

Simulationsgestützter Entwurf von Mess- und Prüfsystemen für Mikrosysteme

Peter Schneider, Uwe Eichler, Karsten Einwich, Peter Schwarz

Fraunhofer-Institut für Integrierte Schaltungen, Außenstelle Entwurfsautomatisierung EAS Dresden,
Zeunerstraße 38, 01069 Dresden

E-Mail: Peter.Schneider@eas.iis.fraunhofer.de

Kurzfassung

Die Entwicklung neuer Mess- und Prüfsysteme kann durch rechnergestützte Entwurfsverfahren, vor allem durch Simulation und Optimierung, erheblich effektiviert werden. Die gemeinsame Modellierung von Messobjekt, Messgerät und (ggf. adaptiven) Signalverarbeitungsalgorithmen ermöglicht eine Gesamtsystemsimulation. Ein verwandte Methode wird in der Elektronik als virtueller Test bezeichnet. Der Einsatz von Werkzeugen wie MATLAB/SIMULINK und SystemC-AMS wird beschrieben.

1 Einleitung

Die Qualitätssicherung im Produktionsprozess mikrosystemtechnischer und mechatronischer Komponenten gewinnt wegen steigender Anforderungen an Produktqualität und Lebensdauer sehr stark an Bedeutung. Bei der Entwicklung von *Mess- und Prüftechnik* für diese Bauteile bestehen besondere Anforderungen. So werden die zu vermessenden Komponenten ständig weiterentwickelt, die zu untersuchenden physikalischen Phänomene immer komplexer, neue Wirkprinzipien und Materialien kommen zur Anwendung und die Bauteile werden funktions- und gewichtsoptimiert sowie weiter miniaturisiert. Zur messtechnischen Erfassung der Bauteilparameter steht immer weniger Zeit zur Verfügung. Außerdem muss die Entwicklung von Mess- und Prüftechnik bereits *parallel* zur Entwicklung des zu vermessenden Bauteils und der Produktionsanlagen erfolgen ("time-to-

market" verringern!). Die zeitige Verfügbarkeit präziser und leistungsfähiger Prüftechnik für die Qualitätssicherung ist somit von essentieller Bedeutung für die schnelle Markteinführung neuer mikrosystemtechnischer Produkte. Der Einsatz rechnergestützter Entwurfsmethoden und -werkzeuge ist eine wichtige Methode, dieses Ziel zu erreichen.

2 Systemsimulation von Mess- und Prüfsystemen

Bei der Entwicklung innovativer Mess- und Prüftechnik werden häufig mikrosystemtechnische Sensoren in Kombination mit intelligenter Signalverarbeitung eingesetzt. Leistungsfähige Mikroprozessoren und digitale Signalprozessoren erlauben es, kostengünstige Sensorik für die Messwert erfassung zu verwenden und diese durch adaptive Signalverarbeitungsalgorithmen zu ergänzen.

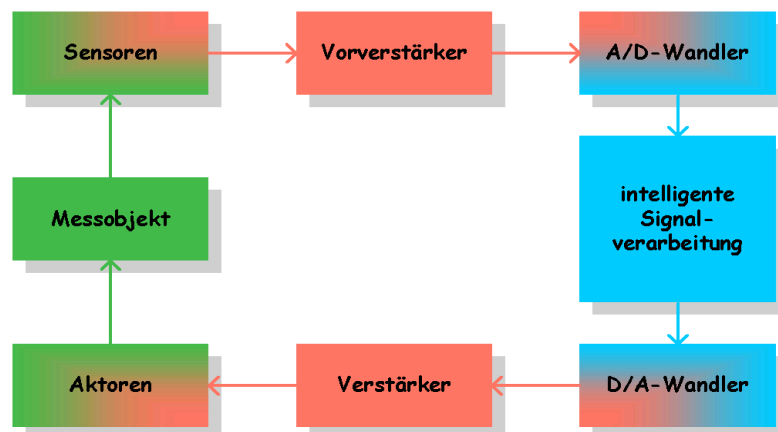


Bild 1: Typische Struktur von Mess- und Prüfsystemen

Die Integration von Sensorik und intelligenter Signalverarbeitung in Mess- und Prüfgeräten stellt aufgrund des komplexen Aufbaus des entstehenden Gesamtsystems hohe Anforderungen an die Entwurfsmethodik und an die verwendeten Verfahren. Bild 1 zeigt die allgemeine Struktur von Mess- und Prüfsystemen. Für den Entwurf von Signalverarbeitungsalgorithmen werden Daten benötigt, mit denen die Erprobung und Optimierung der Algorithmen erfolgt. Ein typisches Beispiel sind Neuronale Netze und deren Training durch umfangreiche Datensätze. Diese Daten müssen durch Messungen gewonnen werden.

