

SystemC-AMS-Modell eines DeltaC-U-Wandlers für ein Inertialnavigationssystem

Erik Markert¹, Hendrik Zeun^{1,2}, Göran Herrmann¹, Dietmar Müller¹ und Ulrich Heinkel¹

¹Fakultät für Elektrotechnik und Informationstechnik, TU Chemnitz, Reichenhainer Str. 70, 09126 Chemnitz

²jetzt bei Endress + Hauser CONDUCTA GmbH + Co.KG, Landsberger Straße 28, 04736 Waldheim

E-Mail: ¹erik.markert@etit.tu-chemnitz.de

Kurzfassung

Dieser Beitrag beschreibt die Modellierung eines analogen DeltaC-U-Wandlers in SystemC-AMS. Der Wandler ist Teil eines inertialen Navigationssystems, das neben analogen elektrischen auch aus mechanischen und digitalen Komponenten sowie aus Software besteht. SystemC-AMS bietet die Möglichkeit, alle Teilkomponenten des Systems in einer Sprache zu beschreiben und gemeinsam zu simulieren. Der DeltaC-U-Wandler setzt die durch Beschleunigungen verursachten Kapazitätsänderungen in einem mikromechanischen Sensor in Spannungsänderungen um. Diese verhalten sich proportional zu den einwirkenden Beschleunigungen. Simulationsergebnisse zeigen die Korrektheit des Modells.

1 Einleitung

Heutige Navigationssysteme basieren meist auf GPS. Bei Anwendungen innerhalb von Gebäuden stoßen diese jedoch an ihre Grenzen, da sie ständigen Kontakt zu Satelliten benötigen. Daher ist es für gebäudeinterne Navigationssysteme notwendig, ihre Position anhand von selbst gemessenen Daten relativ zu einer Ausgangsposition zu bestimmen. Für die dazu notwendige Beschleunigungsmessung wird ein neuentwickeltes mikromechanisches Sensorarray [1] wie in **Bild 1** dargestellt verwendet.

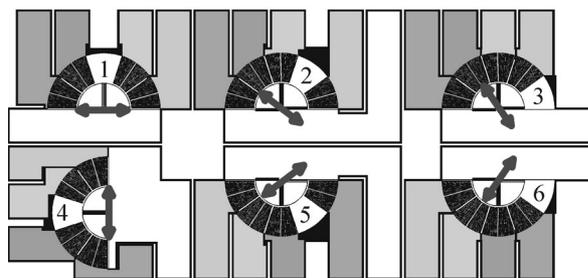


Bild 1: Beschleunigungssensorarray [1]

Jedes Element des Arrays erfordert einen eigenen DeltaC-U-Wandler. Nach der anschließenden Analog-Digital-Wandlung erfolgt die weitere Messdatenverarbeitung in digitaler Hardware und Software. **Bild 2** gibt einen Überblick über das Gesamtsystem. Die Funktionalität der einzelnen Komponenten wird in [2] näher erläutert.

Wie aus Bild 2 ersichtlich besteht das System neben Analogkomponenten aus digitaler Hardware und Software. Um das gesamte System beschreiben und simulieren zu können, ist also eine Beschreibungssprache notwendig, die alle drei Domänen unterstützt. Die im Analogbereich übliche Arbeit mit Spice oder FEM-Simulatoren wie Ansys ist hier nicht möglich. Auch die Beschreibung mit VHDL-AMS [3] wird durch den Softwareanteil deutlich erschwert.

Um solch hochgradig heterogene Systeme trotzdem in ihrer Gesamtheit beschreiben zu können, wird derzeit SystemC-AMS [4] entwickelt. Diese Erweiterungsbibliothek zum digitalen SystemC [5] erlaubt die Modellierung von analogen und Mixed-Signal-Komponenten. Neben der Eignung von SystemC-AMS für digitale Systeme mit elektrischem Analoganteil [6] ist auch die Beschreibung mikromechanischer Systeme erprobt worden [7].

