

SYSTEMHANDBUCH-ZUSATZ
Version 0.3

E/A Modul 515 - 4 D/A

**Erweiterungsmodul
mit 4 analogen Ausgängen**

zum Einsatz in CANopen - Netzwerken

Hinweis:

Dies ist nur eine "vorläufige" Beschreibung, da das Programm für das EA515 mit 4 Analog-Ausgängen zur Zeit (April 99) noch weiterentwickelt und getestet wird. Änderungen sind vorbehalten.

Inhaltsverzeichnis

Inhalt

1. VERSIONS-HISTORIE	3
2. VORWORT	3
3. ANALOGE AUSGÄNGE BEIM E/A-MODUL „4 D/A“	4
3.1 ELEKTRISCHE EIGENSCHAFTEN DER ANALOGEN AUSGÄNGE.....	4
4. ZUGRIFF AUF DIE ANALOGEN AUSGÄNGE	5
4.1 ANSTEUERUNG DER ANALOGEN AUSGÄNGE PER SDO.....	5
4.2 ANSTEUERUNG DER ANALOGEN AUSGÄNGE PER PDO.....	5
4.3 ANSTEUERUNG DER ANALOGEN AUSGÄNGE IM FEHLERFALL UND NACH POWER-UP.....	7
5. PROGRAMMIERUNG "WICHTIGER" CODESTELLEN MIT DEM DIP-SCHALTER	8
5.1 BEDIENUNG.....	8
5.2 BESONDERHEITEN EINIGER CODESTELLEN BEI PROGRAMMIERUNG PER DIP-SCHALTER.....	9
5.2.1 <i>Options-Flags (C19)</i>	9
5.2.2 <i>abgekürzte Sync-Konfiguration (C25)</i>	9
5.2.3 <i>Kommunikationszyklus (C26)</i>	9
5.3 BEISPIELE ZUR MANIPULATION VON CODESTELLEN PER DIP-SCHALTER.....	10
5.3.1 <i>Einstellen der PDO-Übertragungsart auf "Synchron-Empfang"</i>	10
5.3.2 <i>Einstellen der PDO-Übertragungsart auf "SYNC-Senden"</i>	11
6. ANHANG A: ZUSÄTZLICHE CODESTELLEN DES MODULS „4 D/A“	12
7. ANHANG B: OBJEKTE NACH DS401 ZUM ANSTEUERN DER ANALOGEN AUSGÄNGE	13

Zusatzinformation „4 analoge Ausgänge“

1. Versions-Historie

Versionsnummer	Datum	Autor	Bemerkungen, Änderungsgrund
V0.1	21.04.1999	W.Büscher	Ersterstellung
V0.2	08.07.1999	W.Büscher	Beschreibung DIP-Schalter-Konfiguration aus „EA515-4AIN“ übernommen
V0.3	03.Nov.1999	W.Büscher	Kleine Erweiterungen während der Übersetzung ins Englische
V0.4	2008-09-10	W. Büscher	Umstieg auf OpenOffice; ungültigen Verteiler ("Mart2") entfernt; Dokumentname nun offiziell mit Artikelnummer: art85024_DA4_Zusatz (.odt/.pdf) . Export aus OO als <i>tagged PDF</i> mit Hyperlinks .

2. Vorwort

Dieser Zusatz zum Systemhandbuch „EA515“ enthält **ergänzende** Informationen zu den Analogeingängen des E/A-Moduls EA515 „4 D/A“.

Für allgemeine Informationen (auch zu den analogen Ein- und Ausgängen) finden Sie im allgemeinen Systemhandbuch für die EA515 - Familie (Artikelnummer 85021) .

