1st International Conference on Sensing Technology
November 21-23, 2005 Palmerston North, New Zealand

# High-level model of an acceleration sensor with feedback as part of an inertial navigation system

Erik Markert, Göran Herrmann, Dietmar Müller and Ulrich Heinkel
Department of Electrical Engineering and Information Technology,
Chemnitz University of Technology, Reichenhainer Str. 70, 09126 Chemnitz, Germany
e-mail: erik.markert@etit.tu-chemnitz.de; goeran.herrmann@infotech.tu-chemnitz.de

## Abstract

This paper presents an approach for modeling and simulation of an acceleration sensor with feedback. This sensor will be part of an inertial navigation system which is currently developed at Chemnitz University of Technology. Modeling is achieved using the high-level description languages VHDL-AMS and SystemC-AMS. The sensor consists of four capacitive segments and one mass segment, aligned in a semicircle. Each capacitive segment includes fixed and movable combs. Any acceleration causes a rotation and a displacement of the seismic mass coupled with moveable combs. The capacity altered with these movements is measurable. Pulswidth modulated (PWM) voltages are used to reset the sensor's displacement. So the effective range of the sensor is enlarged. The sensor model is simulated and verified.

**Keywords**: acceleration sensor, feedback, PWM, modeling, VHDL-AMS, SystemC-AMS

## 1 Introduction

Today navigation systems based on the Global Positioning System (GPS) are state of the art. But due to limited precision and difficult sattelite connection inside buildings also navigation based only on onboard sensors is necessary. These so called inertial navigation systems determine their position usually on motion data like acceleration and rotation.

A new sensor structure [1] is used to measure acceleration by detecting changes of the capacity. The derivation of its VHDL-AMS-model based on geometry is shown in [2]. This model is broadened by pulsewidth modulated feedback voltages which reset the sensor to its resting position. A new analysis circuit restores acceleration values based on the PWM count.

The navigation system not only includes analogue sensor parts but also a digital and a software subsystem for further signal processing, control data generation and graphical position data presentation.

Usually VHDL-AMS is used for mixed-signal modeling. But description of software in this language is difficult. To fill this gap, an Open SystemC Initiative (OSCI) study group currently develops SystemC-AMS [3], a mixed-signal extension to digital SystemC. This language allows modelling and simulation of analogue, digital and software components in one tool. A first public version (0.13beta) has been released by the OSCI study group.

Section 2 gives an overview of the system structure and summarizes the functionality of the digital and the software subsystem. Section 3 shows the derivation of the analogue sensor model. This model is simulated and validated in section 4. The last section consists of a conclusion and an outlook.

## 2 System model

### 2.1 General overview

The inertial navigation system consists of three main parts: an analogue sensor with additional error correction, a digital coordinate transformation with supporting point inclusion and a PC-based software part. Figure 1 gives an overview of the system structure.

The sensor measures the analogue values sent from the "environment" and converts them to digital values. In the digital part acceleration and rotation values are converted into position values and corrected by information from other signal sources (camera, map, etc.). The PC is used to plot the sensor's position and to generate configuration data (cfg) for the digital and the analogue part. The system parts are explained in the following sections.

### 2.2 Analogue part

The analogue subsystem consists of six sensors (three for acceleration and three for rotation detection), analogue error correction for each value and analogue to digital conversion. Figure 2 shows the structure of

