

Integration von Antrieben in CANopen-basierende Automatisierungssysteme

Prof. Dr.-Ing. K. Etschberger

Steinbeis-Transferzentrum Prozeßautomatisierung (stzp), Leibnizstr. 15, D-88250 Weingarten

etschberger@stzp.de

Das auf CAN basierende CANopen-Protokoll hat sich in wenigen Jahren zu einem der wichtigsten Netzwerkstandards für die Realisierung verteilter Automatisierungssysteme, besonders im Umfeld von Antriebsaufgaben entwickelt. Im Rahmen des Beitrags werden zunächst die besonderen Leistungsmerkmale des zugrundeliegenden CAN-Protokolls angesprochen sowie eine kurze Einführung zum CANopen-Standard gegeben. Anschließend wird die Beschreibung von Antrieben in Form des CANopen Antriebsprofils erläutert und ein neuartiger Stellantrieb mit CANopen-Schnittstelle vorgestellt.

1. Einführung

Neben den bereits seit längerem etablierten Feldbussystemen PROFIBUS und INTERBUS-S wird das auf CAN (Controller-Area-Network) basierende CANopen in zunehmendem Maße für die Realisierung verteilter Automatisierungssysteme eingesetzt. Begründet wird dies nicht nur durch die besonderen Leistungsmerkmale des CAN-Protokolls sondern auch durch die besonders kostengünstige Realisierbarkeit von CAN-Geräteschnittstellen, da Protokollcontroller und Transceiver-Bausteine auf Grund der hohen Stückzahlen der Automobilindustrie sehr preiswert am Markt angeboten werden, bzw. praktisch alle auf dem KFZ-Markt tätigen Hersteller von Mikrocontroller Derivaten mit integrierter CAN-Schnittstelle anbieten. Dies bedeutet, daß für die Realisierung einer CAN-Geräteschnittstelle im Wesentlichen lediglich ein zusätzlicher Transceiverbaustein erforderlich ist.

Im Vergleich mit den oben erwähnten Feldbuslösungen verfügt das CAN-Protokoll [ISO11898] jedoch auch über besondere Leistungsmerkmale, welche die Realisierung von verteilten intelligenten Automatisierungssystemen in besonderem Maße unterstützen. Mit dem auf CAN basierenden CANopen Standard steht darüber hinaus ein standardisiertes Kommunikationssystem für die Implementierung von verteilten Automatisierungssystemen zur Verfügung, welche vor allem auch die besonderen Anforderungen der Antriebstechnik in besonderem Maße berücksichtigt.

2. Die besonderen Leistungsmerkmale des CAN-Protokolls

Im Unterschied zu konventionellen Feldbussystemen, bei denen primär ein Datenaustausch zwischen zwei Netzknoten in Form einer teilnehmerorientierten Übertragung erfolgt, stellt CAN ein Nachrichtenverteilsystem nach dem Producer-Consumer-Prinzip dar. In Bild 2-1 ist das Grundprinzip der Datenübertragung in einem CAN-Netz dargestellt.

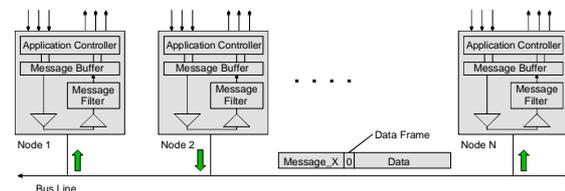


Bild 2-1: Prinzip der Nachrichtenübertragung bei CAN. Die von einem Teilnehmer gesendete Nachricht (im Bild z.B. Nachricht X, gesendet von Knoten 2) kann von allen anderen Teilnehmern übernommen werden.

CAN-Nachrichten werden über einen Nachrichtenidentifizier gekennzeichnet und können von jedem interessierten anderen Netzteilnehmer übernommen werden. Der Wert des Nachrichtenidentifiziers bestimmt gleichzeitig die Priorität der Nachricht be

züglich des Buszugriffs. Hierbei gilt, daß die Priorität einer Nachricht umso höher ist, je niedriger der Wert des im Standardformat 11-Bit langen Nachrichtenidentifiers ist.

Die Übertragung nach dem „Producer-Consumer-Prinzip“ hat viele Vorteile, wie etwa den gleichzeitigen Empfang einer Nachricht durch mehrere andere Teilnehmer. In der Antriebstechnik ist damit z.B. auf einfache Weise die Steuerung synchroner Achsbewegungen möglich.

Als weiteres wesentliches Merkmal des CAN-Protokolls ist dessen Multimasterfähigkeit zu erwähnen. Dies bedeutet, daß jeder Netzknoten gleichberechtigten Zugang zum Bus hat, wobei die Zugangsberechtigung allein über die Priorität der zu sendenden Nachricht bestimmt wird. In CAN-basierenden Netzwerken ist deshalb eine direkte Kommunikation zwischen allen Teilnehmern eines Netzwerks möglich ($n : n$ Kommunikationsstrukturen). Die Bestimmung der Zugangsberechtigung über die Priorität einer Nachricht wiederum ermöglicht die Übertragung von Nachrichten hoher Priorität mit sehr kurzen Latenzzeiten, selbst bei hoher Busbelegung mit niederpriorigen Nachrichten.

