

IM FOCUS

Diagnosekommunikation

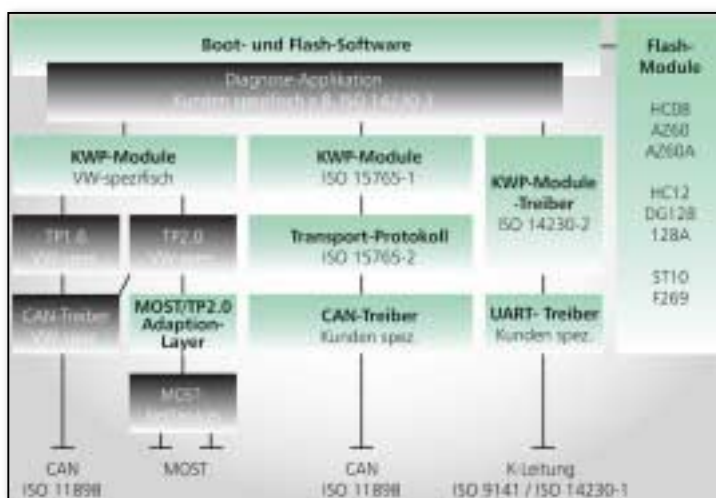
Elektronische Flachbaugruppen, Module und Hybride lassen sich bis heute und leider auch in Zukunft nicht fehlerfrei fertigen. Sie müssen also zu hundert Prozent getestet werden. Dabei unterscheiden wir grob zwei Testverfahren, nämlich den Incircuittest und den Funktionstest.

Steuergeräte verfügen über eine Diagnoseschnittstelle, die zum Beispiel erlaubt, die Identifikation, die Konfiguration und natürlich auch den Fehlerpeicher eines Steuergerätes auszulesen und auszuwerten. Mag dies auch nicht als große Neuigkeit erscheinen – die Erfahrung zeigt, daß bei der Entwicklung von Steuergeräten die Implementierung des Diagnosemoduls häufig hintangestellt oder gar „vergessen“ wird und dann in letzter Sekunde vor der Serienfreigabe eine Lösung „herbeigezaubert“ werden muss.

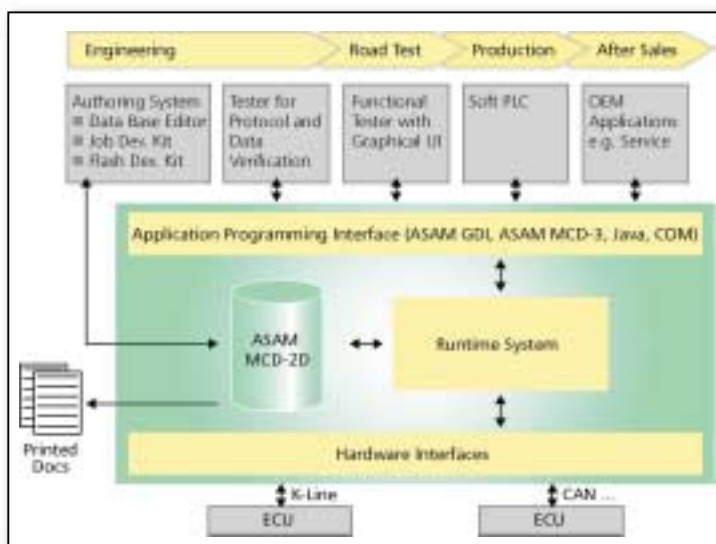
Dabei hat sich gerade in Bezug auf die Diagnose in den letzten Jahren einiges bewegt. Der doch recht frei interpretierbare Standard ISO 9141-2 wurde abgelöst durch das Keyword Protocol 2000 (KWP 2000), definiert in der ISO 14230/Road Vehicles – Diagnostic Systems. Für die herstellereigenspezifische Diagnose wird üblicherweise eine Teilmenge der in der ISO 14230 definierten Diagnostic Services implementiert. Mit ISO 14230 Keyword Protocol 2000 und ISO 15765 Diagnostics on CAN ist die Diagnoseschnittstelle der Steuergeräte nun weitgehend standardisiert.

