

OSEK-System der Zukunft

Mit dem Ziel, einen einheitlichen Standard als Basis für die Software der elektronischen Steuergeräte zu etablieren, wurde in den 1990er Jahren von den großen Fahrzeugherstellern das Betriebssystem OSEK/VDX (Offene Systeme und deren Schnittstellen für die Elektronik im Kraftfahrzeug) eingeführt. Die Vielfalt proprietärer Embedded-Betriebssysteme bei den zahlreichen Zulieferern erwies sich angesichts der zunehmenden Bedeutung der Kraftfahrzeugelektronik als wachsendes Hindernis für eine reibungslose Integration. Neben dem eigentlichen Betriebssystemkern definiert OSEK deshalb auch Kommunikationsdienste und Funktionen für das Netzwerkmanagement.

Hoher Einführungsaufwand macht sich bezahlt

Da sich die meisten Automobilzulieferer zuvor bereits auf ein jeweils bevorzugtes Betriebssystem festgelegt hatten, mussten die Fahrzeughersteller OSEK teils mit Nachdruck einführen. DaimlerChrysler z.B. setzte OSEK sowohl für In-house-Entwicklungen als auch für Zulieferer zwingend als Standard für Neuentwicklungen voraus. Das Unternehmen organisierte OSEK-Trainings, erstellte OSEK-Design-Guides und unterstützte Betriebssystemhersteller; pro Zulieferer wurde eine OSEK-Lizenz finanziert, wobei nur zertifizierte OSEK-Betriebssysteme erlaubt waren. Der Einführungsaufwand erreichte im Jahr 2002 seinen Höhepunkt, fiel danach deutlich zurück. Inzwischen ist OSEK ein Selbstläufer geworden, so dass die meisten Steuergeräte im Bereich der Personenkraftwagen mit OSEK-Betriebssystemen laufen.

Der Aufwand hat sich bezahlt gemacht: Die Applikationen auf Basis von OSEK erfüllen die in sie gestellten Erwartungen durch verbesserte Code-Qualität, Strukturierung und Integrationsmöglichkeiten der Komponenten verschiedener Zulieferer.

Im Fokus: Timing, Speicherschutz und Fehlererkennung

Das Echtzeit- und Multitasking-Betriebssystem OSEK ist heute Standard bei Automotive-Embedded-Entwicklungen. Zu den wichtigsten Eigenschaften zählen der geringe Verbrauch an Prozessorressourcen und Speicher sowie eine ereignisgesteuerte Taskverwaltung, die sowohl zyklischen als auch azyklischen Programmblöcken gerecht wird. Mit der Weiterentwicklung der Automobilelektronik und den neuen Standardisierungsinitiativen HIS und AUTOSAR stellen sich auch an das Betriebssystem neue Forderungen bezüglich Timing und Speicherschutz.

Von Peter Liebscher

Bei dem Gateway-Hersteller K2L haben sich Lösungen mit „osCAN“, dem OSEK/VDX-konformen Betriebssystem von Vector Informatik, aufgrund kurzer Reaktionszeiten und eines genauen Timings bewährt. Ausschlaggebend für OSEK in Verbindung mit einem tabellengesteuerten Interpreter-Konzept sind nicht zuletzt die gewonnene Flexibilität in Richtung kurzer Entwicklungszeiten sowie einfacher Variantenerstellung. Vorausgegangen waren dabei Vergleiche von OSEK mit seinem pre-emptiven Scheduling und einem fest codierten statischen Ansatz mit kooperativem Scheduling. In Bild 1 sind die Gateway-, System- und Bus-APIs des K2L-Mul-

ti-CAN-/MOST-Gateway dargestellt. Die OSEK-Implementierung „osCAN“ bewährt sich nicht nur in Steuergeräten, sondern auch in der Schnittstellen-Hardware desselben Unternehmens. In dem MOST-Interface VN2600 stellt osCAN seine Leistungsfähigkeit unter einem 133-MHz-Altera-Excalibur-Controller unter Beweis (Bild 2): Die Verarbeitung von bis zu 35 000 Events/s entspricht einem Datendurchsatz von 1,7 Mbyte/s.