Gerade bei adaptiven Algorithmen oder wenn über Aktoren auf den Prüfling Einfluss genommen wird (z.B. ein Regelkreis zum Ausgleich von Alterungsvorgängen), erweist sich dieser „klassische“ Ansatz als nicht effektiv genug. Viele Eigenschaften des Messobjektes und bestimmte Systemzustände können nicht oder nur mit sehr hohem zusätzlichen Aufwand für die Fertigung neuer Testobjekte oder von Sensoren mit geänderten Eigenschaften sowie durch umfangreiche Messreihen erfasst werden. Diese Nachteile können durch den Einsatz von *Simulationen* im Entwurf des Messsystems verringert werden. Im Bild 2 ist der Übergang vom realen Messsystem zu einem simulationsfähigen Modell skizziert. Die Grundidee besteht darin, das *gesamte Messsystem* (den Prüfling, die Sensorik, die intelligente Signalverarbeitung sowie die Prüfsignale und die Aktoren in ihrem Zusammenspiel) in *Simulationsexperimenten* zu untersuchen. Dieser Ansatz bietet verschiedene Vorteile:

- Konzeptstudien zu Varianten des Messsystems lassen sich frühzeitig und kostengünstig durchführen,

- mit der Entwicklung von Messverfahren kann schon *vor* dem Vorliegen erster Muster des Prüflings begonnen werden,
- Parameter des Prüflings können im Simulationsexperiment gezielt durch Parametervariationen geändert werden - ein großer Teil des Aufwandes für die Herstellung verschiedener Prüflinge und für messtechnische Studien kann somit entfallen,
- adaptive Signalverarbeitungsalgorithmen können am Simulationsmodell entworfen, trainiert und erprobt werden.

In der Mikroelektronik ist dieses Vorgehen seit einigen Jahren als *virtueller Test* eingeführt worden [EST98], [KRE00] und dient dort vor allem der Korrektur (debugging) und der Verbesserung der Testprogramme bereits vor dem Vorliegenden der ersten Fertigungsmuster neuentworfener Schaltkreise. Dadurch konnte "time-to-market" verkürzt und die Blockierung teurer Fertigungstester beim Einfahren der Testprogramme verringert werden.

Voraussetzung für die Durchführung dieser Gesamtsystemsimulation ist die Verfügbarkeit hinreichend genauer *Modelle* aller Komponenten sowie eines *einheitlichen Simulationsansatzes* für die z.T. sehr unterschiedlichen Teilsysteme:

- Prüfling, Sensoren und Aktoren mit gekoppelten physikalische Effekten,
- analoge Elektronik für Signalaufbereitung und Aktoransteuerung,
- digitale Elektronik für Signalvorverarbeitung und Steuerungsaufgaben,
- Software zur Realisierung adaptiver Signalverarbeitungsalgorithmen.

Eng verknüpft ist die *Gesamtsystemsimulation* bei mechatronischen Systemen immer mit der Simula-

