the car suspension components. The network consists of three inertial sensors based on the 3D single chip LIS3LV02DQ produced by ST [2].

The sensor network assembly is shown at figure 4 and the measured data of the sensor network are given at Figure 5.



Figure 4. Sensor network assembly

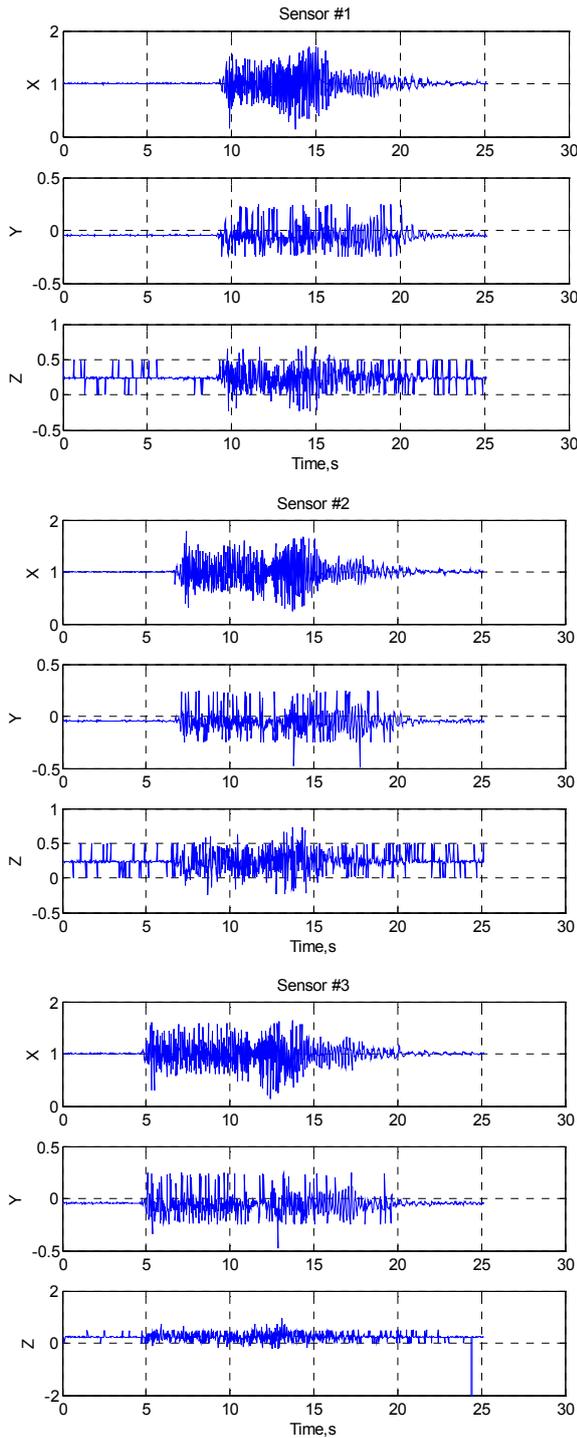


Figure 5. Network sensor measured data

The measured accelerations allow to calculate the relative movements between the suspension components according to the equations:

Relative velocities:

$$\begin{aligned} v_x &= \int a_x dt \\ v_y &= \int a_y dt \\ v_z &= \int a_z dt \end{aligned} \quad (1)$$

Relative shifting:

$$\begin{aligned} S_x &= \int v_x dt \\ S_y &= \int v_y dt \\ S_z &= \int v_z dt \end{aligned} \quad (2)$$

As the obtained result are read via SPI interface of the inertial sensor, these data are digital ones and the equations (1) and (2) are modified as follows:

$$v_{i+1,x} = v_{i,x} + (a_{i+1,x} - a_{i,x})\Delta t$$

$$v_{i+1,y} = v_{i,y} + (a_{i+1,y} - a_{i,y})\Delta t$$

$$v_{i+1,z} = v_{i,z} + (a_{i+1,z} - a_{i,z})\Delta t$$

$$S_{i+1,x} = S_{i,x} + v_{i,x}\Delta t + \frac{1}{2}(a_{i+1,x} - a_{i,x})(\Delta t)^2$$

$$S_{i+1,y} = S_{i,y} + v_{i,y}\Delta t + \frac{1}{2}(a_{i+1,y} - a_{i,y})(\Delta t)^2$$

$$S_{i+1,z} = S_{i,z} + v_{i,z}\Delta t + \frac{1}{2}(a_{i+1,z} - a_{i,z})(\Delta t)^2$$

The equations (3) and (4) show that the calculation accuracy is increased if the sampling frequency is higher. This fact explains the tendency towards the bus speed multiplication. The used inertial sensors allow reading of the inertial data up to 2560Hz.

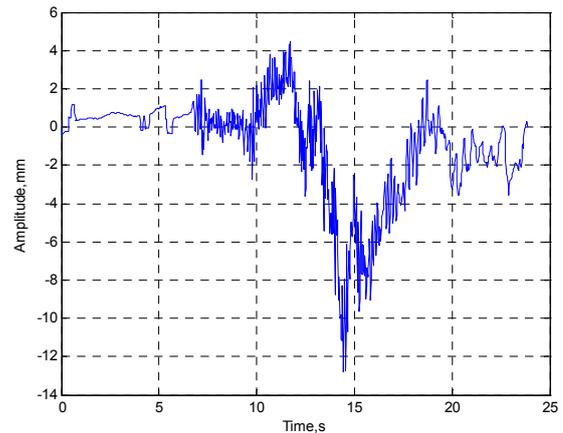


Figure 6. Calculated relative shifting

The calculated relative shifting according to the equation (4) is shown at Figure 6. The shown results may be useful when the component wearing out is established and the obtained data may be compared with the same results measured and calculated for components with good working order and repaired ones.

#### IV.CONCLUSION

The proposed multiprotocol network sensor system is capable of a simultaneous measurement of the inertial data in the given number of points. These data are transferred to PC in a real-time situation and may be further analyzed to calculate the relative velocities and shifting between the installed points.

The described system may be useful when a high level of security towards system failure is required due to the used CAN or USART bus. If the main CAN bus is dropped then

the data may be transmitted via USART bus. This obstacle is very important in a remote controlled devices such as unmanned airplane vehicles (UAV), robots, etc.

#### ACKNOWLEDGMENT

This paper was prepared and supported by the National Fund under **contract** number No.DTK02/2-2009.

#### REFERENCES

- [1] Emil Ivanov Iontchev, Ilian Slavkov Damyanov, Petar Stefanov Kapanakov, Rossen Georgiev Miletiev - Multiprotocol sensor network based on inertial MEMS sensors Part I. Network topologies and analysis, XIX International Scientific and Applied Science Conference *Electronics-ET2010*, Bulgaria, Sozopol, 22-24 September, 2010
- [2][www.st.com/stonline/products/literature/ds/10175/lis3102dq.pdf](http://www.st.com/stonline/products/literature/ds/10175/lis3102dq.pdf)

Copyright of Annual Journal of Electronics is the property of Technical University of Sofia, Faculty of Electronic Engineering & Technologies and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.