



Entwicklung eines kooperativen Traktorgespanns

Während Fahrerassistenzsysteme zunehmend die Kontrolle über Längs- und Querregelung von Fahrzeugen im Straßenverkehr übernehmen [1], schreitet in der Landtechnik die Entwicklung fahrerloser Maschinen voran. Diesen Trend unterstützt auch die am Lehrstuhl für mobile Arbeitsmaschinen des Karlsruher Instituts für Technologie, KIT, in Kooperation mit den Firmen AGCO und geo-konzept entwickelte elektronische Deichsel für landwirtschaftliche Arbeitsmaschinen [2].

Automatisiertes Fahren steht in der Automobilbranche hoch im Kurs. Auch bei der Entwicklung landwirtschaftlicher Arbeitsmaschinen gewinnt dieses Thema zunehmend an Bedeutung. In einem Forschungsprojekt beim Karlsruher Institut für Technologie wurde nun ein System entwickelt, bei dem eine bemannte Landmaschine als Leitfahrzeug voraus fährt und sich eine weitere autonom und fahrerlos daran orientiert. Ziel des Projektes: Steigerung der Flächenleistung und Optimierung der Maschinenauslastung in der Agrarwirtschaft ohne zusätzlichen Personalaufwand.

Elektronische Deichsel für Traktoren

Die elektronische Deichsel ermöglicht das zeitgleiche Führen von zwei Traktoren mit nur einem Fahrer. Hierzu werden zwei baugleiche Traktoren, ausgestattet mit identischem Anbaugerät, mit hochgenauen RTK-GNSS-Empfängern (Real-Time-Kinematics) versehen und per Datenfunk miteinander verbunden. Während ein Traktor, als Masterfahrzeug konfiguriert, bemannt vorweg fährt, folgt ein unbemanntes, als Slave konfiguriertes Fahrzeug mit vorgegebenem longitudinalen und lateralen Ver-

satz. Die Arbeitseinstellungen des Masterfahrzeugs werden zur Umsetzung des Arbeitsprozesses ortsgebunden vom Slave kopiert. Eine gute Bedienbarkeit und hohe Sicherheit wird durch die Nutzung von Umfellsensorik zur Hinderniserkennung (Bild 1) und webbasierten Geo-Informationen erzielt.

Systemfunktion

Zum Betrieb der elektronischen Deichsel stehen dem Fahrer fünf Fahrmodi zur Verfügung. Als Start- und Arbeitsmodus dient der Modus Parallel Driving. Hier folgt der Slave parallel der



Fahrspur des Masters und erfüllt dieselbe Arbeitsaufgabe. Soll der Slave exakt in der Fahrspur des Masters folgen, hilft der Modus Tracking. Der Modus Ignore erlaubt es den Slave kurzfristig auszukoppeln, um mit dem Master unabhängige Fahrmanöver durchzuführen. Der Slave schließt bei Verlassen des Modus wieder auf direktem Weg auf. Für das Wendemanöver am Vorgewende steht der Fahrmodus Turn-Over zur Verfügung. Wird ein Hindernis im geplanten Pfad gemeldet, sucht sich der Slave im Modus Evasion einen kollisionsfreien Ausweichpfad.

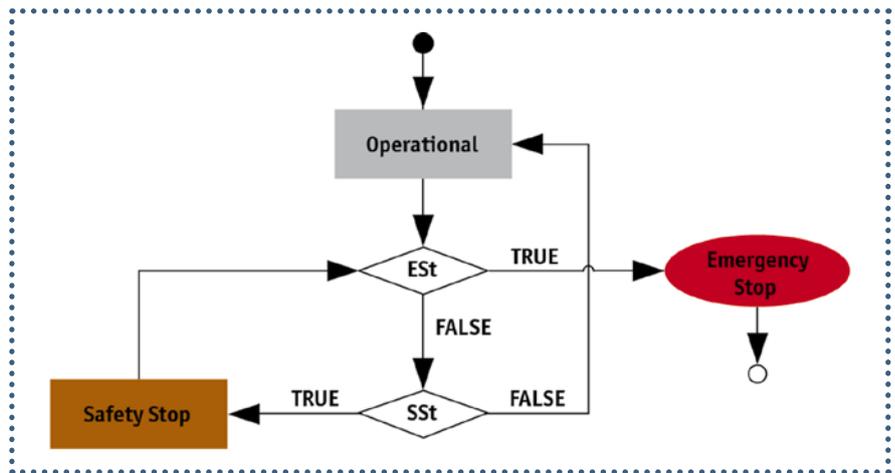


Bild 2: Sicherheitskonzept der elektronischen Deichsel.