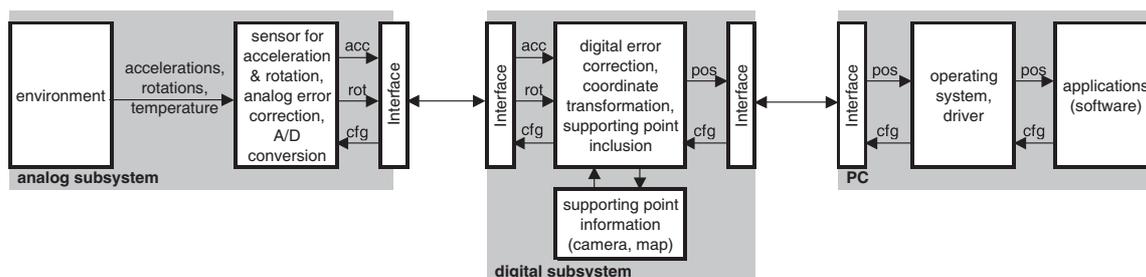


Bild 2: Überblick über das Gesamtsystem

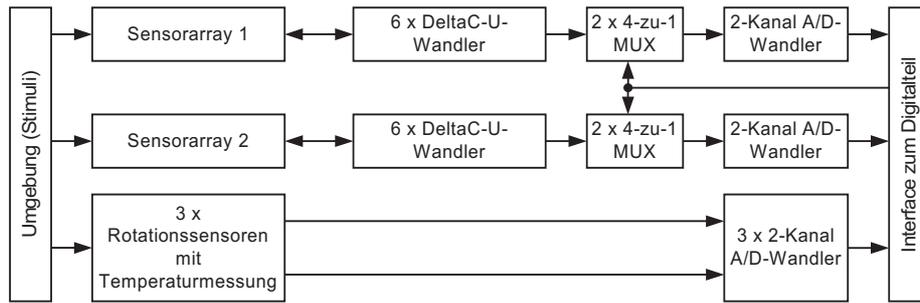


Bild 3: Überblick über das analoge Teilsystem

Im Folgenden soll das analoge Teilsystem kurz erläutert werden. Abschnitt 3 stellt die Modellierungsmöglichkeiten von SystemC-AMS dar, welche in Abschnitt 4 auf den DeltaC-U-Wandler angewandt werden. Abschnitt 5 zeigt einige Simulationsergebnisse.

2 Überblick über das analoge Teilsystem

Das analoge Teilsystem besteht aus zwei Beschleunigungssensorarrays, drei Drehratesensoren sowie der zugehörigen Wandlerelektronik. **Bild 3** zeigt den Aufbau des Subsystems. Jedes Sensorarray umfasst sechs Einzelsensoren. Die Einzelsensoren besitzen eine halbkreisförmige Struktur (siehe **Bild 4**).

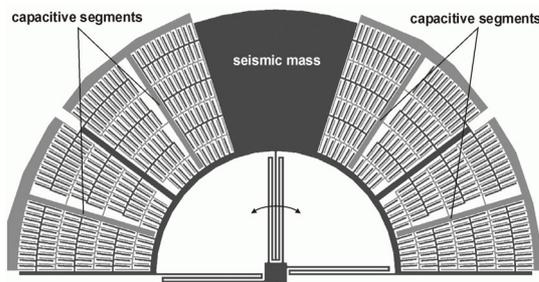


Bild 4: Struktur eines Einzelsensors [1]

Vier der fünf Segmente dienen der kapazitiven Beschleunigungsdetektion. Das fünfte Segment ist als seismische Masse ausgeführt, wobei durch Umsetzen

der Masseposition die Detektionsachse des Sensors verändert werden kann.

Jeweils zwei kapazitive Segmente bilden einen Differentialkondensator. Dieser wird mit einem DeltaC-U-Wandler ausgewertet. Der Wandler generiert eine Dreiecksspannung, die in den Segmenten einen Stromfluss bewirkt. Eine einwirkende Beschleunigung führt zu einer rotatorischen Bewegung der Segmente und damit zu einer Kapazitätsänderung. Durch die gegenläufige Anordnung der Zinken in den Segmenten verdoppelt sich der Kapazitätshub im Differentialkondensator.

