

Prototyping und Test von CANopen-Systemen

Mit minimalem Aufwand rasch zum Ziel

CANopen gilt in zahlreichen Anwendungsbereichen als Standard für die günstige und flexible Vernetzung von Komponenten. Ein spezialisiertes Prototyping- und Test-Tool leistet nicht nur bei der Entwicklung komplexer CANopen-Projekte wertvolle Dienste, sondern auch beim grundsätzlichen Einstieg in die Thematik.

» Mirko Tischer

Die Bandbreite der Aufgaben bei der Entwicklung von CANopen-Systemen reicht von der Entwicklung einzelner Steuergeräte bis hin zur Konfiguration und Inbetriebnahme der Gesamtsysteme. Für Systemhersteller ist der Ersteintritt eines Systems wie CANopen zunächst mit einem nicht kalkulierbaren Einstiegsaufwand verbunden. Dabei lässt sich ein grosser Teil der Entwicklungsaufgaben im Rahmen des V-Modells (Bild 1) dem Bereich Verifikation und Validierung zurechnen. Das Risiko zu spät entdeckter Fehler minimiert man hier sinnvollerweise durch umfassende Tests zu einem möglichst frühen Zeitpunkt.

Systemverifikation über Testaufbauten

Unverzichtbar in allen Entwicklungsschritten eines Gesamtsystems sind begleitende systematische Tests. Häufig besteht die Möglichkeit zum Testen jedoch erst zu einem sehr späten Zeitpunkt, wenn die Systemkomponenten tatsächlich verfügbar sind. Treten dann Probleme auf, droht Gefahr für den Fertigstellungstermin des Systems.

In der Praxis führt man die Tests nicht anhand des realen Systems im Haus des Kunden durch, sondern bedient sich Testaufbauten. Diese enthalten neben speziellen Mess- oder Diagnoseeinrichtungen für den Testbetrieb unter anderem Aktoren, um möglichst reale

Systemumgebungen zu simulieren. In Abhängigkeit von der Projektgrösse kann ein solcher Testaufbau aufwendig und kostenintensiv werden. Da in der Regel nur ein einziger Testaufbau vorhanden ist, sind Engpässe während der Testphase vorprogrammiert. Einen Ausweg aus diesem Dilemma bietet ein Softwarewerkzeug, mit dessen Hilfe man auf einfache Art und Weise einen Prototyp des zukünftigen Gesamtsystems erstellt. Optimalerweise bietet das Werkzeug auch noch umfangreiche Testfunktionen.

Erstellung einer Prototypenumgebung mit Tool-Unterstützung

Prototypen für über CAN vernetzte Gesamtsysteme sollen in erster Linie Test- und Validierungsaktivitäten unterstützen. Weiterhin ist wichtig, Einzelkomponenten über reale oder simulierte Steuergeräte abzubilden. Damit lässt sich die Funktionsfähigkeit realer Steuergeräte im Laufe der Systementwicklung relativ leicht prüfen.

Einen Prototyp für ein komplexes Gesamtsystem zu erstellen ist ein aufwendiges Unterfangen, sodass sich der Einsatz entsprechender Werkzeuge bezahlt macht. Wertvolle Hilfe beim Erstellen der Kommunikation des Prototypensystems leistet CANoe.CANopen von Vector Informatik. Damit erstellt der Anwender über wenige Konfigurationsschritte einen Prototyp, der in seinen Kommunikationseigenschaften dem späteren realen System entspricht.

Das Verhalten der CANopen-Steuergeräte ist in Beschreibungsdateien (EDS – Electronic Data Sheet) definiert, die man zunächst auswählt bzw. erstellt. Im nächsten Schritt werden die später über den Bus ausgetauschten Applikationsdaten miteinander verbunden, beispielsweise der Eingang «DruckVentil» des Geräts mit der Adresse 5 (Druckmesser) mit der Variablen «Gasdruck» im Gerät mit der Adresse 10 (Steuerung). So lassen sich alle PDO-Verbindungen (Process Data Object) im Prototypensystem festlegen.