HIS und AUTOSAR definieren Betriebssysteme der Zukunft

Die Anforderungen an das Betriebssystem steigen parallel mit der weiteren

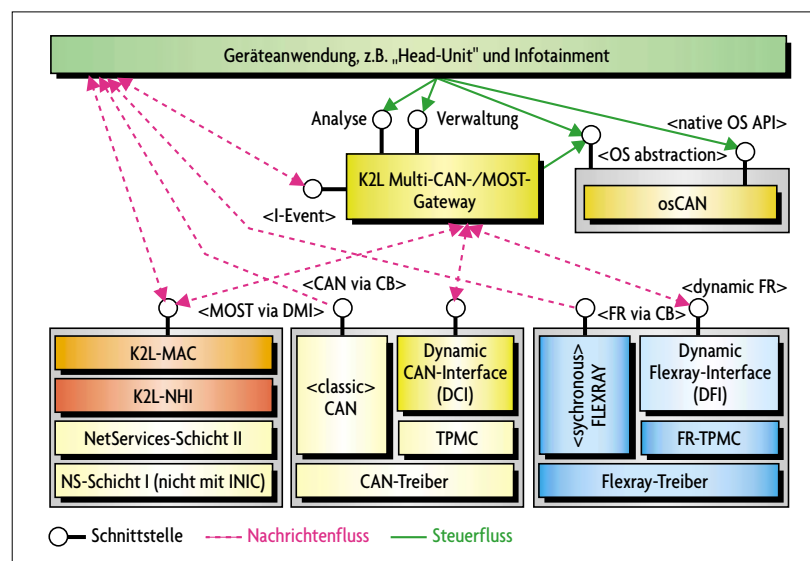


Bild 1. Darstellung der Gateway-, System- und Bus-APIs.

(Quelle: K2L GmbH/Vector Embedded Symposium)

Durchdringung der Technik mit Elektronik. Insbesondere der Vormarsch sicherheitsrelevanter Anwendungen im Fahrzeug, wie z.B. vollelektronisch gesteuerte Lenk- und Bremssysteme (X-by-Wire), machen ein deterministisches Verhalten auch unter Spitzenlast und Fehlerbedingungen unabdingbar. So wird durch die Spezifizierung von „OSEKtime“ das ereignisgesteuerte OSEK durch eine zeitgesteuerte Variante ergänzt.

Fehlertoleranz, Fehlererkennungsmechanismen und Speicherschutz spielen darüber hinaus eine wichtige Rolle für die Systemzuverlässigkeit. Diese Aspekte gewinnen dadurch besondere Relevanz, weil künftig ein Steuergerät für mehrere gleichzeitig laufende Applikationen zur Verfügung steht. In der „Hersteller Initiative Software“ (HIS) verständigen sich die großen deutschen Automobilhersteller über die notwendigen Standards zur Realisierung der genannten Funktionen. Es gibt Arbeitskreise für die Bereiche Software, Software-Test, Process Assessment, Simulation und Tools sowie Flash-Programmierung. Für die neuesten Standards der künftigen Fahrzeuggenerationen ist das AUTOSAR-Konsortium (Automotive Open System Architecture) verantwortlich. Dabei bringt die HIS-Gruppe ihre Ergebnisse in AUTOSAR ein bzw. vertritt dort einheitliche Standpunkte.