Messung, Test

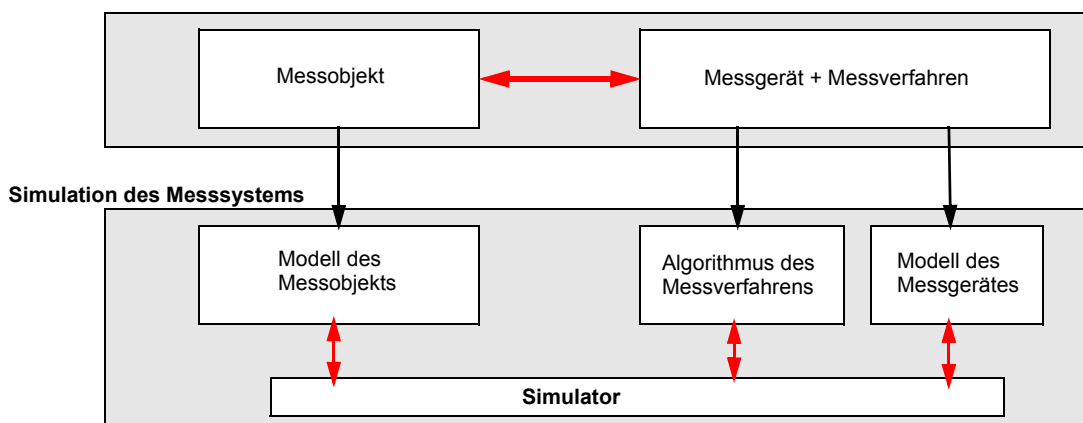


Bild 2: Rechnergestützter Entwurf von Messverfahren, virtueller Test

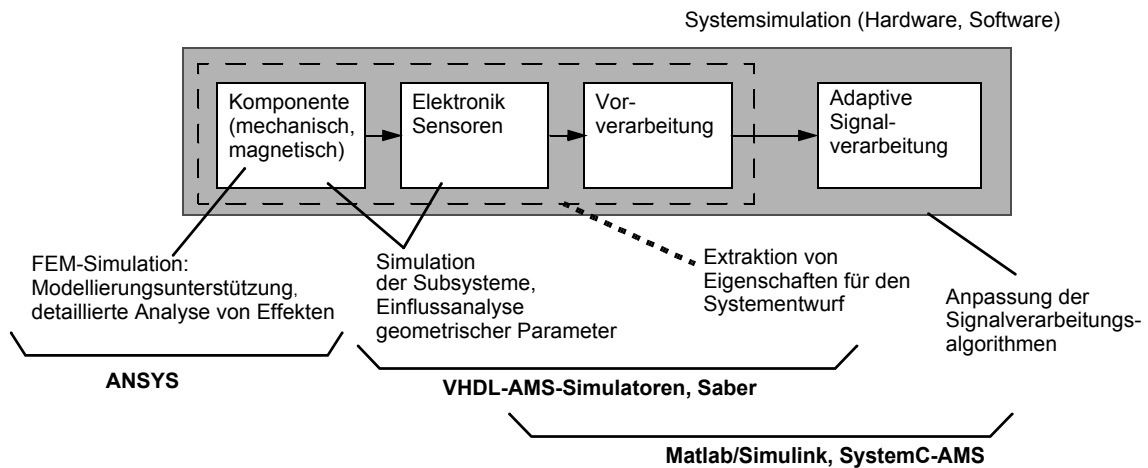


Bild 3: Einsatz von Simulationswerkzeugen beim Entwurf mechatronischer Systeme

tion auf *Komponentenebene* (Bild 3). Einerseits müssen für den Systementwurf Detailuntersuchungen zur Dimensionierung der Komponenten durchgeführt werden, andererseits sind für die Modellbildung auf Systemebene detaillierte Simulationen von Komponenten und Teilsystemen notwendig. Daher kommen unterschiedliche Simulatoren und ihre Modellierungssprachen zum Einsatz und es entstehen Schnittstellenprobleme. Ein weiteres Ziel besteht also darin, mit möglichst wenig Simulatoren und Sprachen auszukommen.

Nachfolgend wird anhand zweier Beispiele auf unterschiedliche Aspekte der Gesamtsystemsimulation von Mess- und Prüfsystemen eingegangen.

3 Systemsimulation mit MATLAB/SIMULINK

Der Einsatz von MATLAB/SIMULINK für die Systemspezifikation und den Systementwurf ist so weit verbreitet, dass er hier nur sehr kurz und in Verbindung mit dem rechnergestützten Entwurf der Messverfahren erwähnt werden muss. Signalverarbeitungs- und Regelalgorithmen können sehr effizient entworfen und auch mit Modellen anderer Systemkomponenten kombiniert werden. Schwierigkeiten ergeben sich bei diesem Ansatz beim Übergang auf die unteren Entwurfsebenen und bei der Einbindung von Schaltungsbeschreibungen aus der Elektronik. Probleme für den Anwender treten bei der Modellierung von Systemen auf, die durch *implizite* nichtlineare Differentialgleichungssysteme beschrieben werden müssen und die schwierig von Hand aufzustellen sind. Der Einsatz von MATLAB/SIMULINK wird am Beispiel des Entwurfs von Signalverarbeitungsalgorithmen für Kalibriersysteme beschrieben.

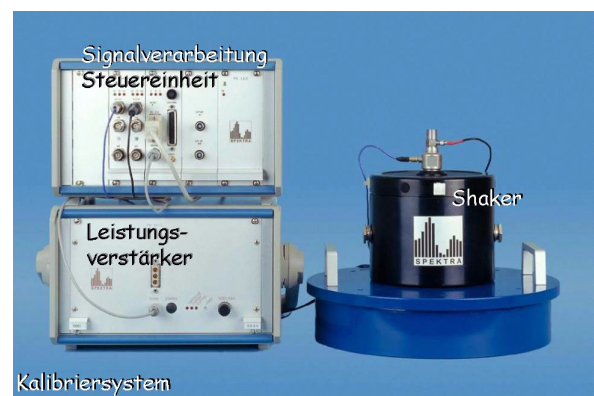


Bild 4: Kalibriersystem für Beschleunigungssensoren

Das in Bild 4 gezeigte Kalibriersystem dient der Untersuchung von Beschleunigungssensoren durch die direkte Messung der Beschleunigung (Primärkalibrierung) oder durch Vergleich mit einem Normal (Sekundärkalibrierung). Das System besteht aus einem elektrodynamischen Schwingungserzeuger (Shaker), einer Steuereinheit, einem Leistungsverstärker und den Sensoren, die auf dem Schwingtisch des Shakers montiert sind. Der Shaker wird durch die Ansteuereinheit mit dem für die jeweilige Messaufgabe vorgesehenen elektrischen Signal angesteuert. Nichtlinearitäten im Schwingungserreger verursachen Abweichungen des Beschleunigungssignals vom benötigten Prüfsignal. Bei Kalibrieraufgaben mit Sinussignalen äußert sich dies z.B. in nichtlinearen Verzerrungen des mechanischen Beschleunigungssignal bei sinusförmiger elektrischer Anregung. Durch Modifikation ("Vorverzerrung") des elektrischen Ansteuersignals lassen sich diese Nichtlinearitäten ausgleichen ("Entklirrung") [TKN03], [TSK04].