Der DeltaC-U-Wandler setzt den aus der Kapazitätsänderung resultierenden Stromfluss in einen Spannungswert um. Sein Aufbau mit angeschlossener Beschleunigungssensor-Differentialkapazität ist in **Bild 5** dargestellt.

Aus einem Spannungsgenerator wird eine Dreiecksspannung (U_{pol}) und die zugehörige Mittenspannung (U_{med}) geliefert. Die Mittelelektrode des Differentialkondensators ist mit der Mittenspannung verbunden, so dass durch die Dreiecksspannung ein ständiger Polaritätswechsel der Außenelektroden und damit eine kontinuierliche Umladung der Kapazitäten gegeben ist. Ein Umschalter am Anschluss der Differentialkapazität sorgt für die notwendige Gleichrichtung des Messsignals.

Wirkt auf den Sensor keine Beschleunigung, sind beide Kapazitäten gleich groß. Damit fließen in die Schaltung identische Ströme durch $In1$ und $In2$, was eine Auslöschung des Signals bei U_{out} zur Folge hat. Verschiebt sich jedoch die Mittelelektrode, tritt eine Stromdifferenz zwischen $In1$ und $In2$ auf.

Der aus fünf OPV bestehende Differentialverstärker setzt diese Stromdifferenz in einen Spannungswert

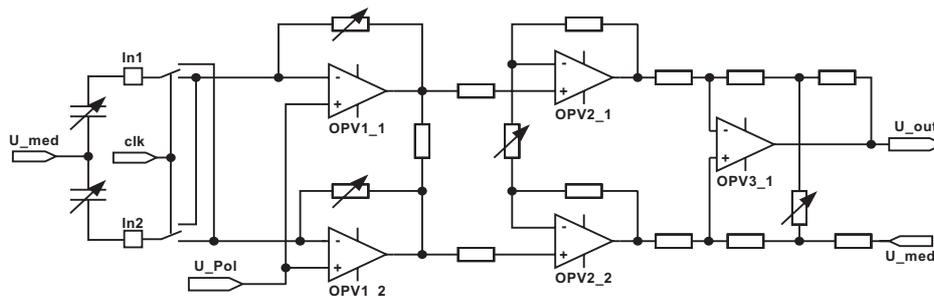


Bild 5: Schaltskizze des DeltaC-U-Wandlers

um. Dabei ist die Verstärkung der Stufen schrittweise einstellbar um Kapazitätsdifferenzen über einen größeren Messbereich detektieren zu können.

Um A/D-Wandler einzusparen, wird das Ausgangssignal in einem 4-zu-1-Multiplexer mit den Ausgangssignalen von zwei weiteren DeltaC-U-Wandlern verknüpft. Zusätzlich zu den drei Ausgängen der Wandler kann im Multiplexer als vierter Eingang das Mittenpotential U_{med} an den A/D-Umsetzer übergeben werden. Dieses wird für die drei den Multiplexer speisenden DeltaC-U-Wandler gemeinsam erzeugt. Die Kanalauswahl im Multiplexer erfolgt durch Steuerleitungen direkt aus dem digitalen Teilsystem. Dort wird aufgrund der gesendeten Auswahl-signale das A/D-gewandelte Messsignal im Eingangsspeicher abgelegt. Der Analog-Digital-Umsetzer ist in der Lage, zwei Kanäle mit einer 12-Bit-Auflösung zu wandeln.

Um Signalleitungslängen so kurz wie möglich zu halten, werden jeweils drei DeltaC-U-Wandler zusammen mit dem zugehörigen Multiplexer in einem Chip integriert. Zwei dieser ICs bilden zusammen mit einem Sensorarray eine Messeinheit. In dieser Einheit werden die Einzelchips direkt auf die Leiterplatte gebondet und mit einer gemeinsamen Schutzhülle umgeben. Als A/D-Wandler kommt ein zugekaufter Chip der Firma Analog Devices zum Einsatz.

