

© dSpace

ENTWICKLUNGSSYSTEME FÜR FAHRASSISTENZ

Effizientes Data Logging, Rapid Prototyping und Data Replay

Die Entwicklung von Fahrassistenz- und insbesondere hochautomatisierten Fahrsystemen stellt Automobilhersteller und Zulieferer vor große Herausforderungen. Daher ist es wichtig, dass leistungsfähige, flexibel konfigurierbare Entwicklungssysteme die vielfältigen Anwendungsfälle im Bereich der datengetriebenen Entwicklung optimal unterstützen, um diese so effizient wie möglich zu gestalten.

Bei Fahrassistenz- und insbesondere autonomen Fahrfunktionen kann die Wirkkette beginnend bei der Umfelderkennung mit Hilfe umfangreicher Sensorik bis hin zur finalen Umsetzung der Fahrstrategie und der Ansteuerung der Aktorik in mehrere Schritte unterteilt werden (Bild 1).

Die Sensorik ist der Startpunkt für alle weiteren Verarbeitungsschritte. Ihre Aufgabe ist es, ein möglichst detailliertes und komplettes Bild der Fahrzeugumgebung zu erfassen. Um dies bei allen Umgebungsbedingungen (Witterung, Lichtverhältnisse etc.) sicherzustellen, werden oft hochauflösende Sensortypen wie Kamera-, Radar-, Lidar und Ultraschallsensoren kombiniert.

In einem nächsten Schritt, der Perception, gilt es die erfassten Sensordaten zu interpretieren, um Objekte und deren relative Bewegung zu erkennen.

Da die verschiedenen Sensortypen unterschiedliche Stärken und Schwächen beim Erkennen dieser Informationen besitzen, werden die Daten im Rahmen der Sensor-Fusion überlagert, um ein möglichst realistisches Umgebungsabbild zu erzeugen. Zusammen mit weiteren Umgebungsinformationen basierend auf z. B. Kartendaten, Fahrzeugpositionen und Vehicle-to-X (V2X) Informationen bildet dieses Umgebungsabbild die Grundlage zur Bewegungsvorhersage des Fahrzeugs. Damit lassen sich die optimalen Trajektorien zur Fahrtplanung berechnen und die Sollgrößen für die Regelung des Fahrzeugs vorgeben.

Entwicklungsphasen und Systemansätze

Um neue Funktionen über alle Schritte der Wirkkette hinweg möglichst effizient

zu entwickeln, haben sich verschiedene Entwicklungsansätze etabliert. Zuerst werden neue Algorithmen oft im Labor prototypisch entwickelt und getestet. Neben klassischen Regelalgorithmen für z. B. die Aktoransteuerung kommen bei Perception und Sensorfusion heute zunehmend auch Methoden der Künstlichen Intelligenz (KI) wie etwa neuronale Netze zum Einsatz. Für ihr Training sind enorme Mengen realer Umgebungsdaten nötig, die u. a. durch zahlreiche Versuchsfahrten im Feld per Data Logging aufgezeichnet werden. Nach der Absicherung im Labor muss die Gesamtfunktionalität unter realen Bedingungen und im Kontext des Gesamtfahrzeugs mit Hilfe von Rapid Prototyping Methoden validiert und optimiert werden. Anschließend wird aus der optimierten Gesamtfunktion eine Serienimplementierung abgeleitet, welche im Labor durch