■ Sicheres Multitasking für mehrere Applikationen

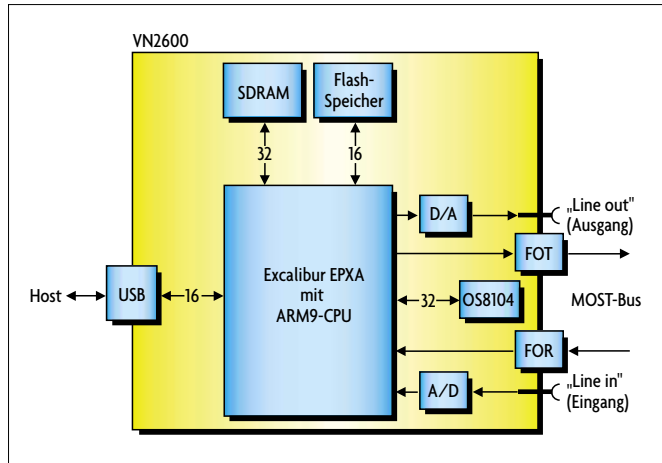
Wenn man bedenkt, dass heute über 50 Steuergeräte in einem Oberklasse-PKW arbeiten und für die Zukunft noch kein Ende neuer Einsatzmöglichkeiten für die Elektronik im Fahrzeug absehbar ist, wird die zahlenmäßige Begrenzung der Steuergeräte zu einem vordringlichen Thema. Dies lässt sich nur dadurch erreichen, dass mehrere Applikationen auf demselben Steuergerät laufen.

Das Multitasking-Betriebssystem OSEK wurde für diesen Zweck spezi-

fiziert. Für die Verwendung in sicherheitskritischen Systemen oder zur Integration von Software verschiedener Hersteller werden jedoch weitere Eigenschaften im Betriebssystem benötigt.

■ Vom „Deadline-Monitoring“ bis zur sicheren Fehlerdetektion

Zum Beispiel darf keine Applikation parallel laufende Applikationen stören. Um dies zu verhindern, konzentrieren sich die Neuerungen im Betriebssystem auf eine optimale Überwachung des Zeitverhaltens der einzelnen Tasks und auf durchgängigen Speicherschutz. Der Fortschritt dieser Bemühungen ist in den verschiedenen Entwicklungsstufen unterschiedlich weit vorangeschritten. Das seit dem Jahr 2001 für zeitgesteuerte Aufgaben spezifizierte OSEKtime bietet mit seinem „Deadline-Monitoring“ nur ansatzweise die in Zukunft erforderlichen Funktionen. Durch Deadline-Monitoring lässt sich detektieren, ob ein Task bis zu einem vorgegebenen Zeitpunkt rechtzeitig beendet ist. Leider kann das Verfahren jedoch bei Deadline-Verletzungen nicht die Ursachen hierfür ausmachen (Bild 3). Wird der überwachte Task z.B. durch einen Task mit höherer Priorität



■ Bild 2. Funktionale Blockschaltung mit dem MOST-Interface VN2600 und einem Altera-Excalibur-Controller.
(Quelle: Vector Embedded Symposium)

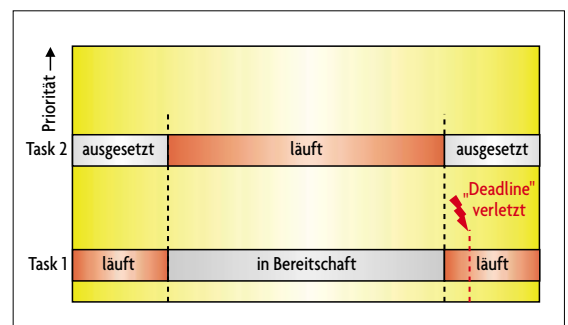
unterbrochen, trifft ihn letztlich keine Schuld, wenn er das vorgegebene Zeitfenster nicht einhalten kann.

In den HIS-konformen OSEK-Erweiterungen sind zusätzlich zum Deadline-Monitoring Speicherschutzfunktionen definiert. Am fortschrittlichsten ist AUTOSAR mit seinen Methoden des „Execution Time Enforcement“ und „Arrival Rate Enforcement“, das auch Tasks mit niedrigerer Priorität

ein Minimum an Zeitbudget zwingend zur Verfügung stellt; hier lassen sich Fehlerquellen eindeutig identifizieren. Vector Informatik hat in den entsprechenden Betriebssystem-Versionen darüber hinaus Möglichkeiten für Laufzeitmessungen integriert. Der osCAN TimingAnalyzer erlaubt die Simulation von Scheduling-Tabellen (Fahrplänen) und die Berechnung der Schedulability. Die Analyse wird mit den Informationen Priorität, Periode, Ausführungszeit und Deadline für jeden Task und jede ISR (Interrupt-Service-Routine) durchgeführt (Bild 4).