3 Modellierung mit SystemC-AMS

Die C++-Bibliothek SystemC und ihre Erweiterung SystemC-AMS ermöglichen die Beschreibung von Hardware mit Standard-C++-Compilern. Spezielle Konstrukte erlauben die Modellierung von Parallelität, Reaktivität und Zeitverhalten in der sequentiellen Abarbeitung von C++-Softwarecode. Jegliche Hardware wird in SystemC(-AMS)-Module gekapselt. Ein Scheduler überwacht die parallele Ausführung der Modulfunktionen. Die Zeitweitschaltung erfolgt im digitalen SystemC ereignisgesteuert. Im analogen SystemC-AMS wird hingegen mit einer möglichst kleinen Zeitschrittlänge ein kontinuierlicher Zeitverlauf nachgebildet. Diese Zeitschrittlänge

hat jedoch signifikanten Einfluss auf die Simulationsgeschwindigkeit.

Zur Beschreibung von analogen Systemanteilen stehen in der genutzten SystemC-AMS-Version konservative und Datenfluss-Netze (SDF, static dataflow) zur Verfügung. In SDF-Netzen kommunizieren die Einzelelemente über gerichtete Datenströme miteinander. Jede Schnittstelle eines Moduls ist als Ein- oder Ausgang zu charakterisieren. Für die Beschreibung von Transferfunktionen und Differentialgleichungen steht in Version 0.15 eine Funktion zur Lösung von Laplace-Übertragungsfunktionen zur Verfügung. Nichtlineare Löser sowie Funktionen zur Lösung von State-Space-Gleichungen und Pol-Nullstellen-Problematiken befinden sich in der Entwicklung.

Neben der Modellierung von Datenflussnetzen bietet SystemC-AMS die Möglichkeit der Beschreibung linearer elektrischer Netze, die per MNA (Modified Nodal Analysis) gelöst werden. Dazu können lineare Bauelemente (Widerstände, Spulen, Kapazitäten) sowie gesteuerte Strom- und Spannungsquellen verwendet werden. Zur Anbindung an analoge Datenfluss-Netze sowie digitale Module sind Wandlerkomponenten definiert, welche Strom- bzw. Spannungswerte in Datenfluss-Werte sowie in Gegenrichtung umsetzen.

Für beide Modellierungsmöglichkeiten können sowohl Zeitbereichs- als auch Frequenzbereichssimulationen durchgeführt werden. Die Ausgabe der Simulationsdaten erfolgt in einer MATLAB-geeigneten Syntax, die mit geeigneten Werkzeugen wie MATLAB oder GNU Octave grafisch darstellbar ist.

Bild 6 zeigt den resultierenden Schichtenaufbau von SystemC-AMS. Auf den SystemC-Kernel setzt eine Synchronisationsschicht auf, welche die analogen Löser des 'Solver Layer' untereinander synchronisiert sowie die analoge Erweiterung an das digitale SystemC anbindet. Die Löser stellen dem Nutzer im 'Semantic Layer' Konstrukte zur Systembeschreibung zur Verfügung. Für elektrische Netze können diese bei den Sprachentwicklern der FhG EAS/IIS Dresden online eingesehen werden [8]. Auf den 'Semantic Layer' setzt der Nutzer seine Systembeschreibung auf. Die hier dargestellten Applikationsbereich beziehen sich exemplarisch auf die Realisierung des in Bild 3 dargestellten Systems.

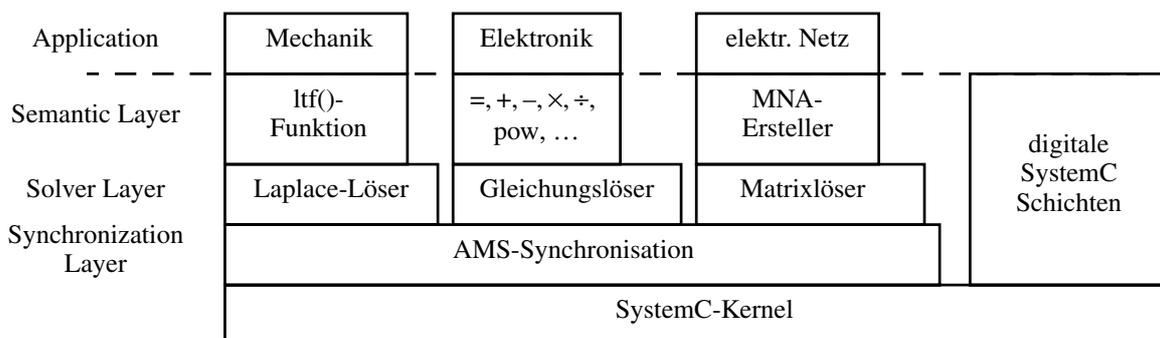


Bild 6: Einordnung in das SystemC-AMS-Schichtenmodell (nach [6])