Weitere wichtige Merkmale des CAN-Protokolls sind u.a.:

- Sicherstellung einer netzwerkweiten Datenkonsistenz
- Sicherstellung einer sehr hohen Datenintegrität durch besonders wirkungsvolle Maßnahmen zur Erkennung von Übertragungsfehlern
- Sehr kurze Fehlererholzeit
- Funktionsfähigkeit auch unter stark gestörten Umgebungsbedingungen auf Grund der kurzen Nachrichtenlänge (maximal 8 Byte Nutzdaten pro Nachricht)
- Maximale Datenrate 1 Mbit/s bei 30 m Buslänge. Ausgedehntere Netzwerke sind möglich bei entsprechend reduzierten Datenraten

Die hohe Leistungsfähigkeit des CAN-Protokolls ergibt sich auch aus der pro Zeiteinheit übertragbaren Anzahl von Nachrichten. Diese liegt bei einer Datenrate von 1 Mbit/s und 8 Datenbytes pro Nachricht bei über 8000 Nachrichten/sec, die Latenzzeit

für die höchstpriorige Nachricht liegt in diesem Fall bei ca. 130 Mikrosekunden.

Eine umfassende Darstellung des CAN-Protokolls findet man z. B. in [CAN99].

3. CANopen, eine standardisierte Systemlösung für verteilte, intelligente Systeme

Das CAN-Protokoll regelt lediglich die Aspekte der in Schicht 1 und 2 des OSI-Modells der Datenkommunikation beschriebenen Aufgaben. In den Funktionsbereich der Schicht 1 fallen z. B. die Art der physikalischen Signaldarstellung, Signalcodierung, Bittiming und Bitsynchronisation, in den Funktionsbereich von Schicht 2 fallen die Regelung des Buszugriffs, die Formatierung von CAN-Telegrammen, Datenübertragungssicherung und Fehlererkennung sowie die Eingrenzung defekter Teilnehmer. Über die Funktionalität der Schichten 1 und 2 hinausgehend, welche letztlich lediglich die gesicherte Übertragung von Nachrichten bereitstellen, sind für die Realisierung von verteilten Systemen jedoch noch eine Reihe weiterer Funktionen erforderlich. Solche Anforderungen sind z.B. Maßnahmen im Zusammenhang mit der Überwachung des Netzwerks, der Zuordnung von Nachrichtenidentifiern auf Nachrichten oder die Bereitstellung allgemein zweckmäßiger Systemdienste. Besondere Anforderungen ergeben sich vor allem dann, wenn das Kommunikationssystem den Betrieb von Geräten unterschiedlicher Hersteller ermöglichen soll, und diese Geräte gegeneinander austauschbar sein sollen. Um diesen Anforderungen gerecht zu werden, wurde CANopen von der CAN-in-Automation User Group CiA¹ spezifiziert. CANopen ist heute neben dem von Rockwell Automation über die ODVA² spezifizierten DeviceNet das wichtigste höhere Protokoll (Profil) für CAN-basierende Automatisierungslösungen. Im Folgenden werden einige der wichtigsten Merkmale des CANopen-Standards vorgestellt. Eine detaillierte Einführung findet man in [CAN99].

Zentrales Element des CANopen-Standards ist die Beschreibung aller Kommunikationseigenschaften sowie der eigentlichen Gerätefunktionalität über ein "Objektverzeichnis". Das Objektverzeichnis (OV) ist

¹ CiA International Headquarters, Am Weichselgarten 26, D-91058 Erlangen. <http://www.can-cia.de>

² Open DeviceNet Vendors Association. <http://www.odva.org>

unterteilt in einen Bereich, welcher allgemeine Angaben über das Gerät, wie Identifikation, Hersteller, usw. sowie dessen Kommunikationsparameter enthält, und einen Teil, der die spezifische Gerätefunktionalität beschreibt. Die Identifizierung eines Eintrags ("Objekt") des Objektverzeichnisses erfolgt über einen 16-Bit Index und einen 8-Bit Subindex. Über die Einträge des Objektverzeichnisses werden die „Anwendungsobjekte“ eines Gerätes, wie z.B. Ein- und Ausgangssignale, Geräteparameter, Gerätefunktionen oder Netzwerkvariablen in standardisierter Form über das Netzwerk zugänglich. Bild 3-1 zeigt das grundsätzliche Modell eines CANopen-Gerätes mit einem Kommunikationsinterface zur Anbindung des Gerätes an das Netzwerk, dem eigentlichen Anwendungsprozeß sowie dem Objektverzeichnis als Schnittstelle zwischen Netzwerk und Anwendungsprozeß.

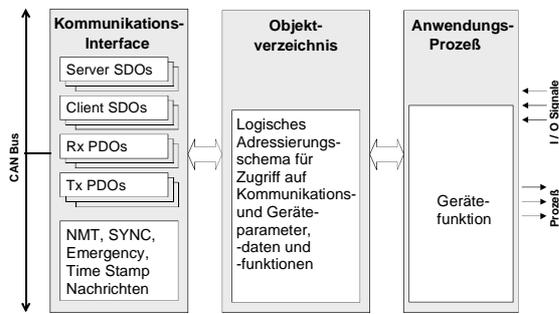


Bild 3-1: CANopen Gerätemodell

Analog zu anderen Feldbussystemen unterscheidet CANopen ebenfalls zwei grundsätzliche Datenübertragungsmechanismen: Den schnellen Austausch kurzer Prozeßdaten über sog. "Prozeßdatenobjekte" (PDOs, Process Data Objects) sowie den Zugriff auf Einträge des Objektverzeichnisses über sog. "Servicedatenobjekte" (SDOs, Service Data Objects). Die Letzteren dienen in erster Linie zur Übertragung von Parametern während der Gerätekonfiguration sowie allgemein zur Übertragung längerer Datenbereiche (z.B. Programme).