■ Speicherschutz: Trusted oder Non-Trusted

Speicherschutzfunktionen begrenzen den Zugriff eines Task auf den ihm zur Verfügung gestellten Speicherbereich. Dabei gilt es insbesondere, Schreibzugriffe auf fremde Datenssegmente zu verhindern, Stack-Überläufe zu entdecken, aber auch das Ausführen von



■ Bild 3. Deadline-Monitoring zur Erkennung von Deadline-Verletzungen. Die Fehlerursache des Task 2 wird hierbei in Task 1 nicht ausgemacht.
(Quelle: Vector Embedded Symposium)

falschem Code zu erkennen. Für Tasks, die zur selben Applikation gehören, braucht man andererseits Zugriffsmöglichkeiten auf dieselben Speicherbereiche (Bild 5). Spezielle Systemfunktionen wie z.B. Treiber könnten darüber hinaus aber völlig unbegrenzten Speicherzugriff benötigen.

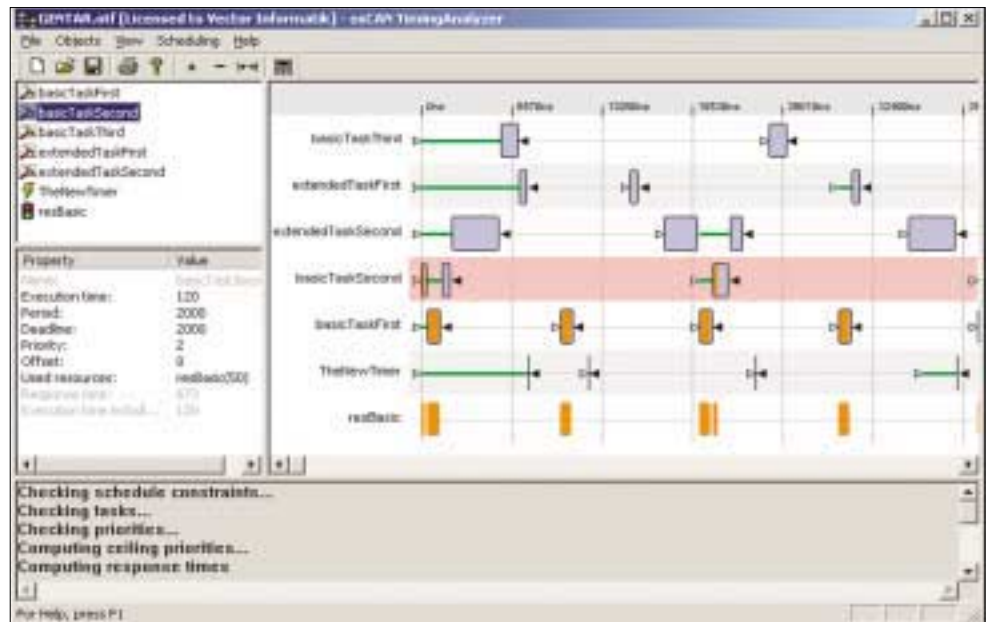
Man unterscheidet zwischen so genannten „Trusted Applications“ mit vollen Zugriffsrechten und „Non-trusted Applications“ mit begrenzten Zugriffsmöglichkeiten. Dabei kann die Namensgebung zu Verwirrung führen: Non-trusted Applications sind gesicherte Programme, die keinen Schaden anrichten können. Den Trusted Applications dagegen vertraut man quasi blind. Letztere lassen sich einfach verwenden, sind aber gefährlich für die Systemsicherheit und bieten keine Identifizierungsmöglichkeit für Fehler bzw. Fehlerquellen.

Gutes Design wird deshalb Funktionen, wo immer möglich, in Non-trusted Applications legen. Vector Informatik bietet daher Implementierungen an, die zusätzlich den Aufruf von Non-trusted Services erlauben. Sie sind prädestiniert für sicherheitskritische Anwendungen und bieten ein Maximum an Überwachung. Ein entsprechender Vorschlag, dies in AUTOSAR ebenso zu handhaben, wird zurzeit in einer Arbeitsgruppe diskutiert.

