

PLCcore-E660 CODESYS mit EtherCAT-SoftMotion

Dokumentenversion: 1.0
Datum: 03.11.2014 11:43:00
Bearbeiter: -np

1 Einleitung

Die Steuerung von Antrieben ist eine häufig vorkommende Aufgabe in Automatisierungsprojekten. Oft geht es darum, Antriebe unterschiedlicher Hersteller und verschiedenster Natur (Schrittmotoren, AC- und DC-Servomotoren u. a.) in einer Automatisierungsanlage gemeinsam zu betreiben. Meist liegen diese Antriebe noch räumlich weit auseinander, so dass eine Feldbus-Lösung für die Antriebssteuerung angebracht ist. Die vorliegende Schrift beschreibt den Betrieb zweier unterschiedlicher Servomotoren von verschiedenen Herstellern an einer SPS in Form eines SYS TEC PLCcore-E660 mit einem CODESYS-Laufzeitsystem. Die Antriebe sind dabei über den Feldbus EtherCAT mit der SPS verbunden. Zur Aufwandsminimierung bei der Programmerstellung werden die SoftMotion-Bibliotheken des CODESYS-Laufzeitsystems im Projekt verwendet.

2 Prinzip

2.1 Kommunikationsmechanismus

Die Antriebe im Projekt und die SPS selbst verfügen über eine Feldbus-Anschaltung. Bei EtherCAT sind dies Ethernet-Interfaces, die das EtherCAT-Protokoll betreiben (siehe hierzu http://www.ethercat.de/download/documents/EtherCAT_EAP_DE.pdf). In der Regel haben die EtherCAT-Feldbusanschlungen der Antriebe zwei Slots zum Anschluss eines Ethernet-Kabels. Das macht es möglich, dass die Netztopologie je nach Aufgabe unterschiedlich sein kann. Möglich sind z. B.:

- Sternstruktur über einen Hub oder einen Switch
- Linienstruktur – die Ethernet-Verbindung verläuft von der SPS zum ersten Antrieb, von dort aus über den zweiten Slot zum nächsten Antrieb usw.
- Gemischte Topologien – Stern- und Linienstrukturen gemeinsam in einem Netz

Steuerinformationen von der SPS zu den Antrieben und Statusinformationen von den Antrieben zur SPS werden per EtherCAT-Protokoll übertragen. Es herrscht ein strenges Master-Slave-Prinzip für die Kommunikationssteuerung. Dabei sind die Antriebe die EtherCAT-Slaves und die SPS der EtherCAT-Master.

Für den Betrieb der Antriebe über EtherCAT unter einem CODESYS-Laufzeitsystem der SPS gibt es für die Antriebe von den Herstellern entsprechend bereitgestellte Bibliotheken, die in das Steuerungsprojekt eingebunden werden müssen. Da diese sich jedoch von Hersteller zu Hersteller und abhängig vom Antriebstyp stark unterscheiden, ist ein Betrieb unterschiedlicher Antriebe an einer SPS vom Anwender recht kompliziert zu handhaben. Es gibt in CODESYS jedoch die Möglichkeit, die Zugriffe

auf die einzelnen Antriebe zu vereinheitlichen und noch weiter zu vereinfachen. Dazu dienen die im CODESYS verwendbaren SoftMotion-Bibliotheken, deren Wirkprinzip folgend kurz erklärt wird.

2.2 SoftMotion

Bei SoftMotion ist jeder verwendete Antrieb eine Achse mit entsprechend parametrierbaren Eigenschaften (z. B. rotatorisch oder linear mit entsprechenden Koordinaten). Um Hardware-Einheiten braucht sich der Anwender nicht zu kümmern – das übernehmen im CODESYS verfügbare SoftMotion-Bibliotheken, die herstellerabhängig die hardwarespezifischen Bibliotheken mit den darunter liegenden Treibern für die einzelnen Antriebe benutzen.

Die dem Anwender damit zur Verfügung stehenden SoftMotion-Funktionsblöcke setzen auf dem Standard PLCopen (siehe auch http://www.plcopen.org/pages/tc2_motion_control/) auf und vereinfachen somit die Handhabung der verschiedenen Achsen innerhalb des SPS-Programms erheblich.