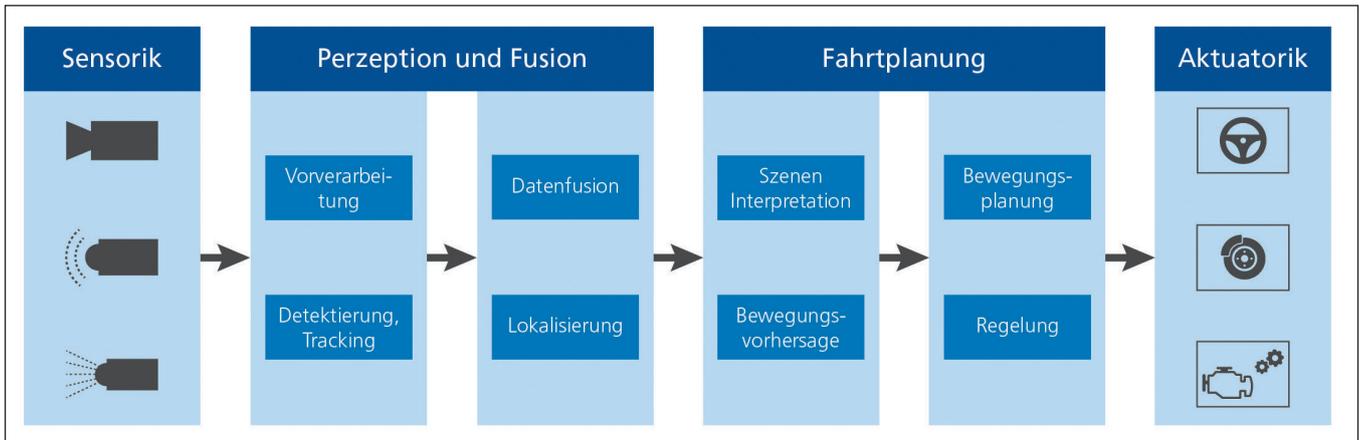


Bild 1: Wirkkette Fahrassistenz & autonomes Fahren. © dSpace

Software-in-the-Loop (SIL), Hardware-in-the-Loop (HIL) oder vermehrt auch Data-Replay/-Reprozessierung getestet werden muss.

Um Zeit und Kosten zu sparen, sollten die Entwicklungsphasen – vom Data Logging, über das Prototyping bis zum Testing – möglichst effizient gestaltet werden. Dabei ist darauf zu achten, dass das eingesetzte Entwicklungssystem für die jeweilige Phase optimal auslegbar ist, eine möglichst hohe Durchgängigkeit zwischen den Phasen existiert und eine Systemwiederverwendbarkeit gewährleistet ist. Im Folgenden werden die genannten Eigenschaften und spezifische Ausprägungen solcher Systeme im Zusammenhang mit den einzelnen Entwicklungsphasen am Beispiel der AUTERA Produktfamilie von dSpace näher erläutert.

High-End Data Logging

Zur Erfassung der Umfeld-Daten werden Versuchsträger mit einer Vielzahl hochauflösender Sensoren ausgestattet. Diese stellen die zu erfassenden Daten oft über verschiedenste Rohda-

tenschnittstellen wie z. B. GMSL, FPD-Link oder CSI aber auch über Ethernet und CAN FD zur Verfügung. Die fahrzeugtaugliche AUTERA AutoBox von dSpace bietet daher für das Data Logging einen modularen und konfigurierbaren Schnittstellen-Ansatz, der eine optimale Anpassung an die Sensoren erlaubt. Wichtig ist dabei, dass trotz der Modularität alle Daten zeitsynchron mit präzisiertem Zeitstempel, ggf. sogar über Systemgrenzen hinweg, versehen werden und in hoher Bandbreite zuverlässig auf einen Datenspeicher mit einer hohen Kapazität von mehreren Terrabyte (TB) gesichert werden können. So lassen sich pro AUTERA AutoBox Daten mit einer Bandbreite von bis zu 50 Gbit/s aufzeichnen. Die Speichergröße von 32 TB (zukünftig 64 TB) erlaubt unterbrechungsfreie Messfahrten auch bei maximaler Bandbreite über mehrere Stunden oder, bei geringerer Bandbreite, über Tage hinweg. Zudem sind die Speichereinheiten komfortabel während des laufenden Betriebs wechselbar, was die Aufzeichnungsdauer bei Testfahrten beliebig erweitert (Bild 2). Hervorzuheben ist in diesem Zusammenhang auch die

Möglichkeit der Integration von GPUs oder FPGAs zur intelligenten Datenanalyse, um schon bei der Erfassung Datenmengen auf relevante Daten gezielt und automatisiert zu reduzieren.

Um die aufgezeichneten Daten im Labor weiterzuverarbeiten, müssen sie üblicherweise in Datenzentren hochgeladen werden. Die AUTERA Upload Station ermöglicht dies z. B. über eine 100-Gbit-Ethernet-Schnittstelle.