Sicherheitskonzept und Zustandsautomat

Sicherheit ist in der Entwicklungsphase eines unbemannten Fahrzeugs ein wesentlicher Aspekt. Nach ISO 25119 sind mithilfe typischer Werkzeuge Risikoquellen und Schwachstellen der funktionalen Sicherheit einer Maschine zu identifizieren und mit geeigneten Maßnahmen zu entschärfen. Eine Risikoanalyse liefert für jedes betrachtete Szenario im Produktlebenszyklus einen sogenannten Agricultural Performance Level (AgPL), aus welchem Anforderungen an Hard- und Software abgeleitet werden können. Der elektronischen Deichsel liegt ein dreistufiges Sicherheitskonzept zugrunde (Bild 2).

Im Zustand Operational liegen keine kritischen Fehler- und Warnmeldungen vor; der geplante Pfad ist frei von Hin-

dernissen. Der Zustand Safety Stop (SSst) ist als aktiver Halt in Bezug auf Fahr- und Arbeitsantriebe zu verstehen. Als höchster Level des Sicherheitskonzepts dient der Zustand Emergency Stop (EST). Hier wird mit sofortiger Wirkung der Motor abgeschaltet und die elektronische Feststellbremse betätigt.

Die Umsetzung des Sicherheitskonzepts erfolgt in Form eines Zustandsautomaten, der als zentrales Softwaremodul auf beiden Fahrzeugen aktiv ist und synchronisiert den Gesamtsystemzustand festlegt. Bild 3 zeigt den Zustandsautomaten der elektronischen Deichsel. Bei Systemstart ist zunächst der Zustand Default aktiv.

Die Rolle des Traktors kann als Master oder Slave festgelegt werden. Nach Festlegung der Rollen muss im

Zustand Assignment mit beiden Fahrzeugen eine Zuweisung durchgeführt werden, in der die Funkverbindung zwischen beiden Fahrzeugen hergestellt wird. Wurden zwei Fahrzeuge zugewiesen, geht das System in den Safety Stop über. Der Slave befindet sich nun im aktiven Halt. Nun kann zu jedem Zeitpunkt der Emergency Stop ausgelöst werden. Um den Fahrbetrieb aufzunehmen, werden die Fahrzeuge nach Freigabe durch den Fahrer im Zustand Docking mit einander gekoppelt, sofern Relativpositionen und Ausrichtung dies zulassen. Das System geht in den Zustand Operation über und der Slave ist eigenständig fahrfähig. Vom Standardfahrmodus Parallel Driving aus kann jeder andere Fahrmodus aktiviert werden. Kollisionsgefahr wird von der Umfeldsensorik bestehend aus 2D-Laserscannern und 3D-Kameras erkannt und führt unmittelbar zum Safety Stop.

Prototypenentwicklung an verteilten Standorten

Die Umsetzung eines komplexen modularen Assistenzsystems mit externen Projektpartnern erfordert eine detaillierte Spezifikation der Schnittstellen. Aufgrund der nicht harmonisierten Entwicklungsplattformen bei den einzelnen Partnern wurde eine verteilte Architektur gewählt, in der verschiedene Soft- und Hardwaremodule über einen proprietären edaugCAN-Bus und über TCP/IP Ethernet miteinander kommunizieren.



Bild 1: Prototypischer Aufbau zum Test der Umfeldsensorik des Slave-Traktors.