■ Bessere Unterstützung durch moderne Hardware-Plattformen

Nur mit entsprechender Hardware-Unterstützung sind die genannten Überwachungsfunktionen für Timing und Speicher effizient umsetzbar. Zum Speicherschutz braucht man „Memory Protection Units“ (MPUs), deren Verwaltungsmöglichkeiten hinsichtlich Zahl und Größe der Speicherblöcke auf die Bedürfnisse der Automotive-Anwendungen zugeschnitten sind. Vielfach lassen sich heute als kleinste Einheiten nur Blöcke mit 16 Kbyte verwalten. Im Automotive-Embedded-Bereich werden jedoch deutlich kleinere Speichereinheiten benötigt.

Grundsätzlich sind die Anforderungen aktueller und künftiger OSEK-Echtzeit-Systeme an die Hardware nur durch ein komplettes Redesign der ak-

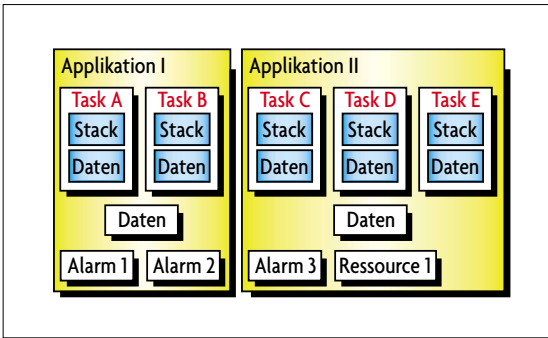


■ Bild 4. Der oSCAN TimingAnalyzer erlaubt die Simulation von Scheduling-Tabellen (Fahrplänen) und die Berechnung der „Schedulability“. Die Analyse wird mit den Informationen Priorität, Periode, Ausführungszeit und Deadline für jede Task und jede ISR durchgeführt.

(Quelle: Vector Embedded Symposium)

tuellen Prozessorkerne zu erfüllen. Über die Wünsche wird mit den Halbleiterherstellern aktuell diskutiert. Zu den wichtigsten Forderungen

gehören außer den bereits genannten Überwachungsfunktionen ein Interrupt-Controller für verschiedene Interrupt-Ebenen mit niedrigen Latenzzei-



Von der systematischen Analyse zum fehlerfreien System

Interessante Möglichkeiten zur Beherrschung der zunehmenden Komplexität der Elektronikentwicklung rund um das Automobil liefern aktuelle Projekte durch den intensiven Einsatz der Mathematik in den Ingenieurwissenschaften. Mit systematischen Analyseverfahren lassen sich im Zeitverhalten eines OSEK-Systems kritische Situationen aufdecken, die man selbst mit

ist, absolut fehlerfreie Embedded-Systeme und Elektronikkomponenten zu entwickeln. Sie wenden die Methoden der formalen Verifikation an, um Gesamtsysteme, bestehend aus Hardware, Software, Betriebssystem, Kommunikation, Anwendung usw., durchgängig zu verifizieren. In Zusammenarbeit mit dem Projektpartner Infineon wurde anhand des Prozessors TriCore 2, dem zukünftigen Flaggschiff der 32-bit-Mikrocontrollerfamilie, erstmals der Nachweis erbracht, dass diese in-

Bild 5. Für Tasks, die zur selben Applikation gehören, werden Zugriffsmöglichkeiten auf dieselben Speicherbereiche benötigt.

(Quelle: Vector Embedded Symposium)

ten, Hardware-Unterstützung beim Task-Wechsel und Prozessorkerne mit möglichst kleiner Registerzahl.

Eigenschaften künftiger Prozessorkerne

Bei den hardwarenahen und zeitkritischen Automotive-Anwendungen kommt es auf schnellstmögliche Reaktionsfähigkeit an. Ein Großteil der Anwendungen besteht aus Treibern und Interrupt-Service-Routinen (ISRs), die im Gegensatz zum Workstation-Bereich hier zur Anwendung gehören. Problematisch ist, dass sich die ISRs auf den aktuellen Controllern häufig nur komplett sperren lassen. Generell müssen Sperrmechanismen effizient realisierbar sein, da diese Grundfunktion sehr häufig durchlaufen wird.