Prozeßdatenobjekte werden im Allgemeinen ereignisorientiert, zyklisch oder auf Anforderung als Broadcastnachrichten mit impliziter Identifikation über den Nachrichtenidentifizier, d.h. ohne zusätzlichen Protokoll-overhead, übertragen. In einer PDO können maximal 8 Byte Daten übertragen werden. In

Verbindung mit einer Synchronisationsnachricht kann das Senden sowie die Übernahme von PDOs netzwerkweit synchronisiert werden ("Synchrone PDOs"). Die Zuordnung von Anwendungsobjekten auf eine PDO (Übertragungsobjekt) wird über eine, im OV abgelegte Strukturbeschreibung ("PDO-Mapping") beschrieben. Da diese über den Bus änderbar ist, kann diese Zuordnung auf einfache Weise auf die jeweiligen Einsatzanforderungen eines Gerätes angepaßt werden. Bild 3-2 zeigt das Protokoll für das Senden („Schreiben“) einer PDO.

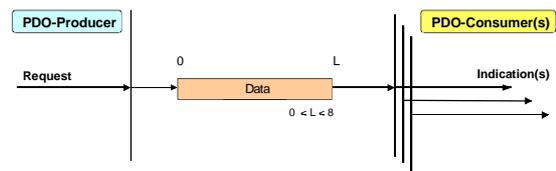


Bild 3-2: Protokoll für das Senden einer PDO.

Die Bedeutung der übertragenen Information ist anwendungsspezifisch und im Objektverzeichnis der Geräte in Form des sog. „PDO-Mapping“ beschrieben (Bild 3-3).

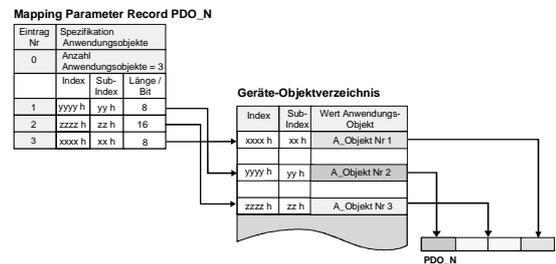


Bild 3-3: Beschreibung der in einer PDO übertragenen Information über die zugehörige PDO-Mapping-Struktur. Diese beschreibt, welche Anwendungsobjekte (Daten) in welcher Reihenfolge im Datenfeld der zugehörigen CAN-Nachricht übertragen werden. Die Anwendungsobjekte werden hierbei über Index/Subindex ihres Eintrags im Geräte-Objektverzeichnis spezifiziert.

Das Senden einer PDO kann auf drei alternative Arten getriggert werden, nämlich ereignis- oder zeitgesteuert beim Auftreten eines spezifizierten Ereignisses oder nach Ablauf einer bestimmten Zeitintervalls, auf Anforderung durch einen anderen Teilnehmer oder ausgelöst durch das Eintreffen einer sog. „Synchronisationsnachricht“ (Bild 3-4). Die

Synchronisationsnachricht wird als hochpriorie Nachricht ohne Datenteil durch einen hierfür verantwortlichen Netzteilnehmer (Synchronisations-Producer) erzeugt. Im Falle von Nachrichten, welche nur in Verbindung mit dem vorherigen Eintreffen einer Synchronisationsnachricht gesendet werden, spricht man auch von „Synchronen PDOs“, im Gegensatz zu „Asynchronen PDOs“.

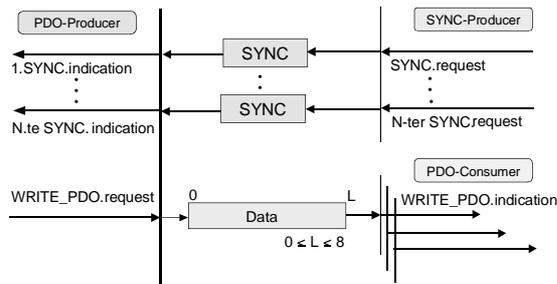


Bild 3-4: Senden und Empfangen einer Synchronen PDO. Im synchronen Betrieb werden die zu sendenden Prozeßdaten vom Producer nach Empfang einer einstellbaren Anzahl von Synchronisationsnachrichten aktualisiert und gesendet. Ein Consumer übergibt die vorher empfangenen Prozeßdaten erst nach Empfang der nächsten Synchronisationsnachricht an seinen Anwendungsprozeß.