Mit SoftMotion ist es unter CODESYS auch möglich, komplette CNC-Steuerungen zu programmieren (siehe auch <http://de.codesys.com/produkte/codesys-motion-cnc/softmotion.html>)

2.3 Aufgaben für den Anwender

Unter Benutzung der CODESYS-SoftMotion-Bibliotheken hat der Anwender beim Design einer Automatisierungsanlage mit feldbusgesteuerten Antrieben folgende Aufgaben zu erledigen:

- Einhängen des Feldbusses in den Gerätebaum der SPS
- Parametrieren des Feldbusses
- Einhängen der Antriebe (Achsen) in den Gerätebaum unterhalb des Feldbusses
- Parametrierung der Achsen (Art, Skalierung, Grenzwertparameter usw. – jeweils abhängig vom verwendeten Antrieb)
- Taskkonfiguration
- Programmierung der Antriebe auf Basis der SoftMotion-Funktionsbausteine im CODESYS

Es wurde ein einfaches Beispiel (siehe unten) realisiert, das die prinzipielle Herangehensweise an komplexe Antriebssysteme erklären soll. Dazu wurden zwei Servomotoren unterschiedlicher Bauart und von unterschiedlichen Herstellern über den EtherCat-Feldbus an eine SPS mit einem PLCcore-E660 CODESYS angeschlossen und mit SoftMotion betrieben.

3 Realisierung

3.1 Hardware

Zwei Antriebe sind über EtherCat an die SPS SYS TEC PLCcore-E660 CODESYS angeschlossen (siehe Abbildung 1). Der EtherCAT-Feldbus hat hier eine Linien-Topologie. Es wurden ein Wechselstrom-Servomotor mit Resolver und ein Gleichstrom-Servomotor (elektronische Kommutierung) mit Encoder an ihre zugehörigen Servocontroller angeschlossen. Letztere sind über EtherCAT mit der SPS verbunden.

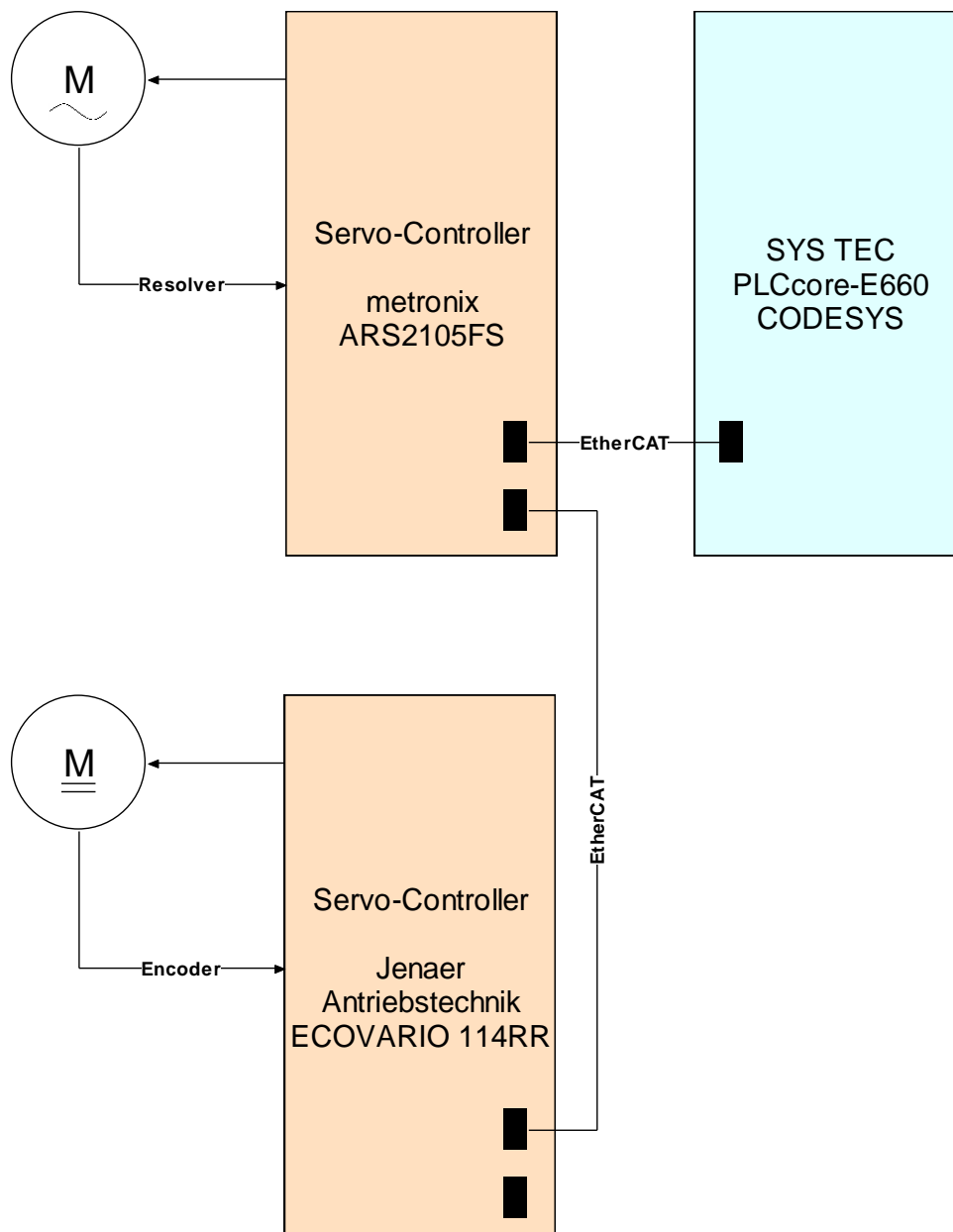


Abbildung 1: Hardwarestruktur des Realisierungsbeispiels

3.2 Software-Projekt

Das Beispielprojekt ist realisiert in CODESYS V3. Die verwendeten Software-Komponenten und -Werkzeuge sind:

- CODESYS V3.5 SP4 Patch 2 als Entwicklungsumgebung (neuere Versionen als kostenloser Download verfügbar bei <http://de.codesys.com/>)
- Laufzeitsystem für Linux auf x86-Prozessoren "CODESYS Control x86 SYS TEC electronic target" mit statisch gelinkter SoftMotion-Bibliothek – eine Demo-Version mit 2 Stunden Laufzeitlimit für das Laufzeitsystem und 30 Minuten für die EtherCAT-SoftMotion-Applikation
- Device-Deskriptor für die SPS "CODESYS Control x86 SYS TEC electronic target" – ins Geräte-Repository der Entwicklungsumgebung zu installieren
- Betriebssystem Linux auf dem SYS TEC PLCcore-E660 CODESYS

Wie in Abbildung 2 gezeigt wird, sind für den Projektaufbau folgende Regeln zu beachten:

- Als SPS ist "CODESYS Control x86 SYS TEC electronic target" auszuwählen.
- Der EtherCAT-Master ist im Gerätebaum unterhalb der SPS "CODESYS Control x86 SYS TEC electronic target" einzuhängen – darunter die Antriebe von metronix und Jenaer Antriebstechnik (unterer Teil der Abbildung, rot gekennzeichnet).
- Die Taskkonfiguration hat so zu erfolgen, dass die Tasks für beide Antriebessteuerungen unterhalb des "EtherCAT_Master" einzuhängen sind, wobei "EtherCAT_Master.EtherCAT_Task" bei Aufbau des Projekts automatisch eingehängt wird (mittlerer Teil der Abbildung, ebenfalls rot gekennzeichnet).
- Die den Tasks für den Antrieb von Jenaer Antriebstechnik und für den Antrieb von metronix zugeordneten Programme EtherCAT_JAT_PRG bzw. EtherCAT_PRG sind im Gerätebaum unter "Application" einzuhängen (oberer Teil der Abbildung, grün gekennzeichnet). Innerhalb dieser beiden Programme in ST-Code erfolgt die anwenderspezifische Programmierung des jeweiligen Antriebs. In dem beim Projektaufbau automatisch eingehängten, zur "MainTask" gehörigen Programm PLC_PRG (ebenfalls im ST-Code) kann der Anwender z. B. zusätzliche Steuersequenzen allgemeingültiger Natur bzw. zur Synchronisation der beiden Antriebe platzieren – im gegebenen Beispiel enthält dieses Programm keinen Code.

Wie sehr die Benutzung der SoftMotion-Funktionsbausteine die Programmierung der Antriebe vereinfacht, ist aus den folgenden ST-Code-Fragmenten ersichtlich.

Code-Fragment für das Auslesen der aktuellen Position einer Achse

```
(* Read actual position of the drive*)
ReadActPosition(
    Enable := TRUE,
    Axis   := SM_Drive_ETC_JAT_ECOVARIO
);
fReadDone := ReadActPosition.Valid;
fReadErr  := ReadActPosition.Error;
wReadErrId := ReadActPosition.ErrorID;
lActPos   := ReadActPosition.Position;
```

Code-Fragment für einen Bewegungsablauf

```
(* Perform drive movement profile *)
...
...
CASE iStatusMove OF
  0:
    Power(Enable:=TRUE, bRegulatorOn:=TRUE, bDriveStart:=TRUE,
          Axis:=SM_Drive_ETC_JAT_ECOVARIO);
    fPowerStatus := Power.Status;
    fPowerErr    := Power.Error;
    iPowerErrID  := Power.ErrorID;
    IF fPowerStatus
    THEN
      iStatusMove := iStatusMove + 1;
    END_IF;

  1:
    p := 100;
    MoveAbsolute(Execute:=TRUE, Position:= p, Velocity:=1000,
                Acceleration:=100,
                Deceleration:=100, Axis:=SM_Drive_ETC_JAT_ECOVARIO);
    fMoveDone := MoveAbsolute.Done;
    fMoveErr  := MoveAbsolute.Error;
    iMoveErrId := MoveAbsolute.ErrorID;
    IF fMoveDone
    THEN
      MoveAbsolute(Execute:=FALSE, Axis:=SM_Drive_ETC_JAT_ECOVARIO);
      iStatusMove := iStatusMove + 1;
    END_IF;

  2:
  ...
  ...
```