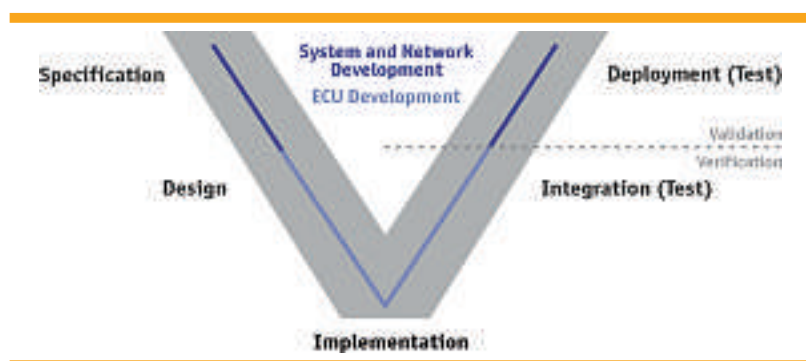


Bild 1: Darstellung des V-Modells

Autor

Dipl.-Ing. Mirko Tischer, Produktmanager für CANopen, mirko.tischer@vector-informatik.de, Vector Informatik GmbH

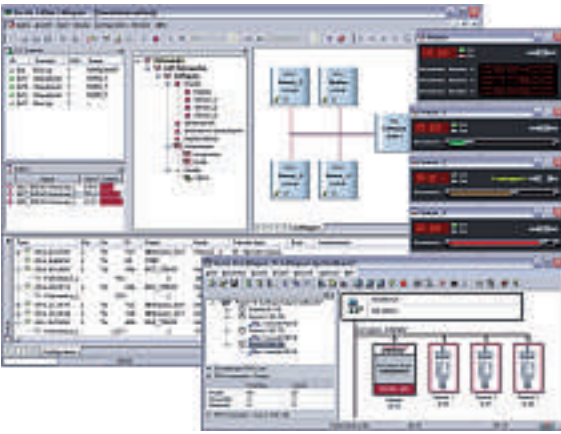


Bild 2: Beispielkonfiguration für ein einfaches, simuliertes CANopen-Netzwerk mit zentraler Steuerung und E/A-Knoten

Das Tool berechnet automatisch das resultierende Mapping, das nachträglich bei Bedarf veränderbar ist. Aus diesen Konfigurationsinformationen (DCF – Device Configuration File) generiert der Anwender nun die Prototypenumgebung, d.h. CANoe erzeugt für jedes reale Steuergerät ein Pendant mit identischen Kommunikationseigenschaften. Beim Start von CANoe steht der Kommunikationsteil der Prototypenumgebung bereits zur Verfügung (Bild 2).

Definition des Applikationsverhaltens

Das Applikationsverhalten der beteiligten Steuergeräte lässt sich nicht direkt aus den EDS-Dateien ableiten, da diese nur die Struktur eines Objektverzeichnisses abbilden. Die Formulierung des Applikationsverhaltens ist daher immer mit zusätzlichem Programmieraufwand verbunden. Die in CANoe integrierte Programmiersprache CAPL dient dem schnellen Programmieren des Steuergeräteverhaltens. Alternativ ist es möglich, das Steuergeräteverhalten in einer DLL zu kodieren, die man in die Prototypenumgebung unter CANoe einbindet. Auch Matlab/Simulink-Modelle sind problemlos integrierbar (Bild 3).

Testfunktionalität mit mehreren Stufen

Nach Fertigstellung des Prototypensystems folgen die notwendigen Tests für das gesamte Projekt. CANoe unterstützt den Anwender hier sowohl beim Erstellen der Tests als auch bei der Aufzeichnung und Auswertung. Die

für ein CANopen-System nötige Testfunktionalität umfasst mehrere Stufen:

- **Protokollebene:** Die Tests auf dieser Ebene stellen sicher, dass die CANopen-spezifischen Kommunikationsprotokolle der einzelnen Geräte innerhalb des Gesamtsystems gemäss der Spezifikation implementiert sind.
- **Kommunikationsebene:** Hier geht es nicht um die Richtigkeit des Protokolls, sondern um die korrekte logische Abfolge der (unabhängigen) Protokollsequenzen, zum Beispiel bei der Konfiguration von PDO. Die Beschreibung der PDO-relevanten Einträge im Objektverzeichnis muss eine bestimmte Reihenfolge einhalten.
- **Applikationsebene:** Applikative Tests prüfen den Zusammenhang zwischen den Prozessgrössen. Damit solche Zusammenhänge feststellbar sind, bedarf es folgender Voraussetzungen: Einerseits sind die Prozessgrössen über PDO auszutauschen und andererseits muss das System komplett konfiguriert sein. So lässt sich zum Beispiel der Zustand eines Ventils in Abhängigkeit einer Temperatur oder eines Drucks prüfen. Dabei wird offensichtlich, dass zur Formulierung des Tests Anwender-Know-how erforderlich ist.