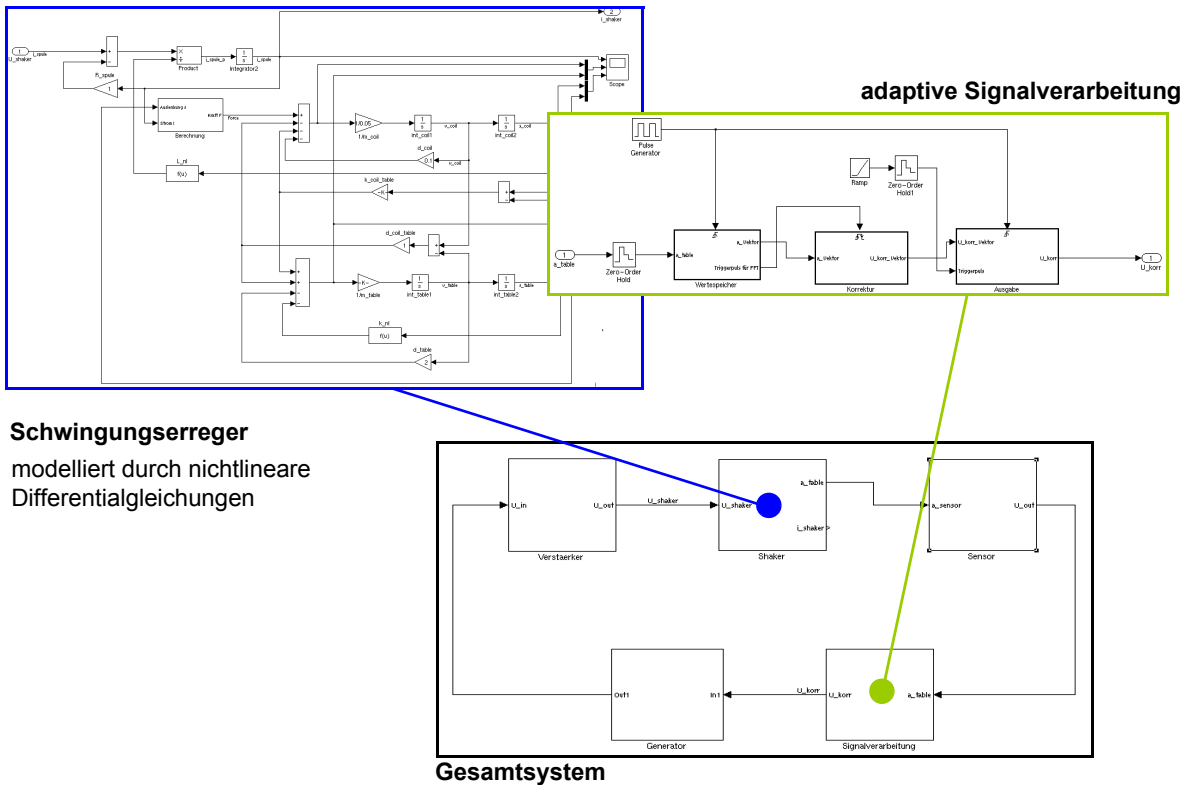


Bild 5: Systemmodellierung - Modelle des Gesamtsystems, des elektrodynamischen Schwingererreggers und der Signalverarbeitung

Für die Entwicklung der dafür benötigten Algorithmen stellt die Gesamtsystemsimulation ein effizientes Hilfsmittel dar, das es erlaubt:

- algorithmische Konzepte für adaptive Signalverarbeitung zu entwickeln und zu erproben,
- Parameter von wichtigen Systemkomponenten (z.B. des Shakers) systematisch zu variieren,
- die automatische Adaption an das jeweilige Messobjekt und an die zeitlichen Veränderung von Systemparametern (z.B. durch Alterung) im späteren Betrieb nachzuweisen.

Natürlich erfolgen die endgültige Erprobung und die Feinanpassung der Algorithmen am realen System und nicht mehr im Simulationsexperiment. Bild 5 zeigt das Gesamtsystemmodell. In den dargestellten Blöcken sind wiederum detailliertere Modelle für alle Komponenten des Systems enthalten, die wesentlichen Einfluß auf den Entwurf der Algorithmen haben. Unterstützt durch Simulationen konnten für verschiedene standardisierte Prüfsignale (Sinus, Halbsinus-Stoß, Trapez- und Sägezahn-Stoß sowie Multi-Sinusanregung) Algorithmen für das Kalibriersystem entwickelt werden, die den standardkonformen Signalverlauf der Beschleunigung am Schwingkopf des Shakers garantieren [TSK04].

4 Systemsimulation mit SystemC-AMS

Eine weitere Herausforderung bei der Gesamtsystemsimulation von Mess- und Prüfsystemen entsteht, wenn der *Softwareanteil* (z.B. in Form von C-Programmen) sehr hoch ist und daher gemeinsam mit der Hardware untersucht werden muss. Die Einbeziehung derartiger Softwarekomponenten in die üblichen System simulationsprogramme ist grundsätzlich möglich, oft aber mit erheblichem Effizienzverlust bei der Simulation verbunden. Die gerade bei adaptiven Algorithmen notwendigen umfangreichen Simulationsexperimente erfordern daher neue Simulationsansätze.

4.1 Die Modellierungssprachen SystemC und SystemC-AMS

SystemC ist eine in den letzten Jahren entwickelte und inzwischen sehr populäre Beschreibungssprache und Methodik zur Modellierung und Simulation komplexer *digitaler Hard- und Softwaresysteme* (z.B. der Telekommunikation) auf unterschiedlichen Abstraktionsebenen. Sie ist vor allem auf System-Level-Modellierung ausgerichtet, bietet aber auch alle von anderen Hardware-Beschreibungssprachen wie VHDL und Verilog bekannten Konzepte [MRR03], [BES03]. Sprach-

elemente von SystemC, z.B. zur Beschreibung von Verhalten und Struktur durch Module, Ports und Signale, basieren auf C++. Damit sind alle üblichen C++ Entwicklungswerkzeuge und Bibliotheken verwendbar und die Einbeziehung von in C oder C++ programmierter Software ist sehr einfach möglich. Zur Simulation von SystemC-Beschreibungen ist eine C++ Bibliothek erforderlich, welche die Sprachelemente implementiert und u.a. auch den Simulorkern enthält. Diese Bibliothek existiert als freie Referenzimplementierung und in verschiedenen kommerziellen Werkzeugen. Die Hardware-Software-Cosimulation kann daher sehr effizient durchgeführt werden.