Zusatzinformation „4 analoge Ausgänge“

3. Analoge Ausgänge beim E/A-Modul „4 D/A“

3.1 elektrische Eigenschaften der analogen Ausgänge

Die vier gleichwertigen analogen Ausgänge dieses Moduls sind bipolare Spannungsausgänge mit den folgenden Eigenschaften:

- Spannungsbereich -10 V bis $+10\text{ V}$
- Auflösung 12 Bit **plus** Vorzeichen
- minimaler Lastwiderstand $2\text{k}\Omega$ (?)
- per Hardware realisierter Tiefpaß zur Rauschunterdrückung, Zeitkonstante ca. 0.5ms
- Kalibrierung erfolgt „rein hardwaremäßig“ durch Präzisionswiderstände
- nach Einschalten der Versorgungsspannung liefern alle Ausgänge zunächst 0V

Zusatzinformation „4 analoge Ausgänge“

4. Zugriff auf die analogen Ausgänge

Die analogen Ausgänge werden vom Prozessor der E/A-Baugruppe über mehrere Digital/Analog-Wandler seriell angesteuert. Die Ausgänge können z.Z. alle 5 Millisekunden aktualisiert werden (abhängig von der Art des Zugriffs per CAN-Bus). Prinzipbedingt vergehen zwischen Ansteuerung des ersten und letzten analogen Ausgangs innerhalb eines Zyklusses etwa 0.5 Millisekunden, auch wenn per CAN-Bus alle vier Ausgänge „gleichzeitig“ angesteuert werden sollen.

4.1 Ansteuerung der analogen Ausgänge per SDO

Die Ansteuerung kann per SDO (Service-Daten-Objekt) oder PDO (Prozeß-Daten-Objekt) erfolgen. Die dazu **nötigen** CANopen-Dictionary-Objekte finden Sie im Anhang B. Das wichtigste Objekt zum Ansteuern der analogen Ausgänge ist Objekt 0x6411 nach CiA-Draft Specification 401. Mit diesem Objekt können Sie per SDO-CAN-Telegramm je einen analogen Ausgang ansteuern, indem Sie die gewünschten Ausgangspegel als „16-Bit-Integer-Wert“ in das Objekt 0x6411 schreiben. Dabei entspricht –10 Volt dem Wert „-32768“ und +10 Volt dem Wert „+32767“¹.

Normalerweise werden elektrische Ausgänge allerdings nicht per SDO angesteuert, weil diese Zugriffsmethode recht langsam ist und nicht synchron erfolgt. Außerdem kann pro SDO-Schreibzugriff immer nur **ein** Ausgang gesetzt werden.

4.2 Ansteuerung der analogen Ausgänge per PDO

Falls Sie alle vier analogen Ausgänge mit **einem** CAN-Telegramm aktualisieren wollen, müssen Sie die Analog-Daten per PDO übertragen. Dazu können Sie z.B. die Subindizes 1...4 aus Objekt 0x6411 in den Prozeßdatenkanal einblenden („mappen“).

Die prinzipielle Funktion des „PDO-Mappings“ ist im allgemeinen Systemhandbuch (Kapitel „CANopen“, Unterkapitel „PDO-Mapping“) beschrieben.

Analogwerte werden beim EA515 wie alle anderen Prozessdaten in den ersten (und einzigen) PDO gemappt. Als „Werkseinstellung“ sind alle D/A-Wandler-Kanäle als 16-Bit-Worte in den PDO gemappt. Der verwendete CAN-Identifizier ist hier nicht vom Inhalt des PDO's abhängig² !

¹ generell ist „+32767“ immer die Codierung des „maximal möglichen Ausgangspegels, egal ob dies nun +10 Volt oder +20 mA entspricht und ob der Wandler eine Genauigkeit von 10, 13 oder 16 Bit hat. Um diese Tatsache zu verdeutlichen, steht in der Beschreibung zu Objekt 0x6411 in DS401 der Zusatz „unconverted“.

Dies bedeutet nicht, daß die Zahlenwerte in Objekt 0x6411 gänzlich unkonvertiert vorliegen. Auch bei „unconverted“ erfolgt grundsätzlich eine Konvertierung vom „linksbündigen“ CANopen-Format ins „rechtsbündige“ Wandlerformat (hier: -2047... +2047) !