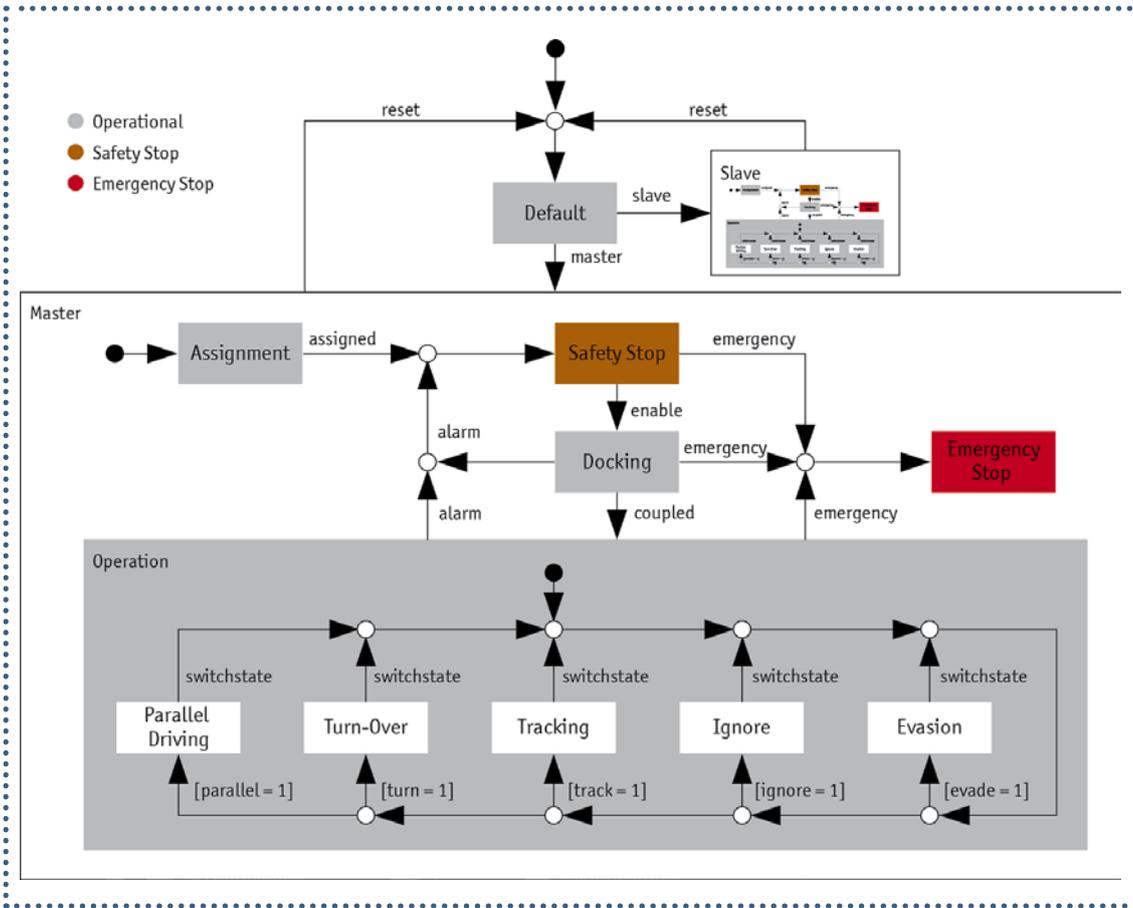


Bild 3: Modell zur Festlegung des Betriebszustands.

Hardwarearchitektur

Während die Umsetzung der Navigationsfunktionen (NAV) und der Zugang zum proprietären, maßgeschneiderten Geo-Datenserver (GIS), sowie der Zugriff auf die Fahrzeugsteuerung (EXT) über das Hauptsteuergerät und die Bedienoberfläche für das serienmäßige Terminal (HMI) in der Verantwortung der Projektpartner lag, wurden am KIT der Zustandsautomat (ZST) und die Umfelderkennung (UFS) auf einer Rapid-Prototyping-Plattform implementiert.

Datenkommunikation

Für den Austausch von System- und Betriebsdaten innerhalb der Fahrzeuge über das proprietäre CAN-Protokoll wurde physikalisch der ISOBUS des Traktors genutzt. Durch die Wahl von 11-bit-Identifiern können beide Busse störungsfrei parallel betrieben werden. Jedes Hardware-Modul verfügt über eine eigene CAN-ID, unter der mittels Multiplexing alle relevanten Signale

übertragen werden. Die bandbreitenintensiven Umfelddaten werden über TCP/IP-Ethernet übertragen. Geringe Teilnehmerzahl und Netzwerklast erlauben einen echtzeitfähigen Versand und Empfang der Daten. Funkkommunikation zwischen den Fahrzeugen erfolgt im Vollduplex-Verfahren. Die verwendeten Modems werden über RS232-Schnittstellen eingebunden.