Für den Einsatz beim Entwurf mechatronischer Systeme und ihrer Mess- und Prüfeinrichtungen sind jedoch Erweiterungen für die Behandlung analoger und gemischt analog/digitaler Komponenten notwendig. Diese Erweiterungen führen zu **SystemC-AMS** (Analog and Mixed-Signal Extension) [ECN01], [MRR03] und werden in einer SystemC Study Group betrieben.

SystemC-AMS soll in drei Entwicklungsphasen eingeführt werden. Phase 1 sieht Erweiterungen für analoge *lineare* dynamische Modelle vor. Das beinhaltet Operatoren für die Modellbeschreibung mit Übertragungsfunktionen oder Zustandsraumgleichungen, lineare Netzwerkelemente (R, L, C, Quellen) sowie eine Synchronisationsschicht zwischen analogen und digitalen Modellen auf Basis einer statischen Datenfluss-Steuerung mit festen Zeitschritten. In Phase 2 soll Unterstützung für *nichtlineare* Differentialgleichungen, variable Zeitschritte und Frequenzbereichs-Simulation hinzukommen. Die *allgemeine Netzwerkanalyse* (z.B. Knotenspannungsanalyse) zur Berechnung von Strömen und Spannungen in nichtlinearen Schaltungen und ein generisches Synchronisationsmodell sind in Phase 3 vorgesehen.

Am Fraunhofer IIS / EAS Dresden, das maßgeblich an den SystemC-AMS-Aktivitäten beteiligt ist, existiert bereits eine proprietäre Implementierung namens MixSigC [ECN01], die in etwa den Funktionsumfang von Phase 1 (und zusätzlich die Frequenzbereichs-Simulation) umfasst.

4.2 Messverfahren für Mikrorelais

Relais werden als elektrische Schalter auch in sicherheitskritischen Systemen wie z.B. Antiblockiersystemen verwendet. Das erfordert die Messung bestimmter Funktionsparameter an *jedem einzelnen* Relais in der Produktionsendprüfung.

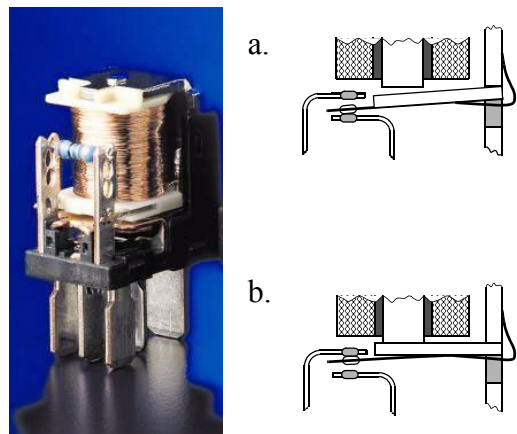


Bild 6: Aufbau eines Mikrorelais, Schaltzustände a) "Aus" und b) "An"

Die für die Qualitätskontrolle erforderliche Messung von Kontaktkräften, Kontaktweg und Abbrandreserve ("Zielgrößen") erfolgt derzeit mechanisch. Dazu müssen die beweglichen Teile des Relais für Sensoren zugänglich sein. Dies ist durch die zunehmende Miniaturisierung oft nicht mehr der Fall. Ausserdem lassen sich mit mechanischen Verfahren die gewünschten Taktzeiten nur schwer erreichen. Daher wird nach alternativen Messverfahren gesucht, die die mechanischen Größen *indirekt* messen. Für Messungen am Relais bieten sich Strom und Spannung der Magnetspule des Relais an, da diese leicht und mit großer Genauigkeit gemessen werden können. Da sich die *mechanischen* Vorgänge während des Schaltens über entsprechende Induktionsspannungen auch *elektrisch* auswirken, kann aus der Kenntnis des Induktionsspannungsverlaufes bei einem bekannten Erregerstrom auf die mechanischen Eigenschaften des Relais geschlossen werden. Das entwickelte Verfahren [SWS04] umfasst also drei Stufen:

1. Messungen des Strom/Spannungsverlaufes an der Erregerspule des Relais,
2. Berechnung geeigneter physikalischer Größen und Extraktion von charakteristischen Werten aus diesen Verläufen,
3. Auswertung der Kenngrößen, um die gewünschte Qualitätsaussage zu erhalten.