Für den Betrieb von verteilten Antriebssystemen ist die netzwerkweit synchrone Erfassung von Istpositionen bzw. die Übernahme von Sollpositionen auf Basis von synchronen PDOs von besonderem Interesse (Bild 3-5). Hierbei werden die aktuellen Istpositionen mit dem Eintreffen der Synchronisationsnachricht in allen Antrieben praktisch gleichzeitig erfaßt und anschließend entsprechend der Priorität der zugeordneten synchronen PDO über den Bus gesendet. Mit dem Eintreffen einer Sync-Nachricht werden gleichzeitig die im vorhergehenden Übertragungszyklus empfangenen neuen Sollwerte von den Antrieben übernommen. Da synchrone PDOs im Allgemeinen mit höherer Priorität wie asynchrone PDOs gesendet werden, wird das Senden von synchronen PDOs auf einen bestimmten Anteil („Synchronous Window Length“) der Synchronisationszykluszeit („Communication Cycle Period“) begrenzt. Auf diese Weise ist sichergestellt, daß auch asynchrone Nachrichten übertragen werden können.

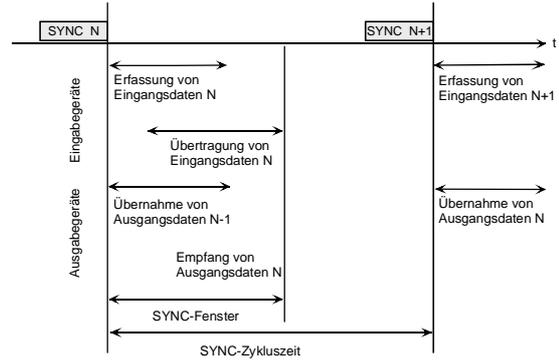


Bild 3-5: Synchrone Operationen unter CANopen.

Auf alle Einträge des Objektverzeichnisses eines Gerätes kann über SDO-Kanäle zugegriffen werden. Über einen SDO-Kanal kann ein Gerät (z.B. ein Konfigurations-Tool) auf das Objektverzeichnis eines anderen Gerätes zugreifen d.h. um OV-Einträge zu lesen oder zu schreiben. Auf diese Weise können z.B. die Kommunikations- oder Funktionsparameter eines Gerätes eingestellt, Daten von einem Gerät gelesen oder bestimmte Gerätefunktionen ausgelöst werden. Für den Betrieb eines SDO-Kanals sind zwei SDOs erforderlich, da der Datenaustausch in diesem Fall in bestätigter Form erfolgt (Bild 3-6). Die Adressierung eines Objektverzeichniseintrages erfolgt durch Angabe dessen 16-Bit Index und 8-Bit Subindex. Über ein Segmentierungsprotokoll können über SDO-Kanäle Nachrichten unbegrenzter Länge übertragen werden. Die Übertragung von SDO-Nachrichten ist deshalb mit einem zusätzlichen Protokolloverhead verbunden.

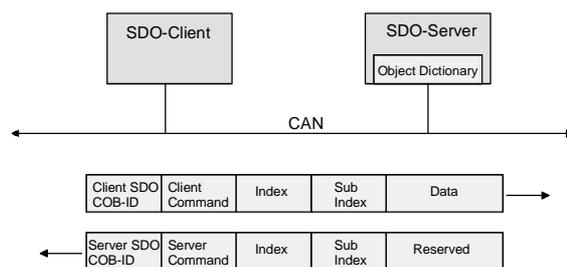


Bild 3-6: Schreiben eines OV-Eintrags über das SDO-Protokoll (Expedited Transfer). Hierzu wird im Datenteil der ersten CAN-Nachricht neben der Funktion („Schreiben OV-Eintrag“) der zu schreibende OV-Eintrag durch Angabe von dessen Index und Subindex spezifiziert. In der Antwortnachricht wird die Übertragung der Daten bestätigt. Mit dem dargestellten Kurzprotokoll

können OV-Einträge bis zu 4 Byte geschrieben werden. Der Transfer längerer Einträge erfolgt über ein Segmentierungsprotokoll.

Für die Meldung von Gerätefehlern sind standardisierte ereignisorientierte Alarmnachrichten („Emergency-Messages“) hoher Priorität vorgesehen, über eine zentrale Zeit-Nachricht kann eine gemeinsame Systemzeit zur Verfügung gestellt werden.

Für Koordination und Überwachung der Kommunikationsfähigkeit der einzelnen Netzteilnehmer ist das sog. „Netzwerkmanagement“ (NMT) verantwortlich. Unter CANopen ist dieses in Form eines logischen Master-Slave-Systems implementiert. Hierbei übernimmt ein Netzteilnehmer die Funktion des Netzwerkmanagement-Masters. Alle anderen Teilnehmer verhalten sich bezüglich des Netzwerkmanagements als Slaves und werden im Rahmen des sog. „Node-Guarding“ vom NMT-Master zyklisch auf ihre Kommunikationsfähigkeit überprüft. Alternativ besteht auch die Möglichkeit, daß die Netzknoten ihre Kommunikationsfähigkeit selbst über eine sog. „Heartbeat-Message“ zyklisch allen anderen Teilnehmern mitteilen. Über den NMT-Master besteht auch die Möglichkeit, den Kommunikationszustand einzelner Netzknoten oder auch aller Netzknoten zu steuern.