Master und Slave tauschen laufend Daten über Position und Bewegungszustand, Systemzustand, Einstellungen

der Arbeitsantriebe, den Fahrzeugzustand und Umfeldinformationen aus (Bild 4). Die pragmatische Gestaltung der Schnittstellen ermöglicht eine unkomplizierte Umsetzung und Validierung der Teilfunktionen. Als Sicherheitsmechanismen zur Diagnose typischer Datenübertragungsfehler wurden Timeouts und Prüfsummen verwendet, um Datenintegrität und die Aktivität der Teilnehmer zu gewährleisten. Ein Überschreiten eines Timeouts auf dem CAN-Bus oder in der Funkstrecke löst

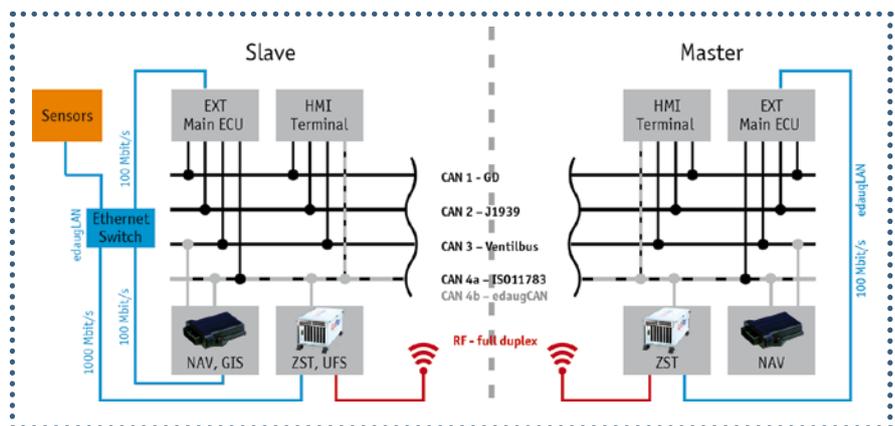


Bild 4: Kommunikationsarchitektur im Prototyp.



unmittelbar den Emergency-Stop aus, da die Kontrolle über das Fahrzeug nicht mehr gewährleistet werden kann.

Restbussimulation zur Unterstützung der Softwareentwicklung

In der Entwicklung von Steuergeräten stellt die Restbussimulation (RBS) ein mächtiges Werkzeug zur Unterstützung des Debuggings und zur Validierung der Software unter möglichst realitätsnahen Randbedingungen dar. Nicht nur in der Automobilbranche, sondern auch bei mobilen Arbeitsmaschinen kommunizieren Steuergeräte, Sensoren und Aktuatoren gleichzeitig in verschiedenen Netzwerken. Umfassende Softwaretests sind meist erst möglich, wenn die Schnittstellen des Steuergeräts mit realistischen Werten versorgt werden können. Um Softwaretests in der Entwicklungsphase nicht aufwendig im realen System durchführen zu müssen, kann mithilfe einer RBS die übrige Netzwerkumgebung eines Steuergerätes virtuell nachgebildet werden. Softwaretests erfolgen auf diese Weise kostengünstig, schnell und sicherheitsunkritisch.

CANoe

Die Software CANoe der Firma Vector Informatik unterstützt flexibel die Entwicklung der Netzwerkkommunikation. Die Struktur eines CAN-Netzwerkes wird in CANoe vollständig mithilfe sogenannter Netzknoten abgebildet, in denen das Kommunikationsverhalten der Teilnehmer hinterlegt wird. Darüber hinaus können Bedien-Interfaces, Signalgeneratoren und weitere Kommunikationsschnittstellen als Netzknoten angelegt werden. Das eventgesteuerte oder interaktive Verhalten kann in CAPL, einer C-ähnlichen Programmiersprache [3], oder bspw. mithilfe kompilierter Simulink-Modelle erfolgen. In einer Datenbasis werden alle vorkommenden Nachrichten inklusive Signalaufbau hinterlegt und einzelnen Netzknoten zugewiesen. CANoe ist multibusfähig, d. h. sowohl proprietäre als auch standardisierte Protokolle wie CAN, J1939 oder ISO 11783 und auch FlexRay und Ethernet werden unter-

stützt. Zur Modellierung und zeitsynchronen Auswertung steht ein Set von Stimulations- und Tracing-Funktionen mit vielfältigen Filtermöglichkeiten, grafischer Signaldarstellung sowie eine Übersicht über wichtige Kommunikationsparameter wie Buslast oder Fehler Nachrichten zur Verfügung. Beliebige Steuergeräte können nun wahlweise virtuell oder real betrieben werden, indem die Netzknoten in CANoe aktiviert oder deaktiviert werden. Die Busan Kopplung erfolgt über ein Interface der Firma Vector. CANoe kann darüber hinaus Testscenarien vollautomatisch ablaufen lassen und protokollieren. Diese Testscenarien lassen sich wahlweise tabellarisch sowie in XML, CAPL [3], C# oder grafisch definieren.