Für die Kenngrößenauswertung wurde ein *neuronales Netz* gewählt. Für das *Training* des neuronalen Netzes werden Datensätze benötigt, die die verschiedenen geometrischen und sonstigen Parameter, die Zeitverläufe von Strom und Spannung am Relais und die gesuchten Zielgrößen enthalten. Es ist jedoch sehr schwierig und aufwändig, solche

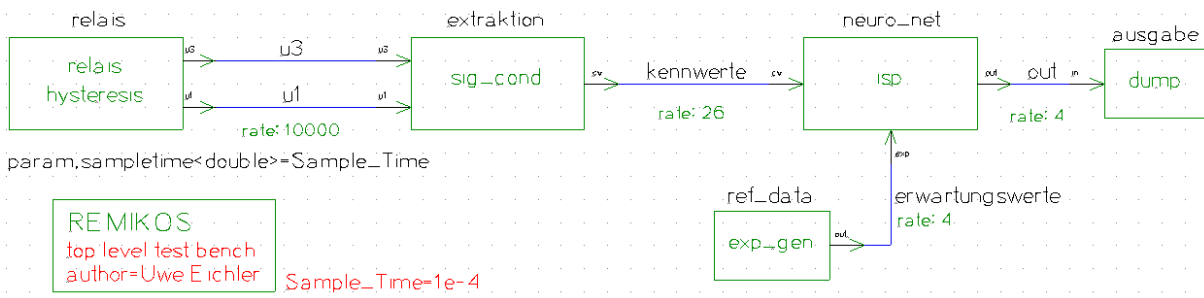


Bild 7: Gesamtmodell des Messsystems

Messungen in der erforderlichen Anzahl durchzuführen. Ausserdem ist es kaum möglich, definiert Relais mit unterschiedlichen Fehlern herzustellen. Daher erfolgt die Erzeugung der Trainingsdaten durch *Modellierung* des Relais mit einem Netzwerkmodell und anschließende *Simulation*.

4.3 Modellierung mit SystemC-AMS

Für die Erprobung und Optimierung anderer Komponenten des Messsystems bietet sich ebenfalls der Einsatz von Systemsimulationen an. Die Schwierigkeit besteht darin, gemeinsam mit den physikalischen Phänomenen am Relais auch die *Signalvorverarbeitung* und die adaptive *Signalverarbeitung* durch das Neuronale Netz - die beide durch sehr umfangreiche C-Programme realisiert werden! - zu simulieren. Dieses Problem war Anlass, das Gesamtsystem (Hardware + Software) vollständig in SystemC-AMS zu modellieren.

Ein solches Gesamtmodell wurde auf Basis der proprietären SystemC-AMS-Implementierung "MixSigC" erstellt und gliedert sich in die drei bereits genannten Hauptmodule Relaismodell (relais), Signalvorverarbeitung (sig_cond) und intelligente Signalverarbeitung mittels Neuronalem Netz (isp) (siehe Bild 7).

Das Modul sig_cond berechnet jeweils aus einer festen Anzahl von Abtastwerten des Strom- und Spannungsverlaufes an der Relaispule 30 charakteristische Kennwerte dieser Hysteresekennlinie, welche wiederum an das Modul isp zur Ermittlung der vier Ergebnisgrößen Arbeitskontaktkraft, Rückanlagekontaktkraft, Kontaktabstand und Abbrandreserve weitergegeben werden.

Das Relaismodell enthält Modelle der elektrischen, magnetischen und mechanischen Komponenten, die auf elektrische Netzwerkelemente abgebildet wurden (Bild 8). Eine besondere Schwierigkeit lag in der Modellierung der *nichtlinearen* mechanischen und magnetischen Effekte des Relais mit der nur für *lineare* Modelle vorgesehenen MixSigC-Bibliothek. Eine Lösung lag darin, das jeweilige Modell in einen linearen und einen nichtlinearen Anteil zu zerlegen und den nichtlinearen Anteil mit einer Verzögerung von einem Zeitschritt zu berechnen. Um dadurch entstehende Instabilitäten zu vermeiden, musste darauf geachtet werden, den linearen - und damit unverzögerten - Anteil am Ausgangssignal groß gegenüber dem nichtlinearen zu halten und möglichst kleine konstante Zeitschritte zu verwenden. Auf Grund der durch diese Einschränkungen möglichen Verwendung sehr einfacher Integrationsal-

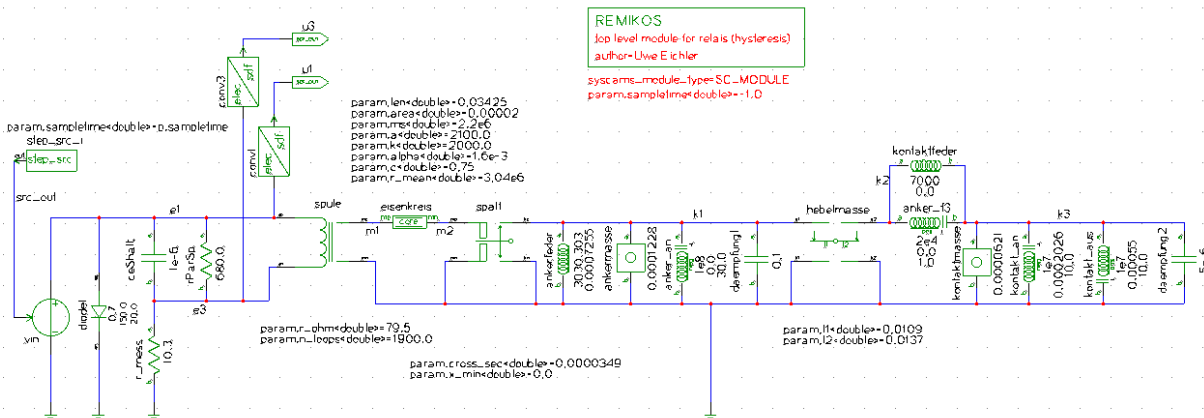


Bild 8: SystemC-AMS-Modell des Mikrorelais (Messobjekt)

gorithmen kann dieser Weg in Abhängigkeit von den konkreten Systemeigenschaften einen erheblichen Performancegewinn gegenüber allgemeinen nichtlinearen Simulatoren, wie z. B. für VHDL-AMS, bringen. Dieser Gewinn wird allerdings mit erhöhtem Modellierungsaufwand und etwas verringertem Genauigkeit erkauft.