Da CAN ein nachrichtenorientiertes Protokoll ist, müssen den PDOs und SDOs Nachrichtenidentifizier zugeordnet werden. Für einfache 1 : n Kommunikationsstrukturen kann dies über vordefinierte, aus der Knotennummer abgeleitete Identifizier erfolgen. Bild 3-7 zeigt dieses Prinzip der Identifizierzuordnung. Über die vordefinierte Identifizierverteilung kann ein Netzwerk mit einem Steuergerät und bis zu 127 Geräten betrieben werden, wobei für jeden Netzknoten bis zu vier Sende- und Empfangs-PDOs, eine Node-Guarding-Meldung, eine Emergency-Meldung sowie ein SDO-Kanal zur Verfügung steht.

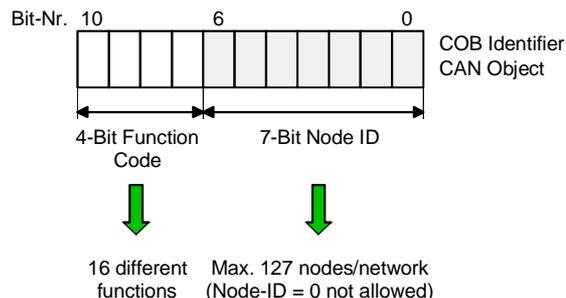


Bild 3-7: Identifizierzuordnung für 1 : n Kommunikationsstrukturen über „voreingestellte“ Identifizier. Pro Knoten sind hierbei 16 unterschiedliche „Funktionen“ möglich.

Für komplexere Kommunikationsstrukturen, d.h. mehr als 4 Sende- oder Empfangs-PDO oder bei direkter Kommunikation zwischen den Netzknoten (n : n Kommunikationsstrukturen) können diese durch Einrichtung der hierfür erforderlichen PDOs und SDOs und Zuordnung von Nachrichtenidentifizier in den Objektverzeichnissen der Geräte realisiert werden. Hierfür stehen komfortable Konfigurationswerkzeuge zur Verfügung [IXX].

4. Beschreibung von Geräten über CANopen-Profil

Die CANopen Profildfamilie besteht aus Geräteprofilen für die wichtigsten Gerätetypen der Automatisierungstechnik, wie digitale und analoge Ein/Ausgabemodule [CIA-401], Antriebe [CIA-402], programmierbare Steuerungen [CIA-405], Sensoren und Regler [CIA-404], Encoder [CIA-406], Bedien- und Anzeigegeräte [CIA-403] sowie dem, für alle Geräte gültigen Kommunikationsprofil [CIA-301]. In den Geräteprofilen wird die Funktionalität von Standardgeräten des jeweiligen Typs festgelegt. Hierbei besteht die Möglichkeit der zusätzlichen Beschreibung herstellerepezifischer Erweiterungen. Im Kommunikationsprofil werden die allgemeinen Gerätedaten sowie die Kommunikationsparameter beschrieben. Kommunikations- und Geräteprofil sind im Objektverzeichnis eines CANopen-Gerätes beschrieben. Dieses ist in einem Gerät nur soweit wie notwendig implementiert und wird vom Gerätehersteller zusätzlich in Form eines sog. „Electronic Data Sheet“ in EDV-gerechter Form z.B. für die Verwendung in einem CANopen-Konfigurationstool bereitgestellt.

Bild 4-1 zeigt die grundsätzliche Struktur des CANopen-Objektverzeichnisses (OV). Das OV besteht aus den Bereichen für die Definition von Datentypen, die Beschreibung von allgemeinen Gerätedaten und Kommunikationsparametern („Kommunikationsprofil“) sowie die Beschreibung von herstellerspezifischen und standardisierten Geräteeigenschaften. Das Kommunikationsprofil [CIA-301] ist bei allen Geräten in gleicher Form aufgebaut.

Index	Object	
0001H-009FH	Data Type	Common to any Device
00A0H-0FFFH	Reserved	
1000H-1FFFH	Communication Profile Area	
2000H-5FFFH	Manufacturer Specific Profile Area	Device specific
6000H-9FFFH	Standardized Device Profile Area	
A000H-FFFFH	Reserved	

Bild 4-1: Struktur CANopen-Objektverzeichnis

Es gibt einfache und komplexe OV-Einträge in Form von Arrays oder Records. Einfache Einträge werden durch einen 16-Bit Index adressiert, komplexe durch 16-Bit Index und 8-Bit Subindex. Jeder Eintrag ist durch eine Anzahl von Attributen wie Namen, Datentyp, Zugriffsart oder voreingestellter Wert beschrieben.

5. CANopen Geräteprofil für Antriebe

Vorrangiges Ziel dieses Profils [CIA-402] ist es, eine gleichartige Beschreibung der über die Kommunikationsschnittstelle sichtbaren Funktionalität von Antrieben, d.h. Schrittmotoren, Frequenzumrichtern und Servoantrieben bereitzustellen. Das Profil lehnt sich eng an das auf INTERBUS-S basierende DRIVECOM-Profil 21 für Antriebe an und erleichtert damit Herstellern und Anwendern solcher Geräte die Nutzung eines bereits vertrauten Profils. Auf Grund der Vielfalt von spezifischen Eigenschaften, mit denen die Hersteller von Antrieben ihre Geräte ausstatten, ist eine einheitliche Beschreibungsmethode erforderlich, welche es gestattet, auch die herstellerspezifischen Leistungsmerkmale darzu-

stellen. Das Profil enthält deshalb neben zwingend geforderten Leistungsmerkmalen (Parametern, Funktionen) für die Beschreibung von Standardgeräten auch optionale und herstellerspezifische Parameter. Auf diese Weise ist zumindest für Standardaufgaben eine Austauschbarkeit von Geräten möglich. Für die Hersteller von Antrieben bedeutet die Verfügbarkeit eines standardisierten Anwendungsprofils zumindest, daß nicht für jeden Kunden ein spezifisches Protokoll implementiert werden muß.