4 SystemC-AMS-Modell des DeltaC-U-Wandlers

Für die Modellerstellung und Simulation wird SystemC-AMS Version 0.15 als Erweiterung zu SystemC 2.1 verwendet. Die jeweils aktuelle Version ist auf der SystemC-AMS-Homepage [4] verfügbar. Als Compiler kommt gcc-3.3 auf Solaris9 zum Einsatz.

4.1 Modell der Innenschaltung

Da für das Modell des DeltaC-U-Wandlers bereits ein Schaltplan existierte, bot sich die direkte Modellierung mittels linearer elektrischer Netze an. Die Topologie der in Bild 4 dargestellten Schaltung kann mit den in SystemC-AMS vordefinierten linearen Elementen umgesetzt werden. Voraussetzung dafür ist jedoch, dass die Operationsverstärker im linearen Bereich arbeiten. Sollten sie bis zu ihrer Begrenzung ausgereizt werden, erfordert dies einen nichtlinearen Netzwerklöser, der für eine der nächsten Versionen von SystemC-AMS geplant ist. Kontrollsimulationen der Schaltung ergaben, dass alle OPV in ihrem linearen Bereich arbeiten. Daher kann das vereinfachte OPV-Modell einer Differenzspannungsverstärkung eingesetzt werden. Dazu bietet sich das SystemC-AMS-Standardelement `sca_vcvs` (Spannungsgesteuerte Spannungsquelle) an. **Listing 1** zeigt den Ausgangs-OPV mit seiner Außenbeschaltung.

```
opv_u5 = new sca_vcvs("OPV5", 1e12);
opv_u5->np(opv5_out);
opv_u5->nn(vss);
opv_u5->ncp(opv5_in_p);
opv_u5->ncn(opv5_in_n);
r61 = new sca_r("R61", 10e3);
r61->p(opv5_in_n); r61->n(n_60);
r62 = new sca_r("R62", 10e3);
r62->p(n_60); r62->n(opv5_out);
r71 = new sca_r("R71", 10e3);
r71->p(opv5_in_p); r71->n(n_70);
r72 = new sca_r("R72", 10e3);
r72->p(n_70); r72->n(gnd);
```

Listing 1: OPV mit Außenbeschaltung

4.2 Modell der Anbindung an die Mikromechanik

Die Mikromechanik des Sensors wird mit Hilfe analytischer Gleichungen für rotatorische Bewegungen und Kräfte im elektrostatischen Feld nachgebildet. Die Herleitung der Gleichungen sowie Einzelheiten der Modellierung können in [9] nachgelesen werden. Eine Abbildung dieser analytischen Zusammenhänge auf elektrischer Ebene durch Analogietransformationen ist sehr aufwändig. Daher wurde die Mikromechanik als SDF-Datenflussnetz implementiert. Somit ergibt sich an der Schnittstelle zum DeltaC-U-Wandler ein Wechsel der Beschreibungsebene. Auf

elektrischer Ebene trägt jede Verbindung zwei Informationen: Strom und Spannung. Auf SDF-Ebene kann eine Verbindung nur eine Information weiterleiten. Eine elektrische Verbindung muss also mit zwei SDF-Signalen beschrieben werden.