Modellierung der elektronischen Deichsel in CANoe

CANoe eignet sich in dieser Projektkonstellation insbesondere zur frühzeitigen Validierung der Steuergeräte, da die Abstimmung von Terminen und Testmöglichkeiten mit den Versuchsfahrzeugen oft kritisch ist. Der Ansatz der RBS ermöglicht parallele Entwicklungspfade, die anschließend leicht zusammengeführt werden können.

Am KIT wurde ein umfassendes Modell der elektronischen Deichsel entwickelt, mit dem einzelne Steuergeräte, sowie die vollständige Kommunikation mit dem Partnerfahrzeug simuliert werden können. Die Software-Module ZST, NAV und HMI wurden dazu als Netzknoten modelliert. Die Detailtiefe der einzelnen Module variiert innerhalb des Modells. Während alle den Systemzustand steuernden Nachrichten in vollem Funktionsumfang abgebildet wurden, genügt zur Validierung der Datenintegrität in den übrigen Nachrichten die Zuweisung konstanter Testparameter. Die Nutzung von CANoe in Verbindung mit der modellbasiert programmierbaren dSpace-Plattform hat sich als besonders hilfreich erwiesen. Die entwickelten Simulink-Modelle können nach Austausch der plattform-spezifischen I/O-Blocks unmittelbar für CANoe kompiliert und in das Restbusmodell integriert werden. Durch die Zugriffsmöglichkeit auf die seriellen

Schnittstellen des PCs in CANoe, konnte auch die Funkstrecke unverändert integriert und in Hinblick auf Datenintegrität getestet werden.

Zusammenfassung

Die elektronische Deichsel hat sich nicht nur in Hinblick auf den innovativen Schwerpunkt Hinderniserkennung und Integration von Geo-Informationen in einem stimmigen Sicherheits- und Bedienkonzept als komplex erwiesen. Die Spezifikation und Pflege von Datenschnittstellen ist in einem sich kontinuierlich ändernden prototypischen System eine begleitende Aufgabe, die von allen Seiten ernst genommen werden muss. Eine sauber spezifizierte modulare System- und Softwarearchitektur erweist sich bei der Kooperation mehrerer Firmen schnell als wesentliche Grundlage für einen schnellen Projektfortschritt. Rapid-Prototyping-Systeme in Verbindung mit einer RBS-Software stellen jedoch einen projektbegleitenden Werkzeugkasten zur effizienten Validierung von Teilfunktionen und ganzen Steuergeräten dar, um im abschließenden Feldtest erfolgreich zu sein. ■ (oe)

» www.vector.com

Literatur

- [1] Winner, H.; Hakuli, St.; Wolf, G. (2012). Handbuch Fahrerassistenzsysteme. 2. Auflage. ISBN 978-3-8348-1457-9. Vieweg + Teubner Verlag, Springer Fachmedien Wiesbaden GmbH.
- [2] Zhang, X.; Geimer, M.; Noack, P.O.; Grandl, L. (2010). Development of an intelligent master-slave system between agricultural vehicles. pp. 250-255. IEEE. Intelligent Vehicles Symposium. 6/21-6/24/2010. San Diego, CA, USA
- [3] Marc Lohmeyer/Roman Markt: Programming ECU Tests More Efficiently – Tips and Tricks for the Use of CAPL. In: CAN Newsletter (2014), H. 2, S. 42-43



Dipl.-Ing. Bernhard Jahnke ist wissenschaftlicher Mitarbeiter am Lehrstuhl für Mobile Arbeitsmaschinen des Karlsruher Instituts für Technologie (KIT).



Dipl.-Ing. Hans-Werner Schaal ist Business Development Manager im Bereich Automotive Ethernet, Car2x und offener CAN-Protokolle wie J1939 und ISO 11783 bei der Vector GmbH.