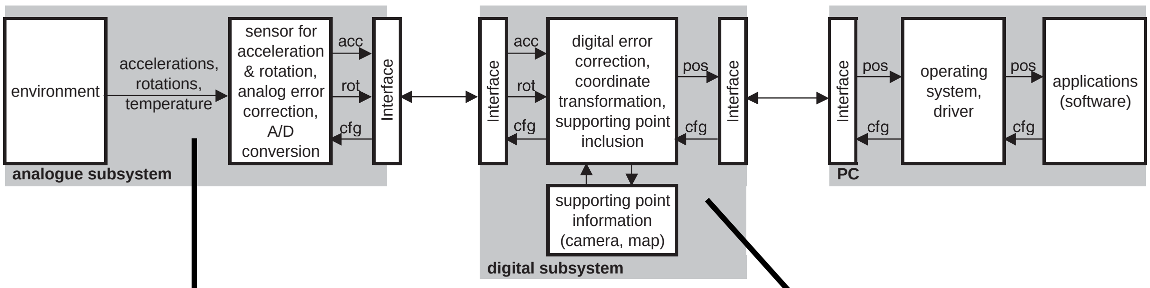


**Figure 1:** System overview

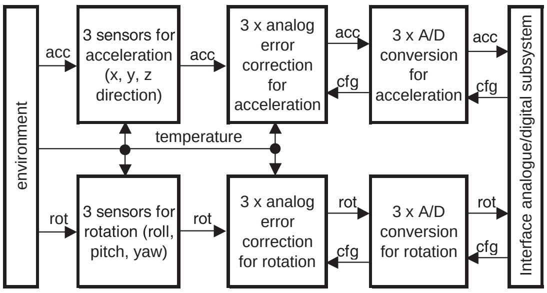


**Figure 2:** Analogue subsystem

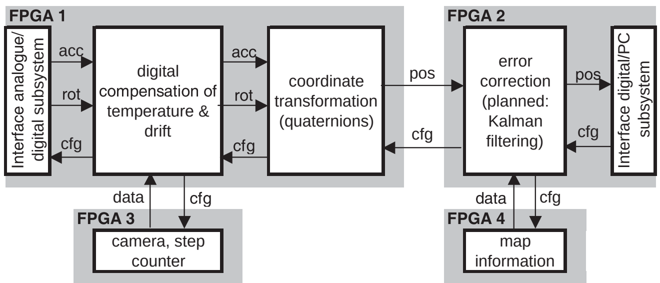


**Figure 3:** Digital subsystem

the subsystem The sensor model is specified in section 3. The analogue to digital converters (A/D) transform the analogue voltage to a digital 12 bit signed value with a peak conversion rate of 1.2 MSample/s.

### 2.3 Digital part

Further signal processing takes place in the digital part. This component consists of three main modules and two supporting units as shown in figure 3. Four XILINX VirtexE 600 FPGAs are planned to carry this functionality as protoype.

The coordinate transformation from the sensor system to the environment system is implemented using quaternion rotation [4]. Quaternions are complex numbers with 3 imaginary parts. A 3D coordinate transformation can be done with only few hardware effort using adders and serial multipliers.

Offset correction will be done using a camera and a step counter. The final error correction is planned to be realised in terms of a Kalman filter supported by additional position information generated by a map.

### 2.4 Software

The software subsystem consists of two main parts: the operating system (OS) and the user applications. The user programs run as procedures scheduled by the OS. Only a simple and abstract OS model is necessary for this inertial navigation system so a small own OS model is implemented. It only communicates with one port (the interface to the FPGA boards) and has no interrupt handling. Data exchange between user programs is done by activation signals.

Recently only two user applications are implemented: One target application saves the received position data to a file in MATLAB-format to allow graphical display of the resulting path. The other application generates configuration information for the digital and the analogue subsystem.

## 3 Modeling of the sensor with feedback

### 3.1 Sensor overview

The sensor consists of one seismic mass and four comb segments for capacitive acceleration detection. Figure 4 shows the structure of the sensor.