Im konkreten Fall folgen In1 und In2 durch die OPV-Rückkopplungsschleifen dem Potential von U_{pol} . Der fließende Strom hingegen wird durch die Differentialkapazität bestimmt. Daher ergibt sich an den Pads In1 und In2 eine Konstruktion wie in **Bild 7** dargestellt. Dabei stellen gestrichelte Linien SDF-

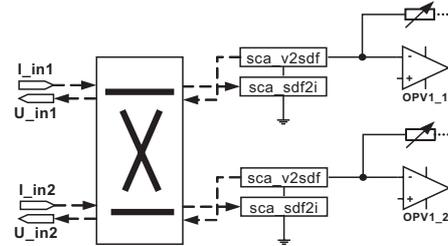


Bild 7: SDF-Elec-Umsetzer mit Switch

Datenpfade und durchgezogene Linien elektrische Leitungen dar. Die Schalteranordnung zur Gleichrichtung der Eingangsströme kann in die Konstruktion integriert werden. Dies ist für die Stabilität des Schaltungsmodells förderlich, da sonst bei wechselnden elektrischen Widerständen das Gleichungssystem des elektrischen Netzes neu initialisiert werden muss. Das Problem der Neuinitialisierung bei Umschaltungen ist nicht auf SystemC-AMS begrenzt, sondern stellt sich auch bei der Vergleichssimulation mit Advance MS (VHDL-AMS).

Listing 2 zeigt die SystemC-AMS-Beschreibung der Elemente am Eingang In1.

```
i_in1 = new sca_sdf2i("I_in1");
i_in1->p(In1); i_in1->n(vss);
i_in1->ctrl(sdf_in1);

in_sw = new cu_input_switch("INSW1");
in_sw->icap1(icap1);
in_sw->icap2(icap2);
in_sw->ucap1(ucap1);
in_sw->ucap2(ucap2);
in_sw->edge_rising(edge_rising);
in_sw->iout1(sdf_in1);
in_sw->iout2(sdf_in2);
in_sw->uin1(sdf_uin1);
in_sw->uin2(sdf_uin2);

v2sdf_in1=new sca_vd2sdf("in1_v");
v2sdf_in1->sdf_voltage(sdf_uin1);
v2sdf_in1->p(In1);
```

Listing 2: Beschreibung des Eingangs In1

Die Beschreibung des Umschalters auf SDF-Ebene besteht aus dem Wechseln von Signalzuweisungen abhängig vom Vorzeichen des Anstieges von U_{pol} . Dies wird durch das Signal `edge_rising` vom Typ `bool` repräsentiert, dessen Erzeugung im Signalgenerator für U_{pol} erfolgt. **Listing 3** zeigt den zum Umschalter gehörenden SystemC-AMS Code.

```

SCA_SDF_MODULE(cu_input_switch){

// Eingänge
sca_sdf_in<double> icap1, icap2;
sca_sdf_in<double> uin1, uin2;
// Umschaltsignal
sca_sdf_in<bool> edge_rising;
// Ausgänge
sca_sdf_out<double> iout1, iout2;
sca_sdf_out<double> ucap1, ucap2;

// Signalverarbeitung, wird zu jedem
// Simulationszeitschritt aufgerufen
void sig_proc(){
    if (edge_rising){
        ucap1 = uin1;
        ucap2 = uin2;
        iout1 = icap1;
        iout2 = icap2;
    }
    else{
        ucap1 = uin2;
        ucap2 = uin1;
        iout1 = icap2;
        iout2 = icap1;
    }
}

//Konstruktor
SCA_CTOR(cu_input_switch){};
};

```

Listing 3: Beschreibung des Eingangsumschalters

5 Simulationsergebnisse

Zur Prüfung der Funktionalität des analogen Teilsystems wurde eine Folge verschiedener Beschleunigungen an den Sensor angelegt. **Bild 8** zeigt diese Beschleunigungsfolge als gestrichelte Linie.

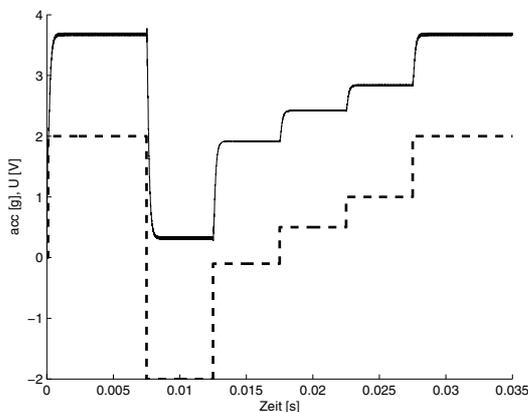


Bild 8: Simulationsergebnisse DeltaC-U-Wandler

Im Diagramm ist außerdem die Spannung am Ausgang des DeltaC-U-Wandlers abgebildet. Sie folgt der angelegten Beschleunigungsfolge, ist jedoch durch den Bezug auf die Mittenspannung im Pegel verschoben. Der nachgeschaltete Analog-Digital-Umsetzer beseitigt diese Verschiebung und wandelt die Spannung in einen vorzeichenbehafteten Ganzzahlwert. **Bild 9** zeigt den Signalverlauf am Ausgang des A/D-Wandlers mit einem Skalierungsfaktor von 400. Für das gesamte in Bild 2 dargestellte System