Umfassendes Prototyping

Oben genannte Anforderungen an Data Logging-Systeme gelten im Wesentlichen auch für Prototyping-Systeme im realen Fahrversuch. Zudem ist es sinnvoll, während des Prototypings gleichzeitig Sensordaten oder interne Berechnungsgrößen aufzuzeichnen. Jedoch steht beim Prototyping die direkte Datenweiterverarbeitung zur Validierung und Optimierung der Algorithmen im Vordergrund. Dabei ist es wichtig, dass Prototyping-Systeme im Gegensatz zu Serienhardware ausreichende Leistungsreserven besitzen, damit nicht schon in frühen Phasen der Algorithmen-Entwicklung Einschränkungen nötig sind. In Fällen, in denen KI-Technologien genutzt werden, ist zusätzlich die Erweiterbarkeit des Systems um Hardware-Beschleuniger wie GPUs oder FPGAs sinnvoll. So basiert die AUTERA AutoBox auf Server-Technologie und erlaubt optional die Integration von leistungsstarken Hardware-Beschleunigern verschiedener Hersteller. In Fällen, in denen teilweise noch kein finales Aktuator-Steuergerät vorliegt (bzw. ein existierendes weiter zu entwickeln ist) hat sich ergänzend die Nutzung von separat und »



Bild 2: Die Speichereinheiten des AUTERA-Systems können während des Betriebs komfortabel gewechselt werden ohne das System neu starten zu müssen.

© dSpace

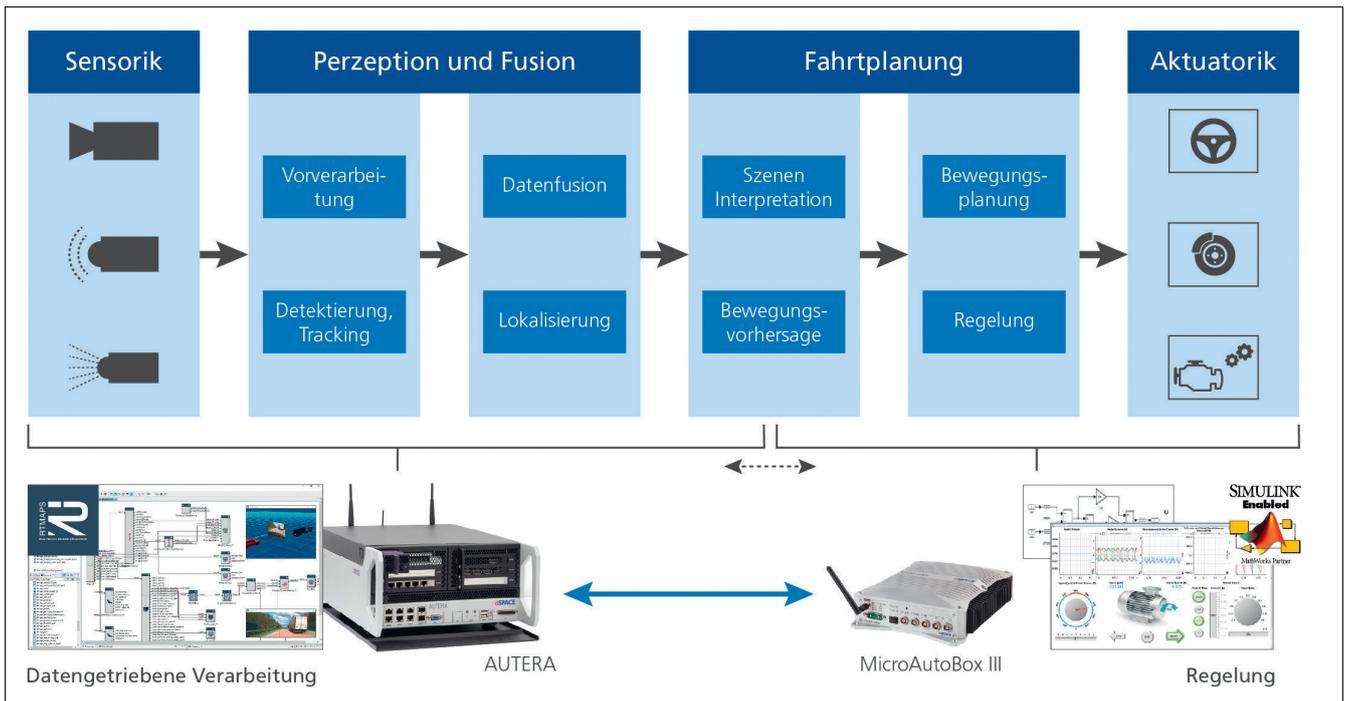


Bild 3: Umfassende Prototypinglösung für die gesamte Wirkkette. © dSpace

damit skalierbar koppelbaren Rapid Control Prototyping (RCP) Systemen wie der MicroAutoBox von dSpace bewährt. Auf ihr lässt sich komfortabel regelungstechnische Algorithmik mit hohen Echtzeitanforderungen implementieren und ausführen sowie die Aktuatorik über diskrete I/O Schnittstellen mit Hilfe von Leistungsendstufen direkt ansteuern (Bild 3).