Erstellen und Ausführen von Testprozeduren mit CANoe

Testabläufe formuliert man unter CANoe mithilfe der integrierten Programmierspra-

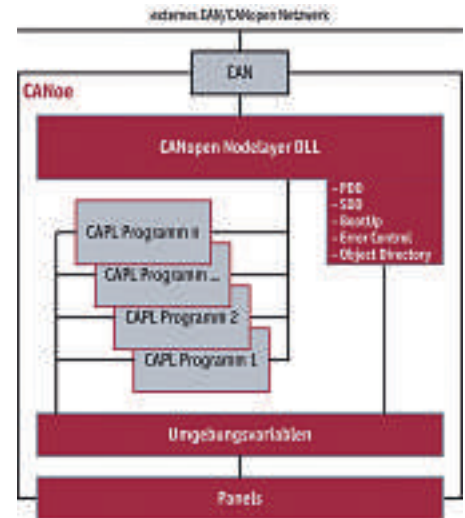


Bild 3: Darstellung der CANoe.CANopen-Umgebung

che CAPL. Jedes CAPL-Testmodul repräsentiert einen separaten Test, bestehend aus einzelnen Testfällen, die wiederum mehrere Testschritte umfassen. Während des Testlaufs arbeitet CANoe die einzelnen Testfälle sequenziell ab. Eine geeignete Testablaufsteuerung erlaubt das Überspringen oder Wiederholen einzelner Tests. Gleichzeitig lässt sich – quasi im Hintergrund – die Einhaltung weiterer Bedingungen und Einschränkungen überwachen. Sinnvoll ist dieses Vorgehen, wenn während des Wartens auf eine bestimmte Nachricht von Interesse ist, ob eine bestimmte andere Nachricht zyklisch am Bus gesendet wird.

Insbesondere bei automatisierten Tests ist es wichtig, die Ergebnisse der einzelnen Testschritte detailliert zu protokollieren. Weitere CAPL-Funktionen übernehmen das Schreiben von Testergebnissen in eine XML-Datei zur späteren Weiterverarbeitung. Testabläufe für CANoe lassen sich auch unter XML spezifizieren. Dies bietet Vorteile, wenn eine Vielzahl gleichartiger Testabläufe mit einem Werkzeug zu generieren ist. CANoe nutzt deshalb eine Reihe von in XML spezifizierten Testpatterns und verarbeitet diese entsprechend. <<

Infoservice

Vector Informatik GmbH
Ingersheimer Strasse 24, DE-70499 Stuttgart
Tel. 0049 711 806 700, Fax 0049 711 806 701 11
info@vector-informatik.de
www.vector-informatik.de



Walter Stalder,
Direktor
Wirtschafts-
förderung Luzern,
6005 Luzern

40 JAHRE ELEKTRONIKINDUSTRIE SCHWEIZ

Ausser der Gründung einer Familie, dem Bauen eines Eigenheims sowie vielen persönlichen (Lebens-)Erfahrungen im In- und Ausland gab es meiner Meinung nach zahlreiche grosse (Ver-)Änderungen in den letzten drei, vier Dekaden. Ich zähle dazu die Entwicklung der Multimedia-Landschaft (Radio – TV – Telefonie – Internet), die Umwälzungen in der Industrielandschaft – neue Firmen versus Firmen, die verschwinden, wie Kern Aarau, Wild, Elektrowatt usw., und die Konsolidierung in der globalen und lokalen Bauelementelandschaft – wenig grosse Global Player und Schweizer Nischenplayer. Meine Berufskarriere im Zeitraum 1980 bis 2006: Meine These dazu – was gestern war, gilt heute nicht mehr und was morgen ist wissen wir definitiv nicht.