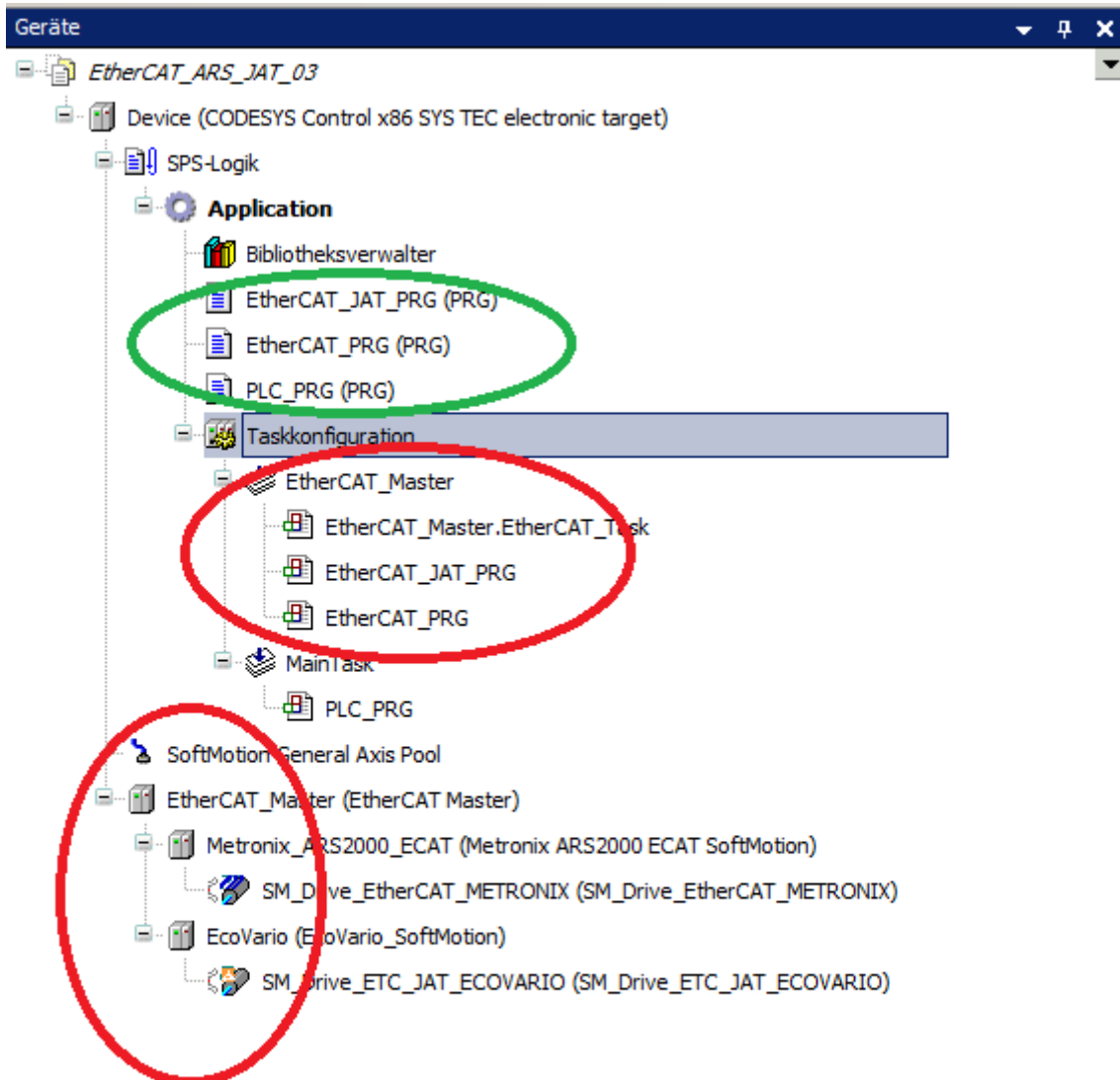


Abbildung 2: Gerätebaum für das Beispielprojekt

3.3 Fazit

Die Verwendung von SoftMotion-Funktionsbausteinen reduziert den Programmieraufwand für den Anwender erheblich. Prinzipiell ist es möglich statt EtherCAT auch andere Feldbus-Kopplungen zwischen SPS und Antrieben zu verwenden (z. B. CANopen). Der Programmteil der Bewegungssteuerung im Anwenderprogramm bleibt prinzipiell derselbe – lediglich die verwendeten Treiber und Bibliotheken sind andere, und die Bezeichnungen der Antriebe ändern sich entsprechend.