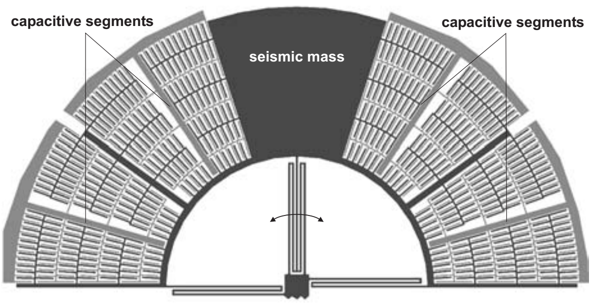


**Figure 4:** Sensor structure

An acceleration causes a linear displacement of the seismic mass. The three springs force the segments on a circular path. This rotational movement leads to capacity changes in the comb segments and therewith to a current flow which could be analysed.

For simplifiaction issues and for simulation speedup the behaviour of the segments is concentrated in their centres of gravity. Figure 5 shows the structure's simplification into 11 points and the used coordinate system..

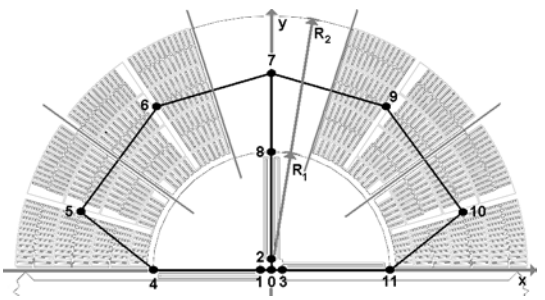


**Figure 5:** Simplification of sensor structure

The sensor is divided into a mechanical (displacement), an electrical (capacity calculation) and a feedback/analysis part. The derivation of modeling equations is described in [2]. In this paper only the resulting equations should be epitomized.

### 3.1.1 Mechanical part

The rotational acceleration $a_s$ of a segment depends on the translational accelerations $a_x$ and $a_y$, the current displacement angle $\alpha$ and the segment's position angle $\beta_s$:

$$a_S = a_x \cdot \sin(\alpha + \beta_S) + a_y \cdot \cos(\alpha + \beta_S) \quad (1)$$

Therefore the segment's torque can be calculated in dependence of the segment's mass $m_s$ and the coordinates of the centre of gravity $x_s$ and $y_s$:

$$M_S = m_S \cdot a_S \cdot \sqrt{x_S^2 + y_S^2} \quad (2)$$

The motion equation for rotation can be refined to:

$$\sum M_S - \sum M_R = \sum J_S \cdot \frac{d^2\alpha}{dt^2} + k \cdot \frac{d\alpha}{dt} + \sum c_{rot} \cdot \alpha \quad (3)$$

Thereby $M_R$ specifies the reset torque caused by applied voltages. The moment of inertia Js and the spring constant $c_{rot}$ are calculated from geometry data and material constants. The rotational displacement $\alpha$ is determined by solving the differential equation.

### 3.1.2 Electrical part

The rotational displacement $\alpha$ causes a change of the segment's capacity $C_s$. This leads to a current flow $I_s$ depending on the measurement voltage $U_s$:

$$I_S = U_S \cdot \frac{dC_S}{dt} + C_S \cdot \frac{dU_S}{dt} \quad (4)$$

In addition the mechanical reset torque results on voltage and capacity changes:

$$M_R = \frac{1}{2} \cdot U_S^2 \cdot \sqrt{x_S^2 + y_S^2} \cdot \left(\frac{dC_{s1}}{ds} + \frac{dC_{s2}}{ds}\right) \quad (5)$$

### 3.1.3 Feedback and analysis part

The dependence of the reset torque $M_R$ and therewith the sensor's displacement $\alpha$ on the applied voltage $U_S$ is utilized to compensate the sensor movement. For this reason the voltages of the segments are pulswidth modulated. Figure 6 shows the structure of the analysis and feedback circuit.