Die Steuergeräte-Kommunikation und deren Standardisierung steht bei Softings Geschäftsbereich Automotive Electronics bereits seit mehr als 10 Jahren im Mittelpunkt des Interesses. Und das gilt insbesondere für das Thema Diagnose mit allen Facetten der Diagnosekommunikation – sei es im Tester- oder im Steuergerätebereich. Softing bietet Lösungen angefangen bei den Implementierungen von Benutzeroberflächen für Prüfabläufe bis hin zu den Diagnoseprotokollen für die tatsächliche Schnittstelle zum Steuergerät. Standardisierung und Optimierung der sogenannten Prozesskette sind seit langem wichtiger Aspekt dieser Lösungen. Aufseiten der Steuergeräte wurden in den letzten Jahren zahlreiche Funktionsmodule – vor

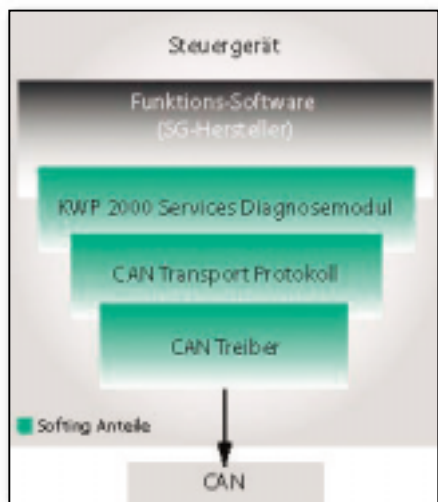
allem Diagnose-Kommunikationsmodule – implementiert. Es wird jeweils das gleiche Diagnosemodul KWP 2000 mit den KWP-2000-Services als Funktionsprototyp eingesetzt. Für die K-Leitung wurde ein K-Leitungstreiber entwickelt, für CAN wird sowohl ein CAN-Treiber als auch ein CAN-Transportprotokoll (Diagnostics on



Schemadarstellung der Diagnose-Schnittstellen



Schemadarstellung des gesamten Prüfsystems



Die Diagnosekommunikation mit dem Steuergerät

CAN oder Transportprotokoll 1.6 oder 2.0) verwendet.

Boot- und Flash-Software für Steuergeräte

Zusätzlich zur Diagnose kann auch eine Boot- und Flash-Software realisiert werden. Dabei enthält die Boot-Software Funktionen zur Ermittlung und zum Test der Checksumme der Steuergeräte-Software und zum Kopieren der Flash-Software von ROM nach RAM. Die Flash-Software enthält alle Funktionen zum Löschen und Programmieren des Flash-Speichers des Steuergeräts. Die Flash-Programmierung eines Steuergeräts läuft dann folgendermaßen ab: Nach je-

dem Reset des Steuergeräts wird die Boot-Software gestartet. Die Boot-Software ermittelt die Checksumme der Steuergeräte-Software und vergleicht sie mit der auf einer festgelegten ROM-Adresse. Ist die ermittelte Checksumme mit der abgelegten identisch, so erfolgt der Sprung zum Reset-Vektor der Steuergeräte-Software. Stimmt die Checksumme nicht überein, so werden die Flash-Software, der KWP-2000-Treiber und das Transportprotokoll ins RAM kopiert und von dort aus gestartet. Wurde bei der Checksummen-Prüfung korrekte Steuergeräte-Software erkannt und aktiviert, kann die Flash-Software über die Diagnostic Services (z. B. „StartDiagnosticSession“, „DownloadRequest“, „TransferData“, ...) aktiviert werden. Mit dieser Aktivierung der Flash-Software werden alle erforderlichen Funktionen ins RAM kopiert und von dort aus gestartet.

Diagnosekommunikation auf dem Steuergerät

Die Bearbeitung von Diagnoseinformationen auf dem Steuergerät und der Austausch von Informationen mit einem Tester wird über folgende Schritte abgewickelt:

- ▷ Die Diagnoseinformation (z. B. der Inhalt des Fehlerspeichers oder die Identifikation) wird von der Steuergeräteapplikation für die Diagnoseapplikation bereitgestellt.

- ▷ Die Diagnoseinformation wird im entsprechenden Diagnostic Service (gemäß ISO 14230) verpackt.

- ▷ Die Diagnose-Nachricht wird segmentiert, d. h. in für die Übertragung geeignete Häppchen zerteilt. Eine Segmentierung ist bei serieller Übertragung (K-Leitung) nicht erforderlich.

- ▷ Die Diagnose-Nachricht wird übertragen.

Für vom Tester eingehende Diagnose-Nachrichten gestaltet sich der Ablauf analog in umgekehrter Richtung.