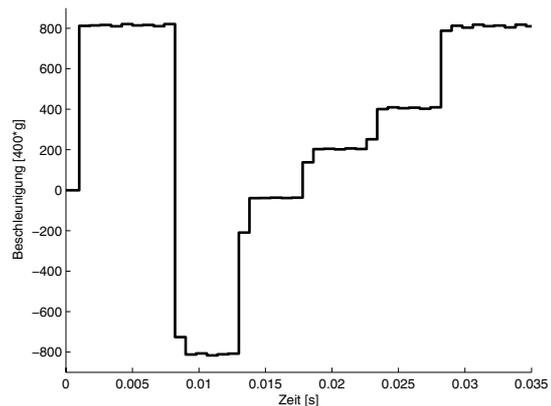


Bild 9: Signalverlauf am A/D-Wandler-Ausgang

beträgt die Simulationsdauer 7min 46s für 35ms Realzeit bei einer „analogen“ Zeitschrittlänge von 500ns. Zur Simulation wurde eine Sun-Blade2000 Workstation mit Solaris9 genutzt.

Das dynamische Verhalten des Wandlermodells wurde mit VHDL-AMS verifiziert. Als Stimuli wurden Ströme direkt an den Eingängen aufgeprägt. Bild 10 zeigt beide Simulationsergebnisse in einer Darstellung. Dabei weicht die durchgezogene Linie

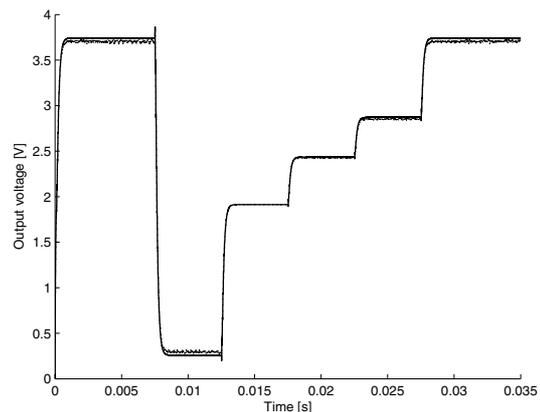


Bild 10: Vergleich der Ergebnisse mit VHDL-AMS

des SystemC-AMS-Ergebnisses kaum vom gepunktet dargestellten VHDL-AMS Ergebnis ab. In Bild 11 ist die relative Abweichung der beiden Simulationsergebnisse korrigiert um die Mittenspannung von 2V dargestellt. Hier ist deutlich zu sehen, dass sich die Differenzen auf einen Bereich zwischen 1% und 4% begrenzen. Die Peaks im Abweichungsgraph resultieren aus den Durchgängen durch die Mittenspannung bei 7,5s und 17,5s.

Eine Validierungssimulation des gesamten analogen Teilsystems inklusive des Sensorarrays in VHDL-AMS auf niedriger Abstraktionsebene erscheint aus Gründen der Simulationsdauer (> 8h) und der sich ergebenden Datenmenge nicht sinnvoll. Erste Messungen am gefertigten IC bestätigen die Simulationsergebnisse von SystemC-AMS.