Elektrische Netzwerkelemente werden in MixSigC durch Differentialgleichungen der Form

$$A(t) \cdot dx/dt + B(t) \cdot x + q(t) = 0$$

dargestellt. Im Modell müssen dazu in der Funktion `matrix_entries()` die Matrizen A , B und q entsprechend ihrer Abhängigkeit vom Strom $i(t)$ belegt werden. Als Beispiel ist der Quellcode des Eisenkernmodells angegeben:

```
// calculates nonlinear part of phi (current)
// (Jiles/Atherton core model)
double mag_core::i_t(double time)
{
    double hl;
    if (time > last_time)
    {
        last_time = time;
        // voltage between ports mp, mn at last
        // sample time
        hl = solver->x[node0] - solver->x[node1];
        h = hl/len;
        double dh = (h-h_last) / solver->get_H();
        h_last = h;
        b = phi_last/area;
        double M = b / MU0 - h;
        double dhs = dh<0.0 ? -1.0 : 1.0;
        double mh = mymanh(h+alpha*M);
        dmirr = mymigrating(dhs, M, mh) *
            (mh-M) / (dhs*k-alpha*(mh-M))*dh;
        dmirr_int += (dmirr+dmirr_last)/2.0 *
            solver->get_H();
        dmirr_last = dmirr;
        phi_last = area*MU0*(h+(1.0-c)*
            dmirr_int+c*mh);
    }
    // subtract the linear part of the current
    // which accounts for the resistance r_mean
    return phi_last-hl/r_mean;
}
// run in initialization phase
void mag_core::matrix_entries(
    long *nodes,
    fhg_matrix_sq<double>& A,
    fhg_matrix_sq<double>& B,
    fnct_vector& q )
{
    node0 = nodes[0];
    node1 = nodes[1];

    // set resistor matrix entries
    B(node0,node0) += 1.0/r_mean;
    B(node0,node1) += -1.0/r_mean;
    B(node1,node0) += -1.0/r_mean;
    B(node1,node1) += 1.0/r_mean;

    // add nonlinear distortions, see i_t() above
    q(node0).add(&elec_elements::i_t,this);
    q(node1).sub(&elec_elements::i_t,this);
}
```

Es besteht aus einem linearen Teil, der eine Widerstandskennlinie repräsentiert und durch Belegung der B-Matrix zustande kommt. Der nichtlineare Teil berechnet mit Hilfe der `i_t()`-Methode die zur Hysterese führenden Abweichungen des Stromes von dieser Kennlinie. Der Rückgabewert von `i_t()` wird jeweils als Element der q-Matrix eingesetzt und somit zum resultierenden Strom addiert. In Bild 9 sind die Ergebnisse der Simulation eines Anzugvorganges für das Mikrorelais dargestellt. Für ein Dreiecksignal der Spulenspannung sind der resultierende Spulenstrom, der magnetische Fluss im Eisenkreis sowie die Wege am Relaisanker und am Kontakt dargestellt.

5 Zusammenfassung

Durch den Einsatz von Modellierung und Simulation bei der Entwicklung von Mess- und Prüfsystemen kann der Entwurfsprozess effektiviert und beschleunigt werden. Der nötige Aufwand für die Modellierung der Teilsysteme wird durch die Möglichkeiten zur virtuellen Erprobung von algorithmischen Konzepten und zur Erprobung und Bewertung von Messverfahren ausgeglichen. Für die Gesamtsystemsimulation von mechatronischen Systemen sind verschiedene Ansätze verfügbar, wobei aber insbesondere hinsichtlich der Ankopplung an Ansätze und Beschreibungsmittel aus dem Komponentenentwurf noch Erweiterungsbedarf besteht.

Ein Teil der diesem Beitrag zugrunde liegenden Arbeiten wurde durch das BMBF im Rahmen des Projektes REMIKOS (Rechnergestützte Entwicklung von Messverfahren durch Integration mikro-systemtechnischer Komponenten und intelligenter Signalverarbeitung) unter dem Kennzeichen 16 S V1415 gefördert. Die Verantwortung für den Inhalt der Veröffentlichung liegt allein bei den Autoren.