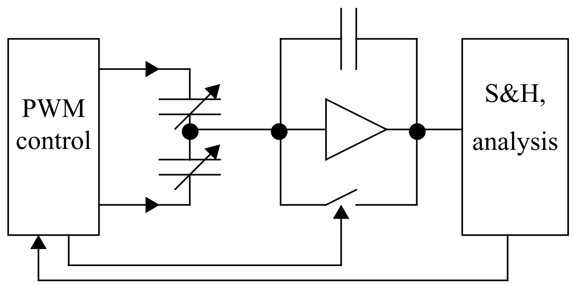


**Figure 6:** Analysis and feedback circuit

An acceleration causes a change in sensor capacity. This change leads to a current flow through the differential capacity formed by two sensor segments. The current is transformed to a voltage by means of an integrating operation amplifier which behaves proportional to the acceleration. An oscillator uses the polarity of the voltage to modify pulsewidth modulated segment voltages which cause a reset movement at the segments.

## 3.2 Model Implementation

At first a VHDL-AMS model of the sensor was implemented as presented in [2]. But software parts complicate the realisation of a VHDL-AMS system model. So the sensor model was transfered to SystemC-AMS [3]. This new language allows descriptions of software as well as digital and analogue hardware. It is an extension library of SystemC which bases on C++, so simulation can be executed with any C++-compiler. Hence software written in C++ can be integrated with only few changes depending on the model of the operating system and the underlying hardware. Several constructs allow the modeling of parallelism, reactivity and runtime performance for digital hardware.

### 3.2.1 Modeling constructs in SystemC-AMS

SystemC-AMS offers two possibilities for the description of analog behaviour. On the one hand linear electrical nets can be used which are solved by Modified Nodal Analysis (MNA). In these networks linear elements (resistors, inductivities, capacities) and current/voltage sources may be used. On the other hand Static DataFlow networks (SDF) allow high level models of analogue problems.

In SDF networks the elements communicate via directed dataflows. Every module interface is characterized as input or output port. Feedback is allowed in these nets but it has to be decoupled by interposing a one cycle delay. Due to oversampling this technique is appropriable in most cases. Inside the modules any time-dependent equation can be used to describe the behaviour. A Laplace solver offers a simple way for modeling transfer functions and differential equations.

Due to the limitation on directed networks and simple linear nets SystemC-AMS simulations run about 60 times faster than VHDL-AMS comparison simulations.

### 3.2.2 Implementation in SystemC-AMS

The decoupling of feedback causes a model structure as shown in figure 7. The delay on the segment voltages are caused in reality by the PWM controller. The behaviour of the subsystems is described using Laplace transfer functions.

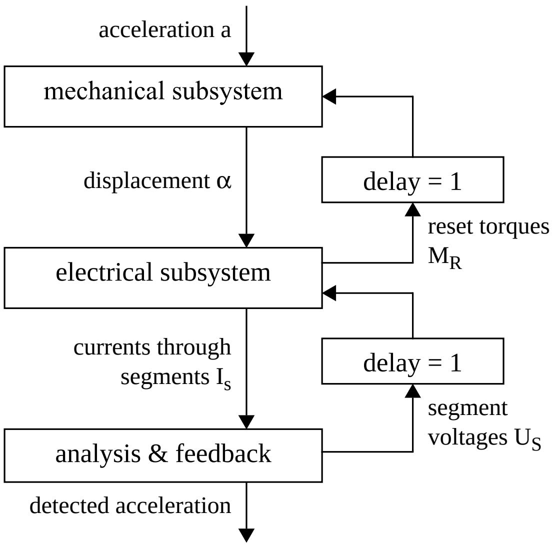


**Figure 7:** Structure of the sensor's SDF-model

## 4 Simulation results

The SystemC-AMS models of the mechanical and the electrical subsystem were verified using VHDL-AMS simulations and an FEM model. As a dynamic testcase the input is stimulated with step functions. The deviations from the reference models amount about 2% in amplitude height.

At the moment parameters for analysis and PWM are still under research. The SystemC-AMS model helps the designers to evaluate new parameter settings quickly. Figure 8 shows the displacement of the sensor as response to a 1g step at 25 ms. The upper dottet curve represents the behaviour without PWM feedback, the lower one with activated PWM loop which forces the sensor back to its resting position.