Das eigentliche Diagnosemodul, das die Diagnostic Services bedient, ist unabhängig vom verwendeten Controller und auch vom verwendeten Übertragungsmedium. Es verfügt über eine wohldefinierte Schnittstelle, sowohl zur Seite der Steuergeräteapplikation als auch zur Seite der Transportschicht bzw. des Treibers.

Wie bereits gesehen, erfolgt die Kommunikation zwischen einem Steuergerät und einem Tester entweder über eine serielle Leitung (K-Leitung) oder über CAN. Bei der Diagnosekommunikation über die serielle K-Leitung wird ein kunden- bzw. prozessor-spezifischer UART-Treiber (Universal Asynchronous Receiver/Transmitter) eingesetzt. In diesem Fall können die vom KWP-2000-Modul erzeugten Diagnose-Nachrichten unsegmentiert (also als vollständige Diagnosetelegramme) übertragen bzw. die Anforderungstelegramme des Testers unsegmentiert empfangen werden.

Falls der CAN-Bus (mit maximal 8 Byte Nutzdaten pro Nachricht) als Übertragungsmedium eingesetzt wird, ist unterhalb des KWP-2000-Moduls eine Transport-Schicht erforderlich. Sie setzt die vom Tester eingehenden, in CAN-Botschaften segmentierte Diagnose-Nachrichten zusammen bzw. segmentiert ausgehende Diagnose-Nachrichten entsprechend dem CAN-Botschaftsformat. Hier kann entweder das in der ISO 15765 spezifizierte Transportprotokoll oder ein kundenspezifisches Protokoll (z. B. Transport-

protokoll 1.6 oder Transportprotokoll 2.0) eingesetzt werden. Der zum Senden und Empfangen der CAN-Nachrichten verwendete CAN-Treiber muß abhängig vom verwendeten Controller implementiert werden.

Kosten-Nutzen-Rechnung

Die Erkenntnis, dass der Einsatz von standardisierten bzw. standard-basierten Lösungen auch der Fahrzeug-Diagnose wesentliche Vorteile bietet, hat sich inzwischen weitgehend durchgesetzt. Der Erfolg einer Steuergeräte-Entwicklung hängt aber auch nicht zuletzt davon ab, daß den Entwicklern ausreichend Zeit für ihre Kernaufgabe zur Verfügung steht. Wenn das Diagnosemodul von einem Spezialisten geliefert wird, können sich Fahrzeughersteller bzw. Zulieferer – ebenfalls als Spezialisten – mit der Funktion des Steuergeräts befassen. Softing hat Komponenten, die für die Diagnosekommunikation des Steuergeräts erforderlich sind, bereits vielfach implementiert und auf mehreren unterschiedlichen Prozessoren (z. B. Infineon C16X, Motorola 68HCXX, Mitsubishi M16C, NEC V850, NEC 780818 und ST10F269) in Betrieb genommen. Der vielfache Einsatz und die mehrfache Optimierung fertiger Module stehen für eine hohe Qualität der Software. Somit beschränkt sich der Testaufwand mehr oder minder auf die Verifikation der Schnittstellenfunktionen. Das Implementierungsrisiko ist auf ein Minimum reduziert. Die portablen und für die unterschiedlichsten Prozessor-Typen modifizierbaren Module sind intensiv ausgetestet und müssen also nur noch an die Applikations-Software des Kunden angepaßt werden. Auch die Implementierungszeit reduziert sich auf ein Minimum. Typischerweise wird eine Implementierung der Diagnostic Services erst sehr spät in Angriff genommen. Die hausinternen Ressourcen beim Fahrzeughersteller/-Zulieferer sind dann meist durch die Inbetriebnahme der Funktionssoftware gebunden. Eine sofort verfügbare, ausgereifte Fertiglösung kommt hier sozusagen wie gerufen. Die Kosten für eine solche Implementierung sind exakt vorhersagbar, Unwägbarkeiten sind so gut wie ausgeschlossen. Durch die Möglichkeit der Wiederverwendung vorhandener Komponenten kommt außerdem ein erheblicher Preisvorteil gegenüber Eigenentwicklungen des Steuergeräteherstellers zustande. Der Kunde kann die Quellcodes zudem in Folgeprojekten einsetzen. Die Entwickler beim Auftraggeber sind frei für ihre Kernaufgabe – die Entwicklung der Steuergerätefunktionen.