Das CANopen-Antriebsprofil CiA DS 402 beschreibt einen Antrieb in die folgenden Teilbereiche gegliedert:

- Allgemeine Motor- und Antriebsdaten
- Gerätesteuerung
- Spezifikation von Umrechnungsgrößen
- Parameter und Steuerung in den einzelnen Betriebsarten

Im OV-Indexbereich von 6402H bis 60FEH werden die allgemeinen Daten des Motors, im Bereich zwischen 6500H und 65FFH die Daten der Antriebssteuerung beschrieben. Für den Motor sind dies z.B. Motortyp, Herstellername, Katalognummer Motorhersteller, Datum der letzten Motorwartung, Serviceintervall sowie technische Motordaten. Die Beschreibung der Steuereinheit umfaßt die Angabe der von der Steuerung unterstützten Betriebsarten, Herstellername, Katalognummer Antriebshersteller sowie technische Daten der Steuerung. Im allgemeinen Beschreibungsteil sind außerdem die mit der Antriebssteuerung zur Verfügung gestellten digitalen Ein- und Ausgänge sowie deren Verwendung (z.B. Bereichsschalter, Endschalter, Bremsenaktivierung) enthalten.

Bei der Steuerung eines Antriebs wird unterschieden in die Steuerung des Gerätes entsprechend der zugrundegelegten Geräte-Zustandsmaschine sowie der Steuerung der eigentlichen Gerätefunktionen im operationellen Zustand entsprechend dem eingestellten Betriebsmodus (z.B. "Geschwindigkeitssteuerung").

Bild 5-1 zeigt das in CiA DS 402 zugrundegelegte Zustandsdiagramm eines Antriebs. Die verschiedenen Zustände sowie die zwischen den Zuständen möglichen Zustandsübergänge beschreiben z.B. in

welcher Form ein Antrieb stufenweise eingeschaltet werden muß oder in welcher Form ein Antrieb nach dem Auftreten eines Fehlerzustandes wieder in Betrieb gesetzt werden kann. Die Zustandsübergänge werden gesteuert über den Empfang eines "Steuerwortes" (Control Word) oder über interne Ereignisse (z.B. Fehlerzustände). Der Zustand eines Antriebs wird über das "Statuswort" (Status Word) entsprechend Bild 5-2 gemeldet.

In den eigentlichen Betriebszustand ("Operation enabled") geht ein Antrieb z.B. dann (Bild 5-1, Übergang Nr. 4), wenn der Antrieb das Kommando "Enable-Operation" von einem Hostcontroller erhalten hat, die Energieversorgung des Antriebs eingeschaltet ist sowie die Betriebsbereitschaft der Antriebsverstärker gegeben ist (Zustand "Switched-On"). Bei Auftreten eines Fehlers geht der Antrieb in den Zustand "Fault-Reaction-Active" (Übergang Nr. 13). In diesem Zustand werden die für den Antrieb im Falle eines nicht fatalen Fehlers eingestellten Aktionen durchgeführt (z.B. Abschalten Antrieb). Anschließend geht der Antrieb in den Zustand "Fault". In diesem Zustand ist die Antriebsfunktion abgeschaltet. Der Übergang (Nr. 15) in den Zustand "Switch-On-disabled" erfolgt nach Erhalt eines "Fault-Reset"-Kommandos vom Hostcontroller und sofern kein weiterer Fehler mehr vorliegt. Für eine weitergehende Beschreibung der Zustandsmaschine wird auf [CIA-402] verwiesen.

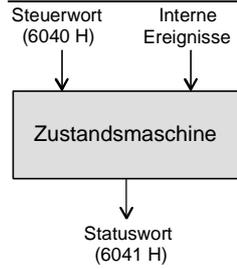


Bild 5-2: Steuerung der Zustandsmaschine eines CANopen-Antriebes

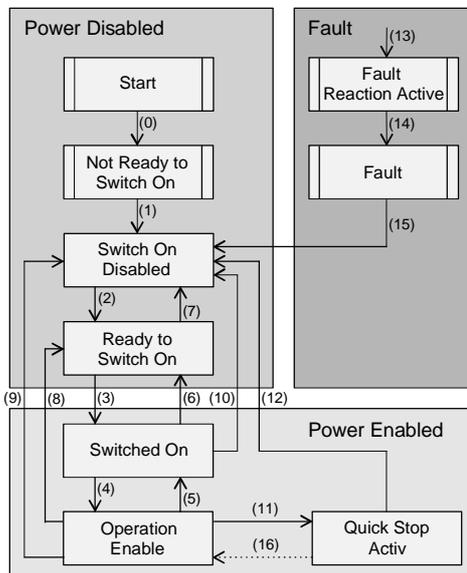


Bild 5-1: Zustandsmaschine eines Antriebes

Steuerwort

Über das 16-Bit-Steuerwort kann ein Hostcontroller über ein entsprechendes Kommando direkt den Zustand des Antriebs steuern. Bild 5-3a zeigt die Belegung des Steuerworts. Über die Bitgruppe 4-6 können betriebsartenabhängige Funktionen (z.B. für den "Profile Position Mode" die Übergabe eines neuen Vorgabewertes) ausgeführt werden. Das Steuerwort wird in den ersten beiden Bytes jeder Kommandonachricht gesendet.