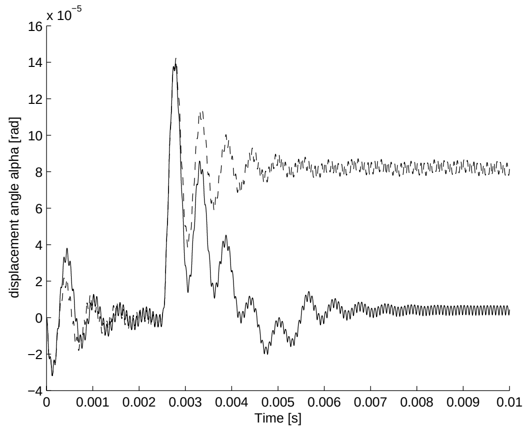


**Figure 8:** Sensor displacement as step response

The sensor output voltage with activated PWM feedback is plotted in figure 9. It represents the measured acceleration in multiples of gravitation constant.

After finishing parameter search by simulation the chosen algorithms will be verified by experiments.

## 5 Conclusion and outlook

The paper presents the modeling of a capacitive acceleration sensor with feedback. The sensor will be part of an inertial navigation system which also consists of digital and software parts. For system modelling SystemC-AMS is used which allows simulation of all three subsystems in one tool. The sensor model is verified using a VHDL-AMS and an FEM model.

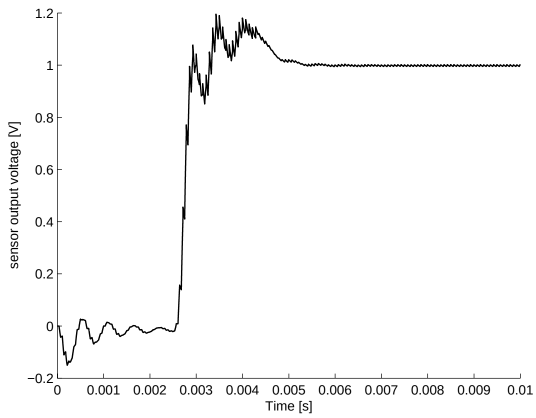


**Figure 9:** Sensor output voltage

After completion of all system components it is planned to use a walking machine (figure 10) as carrier for the inertial navigation system.

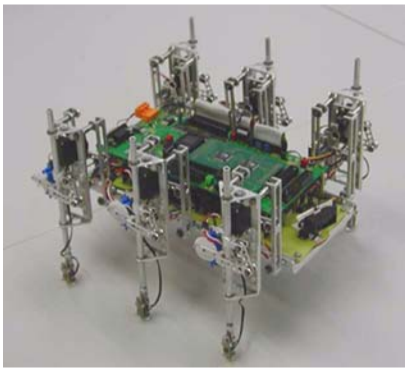

**Figure 10:** Walking machine

## 6 Acknowledgements

The work presented here has been done within the project A2 "System Design" in cooperation with project A1 "Component Development" of the SFB 379 (collaborative research center), which is funded by the German Science Foundation (DFG).

## 7 References

[1] Billep, D., Dienel, M., "Patent Application DE 102004046411.1 Beschleunigungssensor."

[2] Markert, E.; Schlegel, M.; Dienel, M.; Herrmann, G.; Müller, D.: "Modeling of a new acceleration sensor as part of a 2D sensor array in VHDL-AMS." *Proceedings Nanotech 2005*, Anaheim CA, Vol. 3 pp 399-402 (2005).

[3] Einwich, K. et. al., "White Paper SystemC-AMS Study Group", http://www.systemc-ams.org, visited on 31/5/2005.

[4] Hart, J. C.; Francis, G. K.; Kauffman, L. H.: "Visualizing Quaternion Rotation." *ACM Transactions on Graphics*, Vol.13, No. 3, pp 256-276 (July 1994).