Neben der Hardware spielt die Systemsoftware eine nennenswerte Rolle für effizientes Prototyping. Meist werden bei derartigen Prototyping-Systemen Linux-basierte Betriebssysteme eingesetzt, die eine hohe Flexibilität und Offenheit bieten. Zur konkreten Implementierung und Ausführung der Algorithmen werden diese häufig um spezialisierte Software, z. B. ROS (Robot Operating System) oder RTMaps von Intempora erweitert. Um den Komfort und die Produktivität zu steigern, besitzen Werkzeuge wie RTMaps ein intuitive grafische Bedienoberfläche zur Konfiguration der Sensorik sowie der Algorithmen-Implementierung auf Basis von C++, Python oder CUDA. Die so entwickelten Algorithmen berechnen nun die Stellgrößen zur Regelung des Fahrzeugs (Bremsen, Lenkung, Fahrwerk, Antrieb etc.) und müssen mit den Aktuator-Steuengeräten im Fahrzeugnetzwerk austauschen können. Dafür ist die Unterstützung der verschiedenen Busse und Netzwerke inkl. häufig genutzter Beschreibungsformate wie z. B. AUTOSAR XML oder Fibex nötig. Die AUTERA

AutoBox lässt sich daher toolunterstützt u. a. auf Basis dieser Beschreibungsformate für die Kommunikation über CAN FD, Ethernet (1000BASE-T, 1000BASE-T1, 10GBASE-T) und zukünftig FlexRay konfigurieren.

Durchgängiges Data Replay

Nach ihrer Validierung und Optimierung im realen Fahrversuch muss die Algorithmik im Seriensteuergerät implementiert und danach im Labor u.a. mittels Data-Replay/Reprozessierung getestet werden. Insbesondere zur Absicherung von Perzeptions- und Fusionsalgorithmen sind die beim Data Logging aufgezeichneten und annotierten Daten (Ground Truth) zeitlich identisch und hochsynchron genauso abzuspielen, wie sie ursprünglich aufgenommen wurden. Um auch hier den Aufwand im Entwicklungsprozess klein zu halten, ist eine hohe Durchgängigkeit und Kompatibilität vom Data Logging zum Data Replay System ratsam. Im Idealfall kommt sogar ein ähnlicher Systemaufbau zum Einsatz, beispielsweise um die Speichereinheiten vom Data Logging wiederzuverwenden und von diesen die Daten beim Replay abzuspielen. Sind die Daten in Datenzentren abgelegt und ist eine direkte Streaming-Verbindung gewünscht, kann das Data Replay-System auch direkt mit hoher Bandbreite angekoppelt werden. Falls beim Datenstreaming von den Datenzentren Übertragungsschwankungen auftreten, können

ausreichend große Zwischenspeicher im Data Replay-System diese durch Daten-Pufferung kompensieren. Darüber hinaus lassen sich Data Replay Systeme durch die Ausführung synthetischer Sensormodelle oder die Kopplung mit HiL-Systemen auch für den Closed-loop-Anwendungsfall nutzen.

Fazit

Am Beispiel der AUTERA Produktfamilie wurde gezeigt, dass eine skalierbare Systemarchitektur mit hoher Rechenleistung, hohen Bandbreiten und einer Vielzahl von Erweiterungsoptionen wie leistungsfähige Datenspeicher, Hardware-Beschleuniger sowie Rohdaten-, Bus- und Netzwerkschnittstellen zusammen mit umfangreicher Software die Basis für ein effizientes Entwicklungssystem bilden. Eine hohe Durchgängigkeit und Kompatibilität vom Data Logging über das Prototyping bis hin zum Data Replay ist darüber hinaus wichtig, um Brüche im Entwicklungsprozess zu vermeiden. ■ (oe)

www.dspace.com



Marius Müller Ist strategischer Produkt Manager Prototyping Systems bei der dSPACE GmbH.



Frank Mertens Ist Direktor strategisches Produktmanagement Real-Time Test & Development Solutions bei der dSPACE GmbH.