Ein schneller Taskwechsel (**Bild 6**), von dem die Embedded-Echtzeit-Systeme „leben“, findet hardwareseitig derzeit nur rudimentär Unterstützung. Das Sichern und Rücksichern des Kontexts verbraucht einen Großteil der Ressourcen. Zum Kontext gehören neben Kernregistern, Registerbänken, Speicherzugriffsregistern, Gleitkomma- und Arithmetikregistern der Stackpointer, spezielle Peripherie-Einheiten und verschiedene Betriebssystemvariablen. Ideal wäre hier ein vollständig hardwareunterstützter Kontext-Switch.

Weiterhin hat sich gezeigt, dass Prozessoren mit einer geringen Registerzahl eine höhere Rechenleistung ermöglichen. Zahlreiche Register sind nur in typischen Workstation-Umgebungen sinnvoll nutzbar, weil dort die einzelnen Programmsequenzen relativ lange ohne Unterbrechung laufen. Ein möglicher Trend führt hier in Richtung so genannter Softcore-Prozessoren und zu Compilern, die das Konfigurieren der verwendeten Register erlauben.

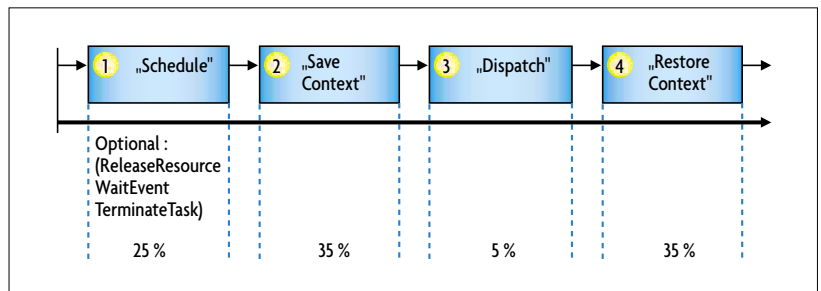


Bild 6. Die Phasen eines Task-Wechsels.

(Quelle: Vector Embedded Symposium)

umfangreichen Tests kaum gefunden hätte. Die Tools der Firma SympTA-Vision erlauben hierbei eine gezielte Suche nach „Flaschenhälsen“ und „Hot Spots“. So können die Anwender in Erfahrung bringen, was im Worst Case passieren wird. Die Vorteile der systematischen Analyse liegen in verringerten Testanstrengungen, in gesteigerter Produktivität und Qualität oder einer umfassenden Systemoptimierung.

Die Wissenschaftler des Verisoft-Projekts gehen noch einen Schritt weiter mit der Aussage, dass es möglich

novative Technik auf hochkomplexe Designs anwendbar ist. Das langfristige angelegte Verisoft-Projekt unter der Projektleitung von Prof. Dr. Wolfgang J. Paul von der Universität Saarbrücken wird vom Bundesministerium für Bildung und Forschung gefördert.

Hand in Hand: Software, Hardware und Tooling

Die Grundsteine und Basistechnologien für zuverlässige Elektroniksysteme im Automobil sind geschaffen. Auf Seiten der Controller-Hersteller müssen noch einige Herausforderungen bewältigt werden. Ansonsten liegt es in der Verantwortung von Fahrzeugherstellern und Zulieferern, die zur Verfügung stehenden Ressourcen optimal einzusetzen. Auf Basis einer leistungsfähigen zertifizierten OSEK-Implementierung bzw. eines AUTOSAR-konformen Embedded-Betriebssystems, zusammen mit einer durchgängigen Toolkette, lässt sich die Komplexität der Elektronikentwicklung in Zukunft rationell beherrschen. gs

Literatur

[1] Vortragsfolien des „Vector Embedded Symposium“:
www.vector-informatik.de/embedded

Dipl.-Ing. (FH) Peter Liebscher
hat an der Fachhochschule Esslingen Nachrichtentechnik studiert. Seit 2002 betreut er als Business Development Manager bei der Vector Informatik GmbH die Produktlinie Embedded Software Components.
peter.liebscher@vector-informatik.de