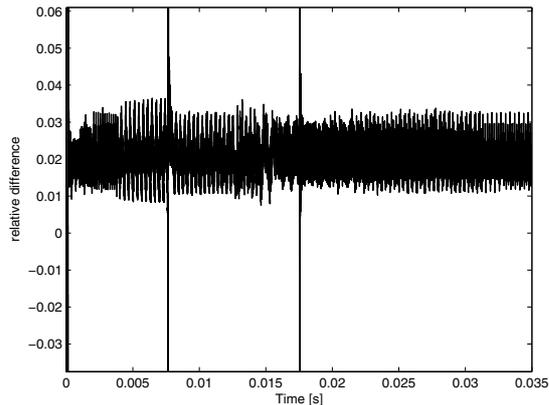


Bild 11: relative Abweichung zu VHDL-AMS

6 Zusammenfassung und Ausblick

Dieser Beitrag beschreibt die Modellierung eines analogen Kapazitäts-Spannungs-Wandlers mit SystemC-AMS. Er ist Teil eines inertialen Navigationssystems, welches neben analogen elektrischen und nichtelektrischen Komponenten auch digitale Hardware und Software umfasst. All diese Teilsysteme können mit SystemC-AMS in einer gemeinsamen Sprache beschrieben und simuliert werden.

Die Grundzüge des analogen Teilsystems sowie die prinzipiellen Modellierungsmöglichkeiten von SystemC-AMS wurden aufgezeigt. Die Verifikation des entwickelten Simulationsmodells erfolgte mittels VHDL-AMS und Messungen am gefertigten IC.

Die Realisierungen der genannten analogen und digitalen Systembestandteile werden derzeit noch separat getestet. Im Herbst 2006 soll das Inertialnavigationssystem auf einer in **Bild 12** dargestellten sechsbeinigen Gehmaschine („Hexapod“) im realen Einsatz erprobt werden.

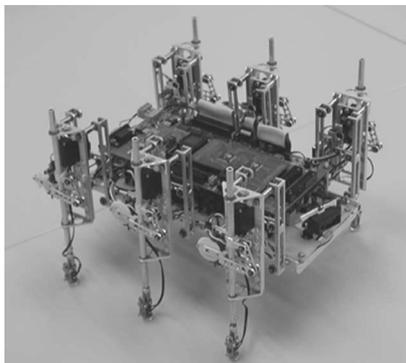


Bild 12: Gehmaschine als Prototypenträger

7 Danksagung

Die hier vorgestellte Arbeit entstand im Rahmen des Teilprojektes A2 „Systementwurf“ des von der DFG geförderten SFB 379 „Mikromechanische Sensor- und Aktorarrays“.

8 Literatur

- [1] Dienel, M.; Billep, D.; Dötzel, W.: Development of a drift compensated acceleration sensor array. 50. Internationales wissenschaftliches Kolloquium, 19.-23.9.2005, Ilmenau, 2005, pp 227-228, ISBN 3-932633-98-9
- [2] Markert, E.; Herrmann, G.; Müller, D.: System Model of an Inertial Navigation System using SystemC-AMS (short paper). FDL 2005, Lausanne, Schweiz, 27.-30.9.2005, pp. 73-76
- [3] IEEE Standard VHDL Analog and Mixed-Signal Extensions, IEEE Std 1076.1-1999, Design Automation Standards Committee of the IEEE Computer Society, 18. März 1999
- [4] SystemC-AMS Erweiterungsbibliothek zu SystemC, <http://www.systemc-ams.org>
- [5] Open SystemC Initiative (OSCI), SystemC 2.1 Language Reference Manual. 2005, <http://www.systemc.org>
- [6] Einwich, K.: Application of SystemC/SystemC-AMS for the Specification of a Complex Wired Telecommunication System. FDL 2005, Lausanne, Schweiz, 27.-30.9.2005, pp. 49-60
- [7] Markert, E.; Schlegel, M.; Michel, M.; Herrmann, G.; Müller, D.: Untersuchung der Anwendbarkeit von SystemC-AMS bei der Beschreibung von MEMS. 5. GMM/GI/ITG-Workshop "Multi-Nature Systems", Dresden, 18. Februar 2005
- [8] SystemC-AMS-Entwicklungshomepage, FhG EAS/IIS Dresden, <http://mixsigc.eas.iis.fhg.de/>
- [9] Markert, E.; Dienel, M.; Herrmann, G.; Müller, D.; Heinkel, U.: Modeling of a new 2D Acceleration Sensor Array using SystemC-AMS. International MEMS Conference 2006, Singapur, 9.-12.5. 2006