6 Literatur

- [BES03] Braun, A., Einwich, K.; Speitel, M.: *SystemC – Quo Vadis?* Elektronik, 03/2004
- [ECN01] Einwich, K.; Clauss, C.; Noessing, G.; Schwarz, P.; Zojer, H.: *SystemC Extensions for Mixed-Signal System Design*. Proc. FDL'01, Lyon, France, September 2001
- [EST98] Einwich, K.; Schwarz, P.; Trappe, P.; Chambers, T.; Krampfl, G.; Zojer, H.; Sattler, S.: *Virtual Test of Complex Mixed-Signal Telecommunication Cir-*

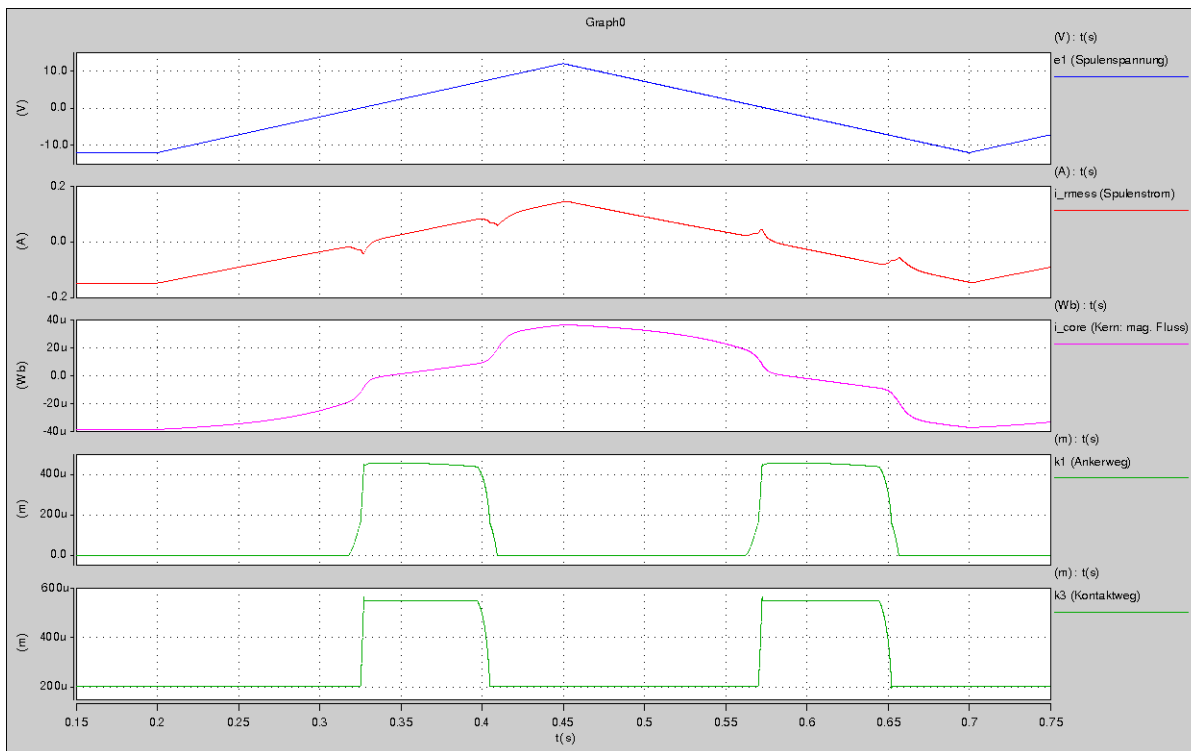


Bild 9: Simulationsergebnisse

cuits reusing System-Level Models.
4th IEEE International Mixed-Signal
Testing Workshop, The Hague, 1998,
237-242

- [KEQ94] Kallenbach, E.; Eick, R.; Quendt, P.: *Elektromagnete*. Teubner, Stuttgart, 1994
- [Kor91] Korn, G. A.: *Neural Networks Experiments on Personal Computers and Workstations*. MIT Press, Cambridge (Massachusetts), 1991
- [KRE00] Krampfl, G.; Rona, M.; Einwich, K.: *Model and Simulator Coupling for Virtual Test*. 6th IEEE International Mixed-Signal Testing Workshop. Montpellier, France, June 21-23, 2000
- [Mül99] Müller, M.: *Anwendung Neuronaler Netze in der Signalverarbeitung*. Diplomarbeit, HTW Dresden, 1999
- [ScS99] Schäfer, E.; Schneider, P.: *Modellierung und Simulation von Mikrorelais*. Proc. 7. Workshop „Methoden- und Werkzeugentwicklung für den Mikrosystementwurf“, Paderborn, 21./22. Januar 1999
- [SSH99] Schneider, P.; Schwarz, P.; Huck, E. u.a.: *Simulation komplexer Systeme unter Einbeziehung Neuronaler Netze*.

Proc. 7. Workshop „Methoden- und Werkzeugentwicklung für den Mikrosystementwurf“, Paderborn 1999, 159-168

- [SWS04] Schneider, P.; Wilde, A.; Schäfer, E.; Sommer, K.: *Innovative Messverfahren für miniaturisierte elektromagnetische Komponenten*. Diese Tagung.
- [TKN03] Trappe, P.; Kurth, G.; Nicklich, H.; Bock, V.; Bühn, U.; Schneider, P.: *Simulationsgestützter Entwurf von Signalverarbeitungs-Algorithmen für Kalibriersysteme*. 9.GMM-Workshop "Methoden und Werkzeuge zum Entwurf von Mikrosystemen", Paderborn 2003, 203-209
- [TSK04] Trappe, P.; Schneider, P.; Kurth, G.; Wielens, S.; Nicklich, H.; Bock, V.: *Modellierung und Simulation im Entwurf von Signalverarbeitungs-Algorithmen für Schwingprüf- und Kalibriersysteme*. Diese Tagung.