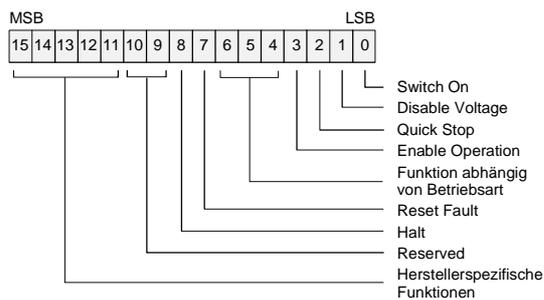


Bild 5-3a: Belegung Steuerwort

Statuswort

Über das 16-Bit-Statuswort wird der aktuelle Status eines Antriebs beschrieben (Bild 5-3b). Das Statuswort wird in den ersten beiden Bytes jeder von einem Antrieb gesendeten PDO übertragen.

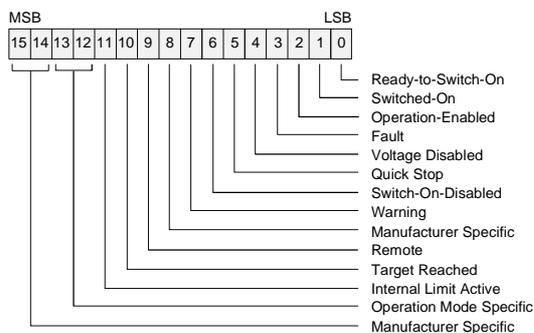


Bild 5-3b: Belegung Statuswort

"Fault-Reaction-Option") werden die, in Verbindung mit den verschiedenen Zustandsübergängen durchzuführenden Aktionen spezifiziert.

Über das Objekt "Modes-of-Operation" wird die Betriebsart des Antriebs eingestellt, über das Objekt "Display-Modes-of-Operation" die aktuelle Betriebsart angezeigt.

Antriebsbetriebsarten

CiA DSP402 spezifiziert die wichtigsten der im Antriebsbereich vorzufindenden Antriebsbetriebsarten. Derzeit sind drei positionsbezogene Betriebsarten (Homing Mode, Profile Position Mode, Interpolated Position Mode), zwei geschwindigkeitsbezogene (Profile Position Mode, Velocity Mode) sowie eine momentenbezogene Betriebsart (Profile Torque Mode) spezifiziert. Die Beschreibung der jeweils zugehörigen Lage-, Geschwindigkeits- und Momentenregler ist derzeit lediglich für die Lageregelung näher spezifiziert.

In den weiteren Objekten dieser Gruppe ("Shutdown-", "Disable-Operation-", "Quick-Stop-", "Stop-")

Objektverzeichnis

Tabelle 5-1 zeigt die Nutzung des standardisierten Geräteprofilbereichs für die Beschreibung von bis zu 8 Achsen sowie optionaler mit einem Antrieb verbundenen E/A-Funktionalität.

Tabelle 5-1: Beschreibung von Antrieben im CANopen-Objektverzeichnis

Index	Beschreibung von
6000H-67FFH	Achse 1
6800H-6FFFH	Achse 2
9800H-9FFFH	Achse 8

PDO-Mapping

Da ein Antrieb mehrere Betriebsarten unterstützen kann, werden 64 Empfangs- und Sende-PDOs pro Achse vorgesehen. Über die Empfangs-PDO erfolgt die Steuerung eines Antriebs durch den Host-Controller, über die Sende-PDOs die Bereitstellung des aktuellen Status eines Antriebs an den Host-Controller.

In Tabelle 5-2 sind die für die erste Achse definierten Empfangs-PDOs beschrieben. Die PDOs für die weiteren Achsen werden jeweils mit einem Offset von 64 weitergezählt, die entsprechenden Objekte über einen Index-Offset von 800H. Zwingend vorgeschrieben ist jeweils nur die erste Empfangs- und Sende-PDO.

Tabelle 5-2: Empfangs-PDOs eines CANopen-Antriebs

PDO-Nr.	Mapping Object Index	Beschreibung	Objekttyp	Datentyp
1	6040 H	Steuerwort	VAR	Integer 16
2	6040 H	Steuerwort	VAR	Integer 16
	6060 H	Betriebsart	VAR	Integer 8
3	6040 H	Steuerwort	VAR	Integer 16
	607A H	Soll-Position	VAR	Integer 32
4	6040 H	Steuerwort	VAR	Integer 16
	6081 H	Sollgeschwindigkeit (Profilbetrieb)	VAR	Integer 32
5	6040 H	Steuerwort	VAR	Integer 16
	6071 H	Soll-Moment	VAR	Integer 16
6	6040 H	Steuerwort	VAR	Integer 16
	6042 H	Soll-Geschwindigkeit	VAR	Integer 16
7	6040 H	Steuerwort	VAR	Integer 16
	60FE H	Digitale Ausgänge	Record	
8	6040 H	Steuerwort	VAR	Integer 8
	6060 H	Betriebsart (Broadcasting)	VAR	
9-20		Reserviert		
21-64		Herstellerspezifisch		

Bild 5-5 zeigt als Beispiel das Format einer Nachricht zur Vorgabe der neuen Sollposition.

Identifizier	Steuerwort	Soll-Position
--------------	------------	---------------

Bild 5-5: Format einer Nachricht zur Übertragung einer neuen Sollposition an einen CANopen-Antrieb

Über die Sende-PDOs übermittelt ein Antrieb seinen aktuellen Zustand ereignisgesteuert an den steuernden Host-Controllers bzw. kann der Zustand des Antriebs von einem Host-Controller abgefragt werden (synchron oder asynchron). In Tabelle 5-3 sind die spezifizierten Nachrichten zusammengefaßt.

Tabelle 5-3: Sende-PDOs eines CANopen-Antriebs

PDO-Nr.	Mapping Object Index	Beschreibung	Objekt-typ	Datentyp
1	6041 H	Statuswort	VAR	Integer 16
2	6041 H	Statuswort	VAR	Integer 16
	6061 H	Eingestellte-Betriebsart	VAR	Integer 8
3	6041 H	Statuswort	VAR	Integer 16
	6064 H	Ist-Position	VAR	Integer 32
4	6041 H	Statuswort	VAR	Integer 16
	606C H	Ist-Geschwindigkeit (Profilbetrieb)	VAR	Integer 32
5	6041 H	Statuswort	VAR	Integer 16
	6077 H	Ist-Moment	VAR	Integer 16
6	6041 H	Statuswort	VAR	Integer 16
	6044 H	Ist-Geschwindigkeit	VAR	Integer 16
7	6041 H	Statuswort	VAR	Integer 16
	60FD H	Digitale Eingänge	Record	
8-20		Reserviert		
21-64		Hersteller-spezifisch		

6. Stellantrieb neuer Technologie mit CANopen-Schnittstelle

Als Beispiel eines Stellantriebs neuester Technologie zeigt Bild 6-1 den Stellantrieb IclA der Firma SIG Positec. Bezüglich der besonderen weiteren Leistungsmerkmale des Antriebs wird auf das Datenblatt des Hersteller verwiesen [SIGpos].

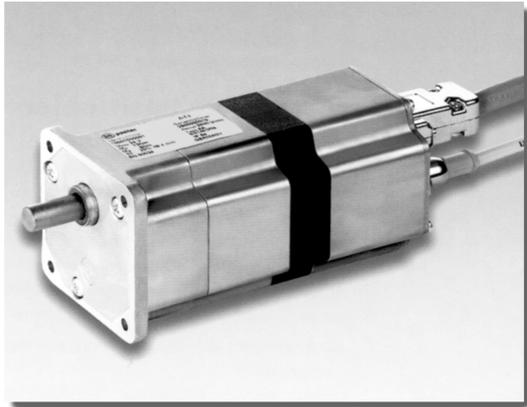


Bild 6-1: Stellantrieb mit CANopen-Schnittstelle (Foto: SIG positec)

Bei diesem System sind alle Einheiten des Antriebs, d.h. bürstenloser Gleichstrom-Motor, Getriebe, Istwerterfassung und Steuereinheit mit CANopen-Schnittstelle komplett in einem Gehäuse untergebracht. Neben der 24 V Stromversorgungsschnittstelle benötigt der Antrieb lediglich zwei Busleitungen für den Anschluß an das CAN/CANopen-Netzwerk. Über dieses kann der Antrieb für die verschiedenen Einsatzbedingungen konfiguriert und parametrieren werden. Außerdem ermöglicht der in der Antriebssteuereinheit integrierte Downloadmechanismus das Laden kundenspezifischer Antriebs-Steuerprogramme. Die Implementierung der CANopen-Geräteschnittstelle erfolgte auf Basis des Protokollstacks der Firma IXXAT Automation [IXX].

7. Literaturangaben

- [ISO11899] ISO-WD 11898-1: Road vehicles - Interchange of digital information - Part 1: Controller Area Network data link layer and medium access control, 1999.
Part 2: High Speed medium access unit and medium dependent interface, 1999
- [CIA-301] CANopen Communication Profile for Industrial Systems, CiA DS 301, Version 4.0, Mai 1999
- [CIA-302] Framework for Programmable CANopen Devices, CiA DSP 302, Version 1.0, March 1998
- [CIA-401] CANopen Device Profile for I/O Modules. CiA DSP 401, Version 1.4, Version 1996
- [CIA-402] CANopen Device Profile Drives and Motion Control. CiA DSP 402, Version 1.0, May 1997
- [CIA-403] CANopen Device Profile for Human Machine Interfaces. CiA DSP 403, Version 1.0, October 1998
- [CIA-404] CANopen Device Profile Regler. CiA DSP 404. Version 1.0
- [CIA-405] CANopen Device Profile for IEC 1131 Programmable Devices, CiA DSP 405, Version 1.0, March 1998
- [CIA-406] CANopen Device Profile Encoder, CiA DSP 404, Version 1.0
- [CAN99] CAN - Grundlagen, Protokolle, Bausteine, Anwendungen. Hanser-Verlag München-Wien, 10/99
- [IXX] <http://www.ixxat.de>
- [SIGpos] <http://www.sig-positec.de>