² In DS401 wird empfohlen, Analogdaten im „2nd PDO“ zu übertragen. Das EA515 hat allerdings keinen „2nd“ PDO, sondern nur „einen“ PDO. Sie können allerdings den CAN-Identifizier des im EA515 verwendeten PDOs flexibel einstellen (Codestelle C50 für PDO-RX, C51 für PDO-TX). Näheres zu diesem Thema finden

Zusatzinformation „4 analoge Ausgänge“

Um das PDO-Mapping selbst durchzuführen, setzen Sie (in der folgenden Reihenfolge):

Objekt 1600h, Subindex 0:

auf 04h (-> vier Objekte werden in den PDO gemappt)

Objekt 1600h, Subindex 1:

auf 64110110h (Objekt 6411, Subindex 1 mit 16 Bit in PDO einblenden)

Objekt 1600h, Subindex 2:

auf 64110210h (Objekt 6411, Subindex 2 mit 16 Bit in PDO einblenden)

Objekt 1600h, Subindex 3:

auf 64110310h (Objekt 6411, Subindex 3 mit 16 Bit in PDO einblenden)

Objekt 1600h, Subindex 4:

auf 64110410h (Objekt 6411, Subindex 4 mit 16 Bit in PDO einblenden)

Speichern Sie diese Konfiguration –wenn nötig- dauerhaft im E/A-Modul ab³.

Danach erwartet das E/A-Modul in den ersten beiden Bytes des Prozeßdatenkanals einen 16-Bit-Wert für den ersten analogen Ausgang (low byte first), in den nächsten beiden Bytes den Wert für den zweiten analogen Ausgang und so weiter⁴.

Im Gegensatz zu der Empfehlung in CiA DS401 verwendet das EA515 zum Übertragen von analogen Prozeßdaten den gleichen PDO-Kanal wie zum Übertragen von digitalen Prozeßdaten. Das EA515 hat **immer** nur einen Transmit- und einen Receive-PDO. Den in DS401 erwähnten „**2nd PDO**“ für Analogdaten **gibt es beim EA515 nicht**. Sollte es Ihre Anwendung erfordern, daß die analogen Prozeßdaten mit dem CAN-Identifizier des „2nd PDOs“ übertragen werden, so können Sie die Identifizier-Basis des PDOs mit Codestelle 50 auf den benötigten Wert setzen (siehe Systemhandbuch und CiA DS401).

Sie im „allgemeinen Systemhandbuch“ der EA515-Familie.

³ Das dauerhafte Abspeichern geänderter Parameter erfolgt mit Objekt 0x1010 im CANopen „Communication Profile“.

⁴ Beachten Sie, daß die PDO-Übertragung nur im Modul-Zustand „Operational“ funktioniert. Per SDO können Sie die analogen Ausgänge bereits im Zustand „Pre-Operational“ setzen.

Zusatzinformation „4 analoge Ausgänge“

4.3 Ansteuerung der analogen Ausgänge im Fehlerfall und nach Power-Up

Es besteht die Möglichkeit, die analogen Ausgänge im Fehlerfall auf einen vorprogrammierten Wert zu setzen. Zur Konfiguration dienen die in Anhang B beschriebenen CANopen-Dictionary-Objekte 0x6443 und 0x6444 (nach CiA DS401). Der einzige (derzeit) vom Modul erkannte Fehlerfall ist der Ausfall der CAN-Verbindung zwischen EA-Modul und übergeordneter Steuerung. Um den Ausfall der CAN-Verbindung überwachen zu können, muß das leider recht komplexe „Extended NMT Bootup“ mit „Nodeguarding-Aktivierung“ nach CANopen-Standard verwendet werden. Eine kurze Beschreibung des dazu nötigen CAN-Protokolls finden sie im allgemeinen Systemhandbuch im Kapitel „CANopen“.

Mit Objekt 0x6443 definieren Sie den „Output Fault MODE“, d.h. das grundsätzliche Verhalten der analogen Ausgänge im Fehlerfall (für jeden Kanal individuell einstellbar). Derzeit sind die folgenden „Fault Modes“ in der Firmware implementiert:

Fault Mode (Objekt 0x6443)	Bezeichnung nach DS401	Bedeutung, Hinweise
0	actual value rest	aktuellen Wert beibehalten
1	revert to default value, unconverted	den durch Objekt 0x6444 definierten Wert einstellen
(2)	revert to default value, converted	(wird derzeit [20.04.99] wie Fault Mode 1 behandelt)

Um das Fehlerverhalten von Kanal 1 zu definieren, verwenden Sie Subindex 1 in Objekt 0x6443, für Kanal 2 ist Subindex 2 zuständig u.s.w.

Mit Objekt 0x6444 können Sie die Werte festlegen, mit denen die analogen Ausgänge im Fehlerfall angesteuert werden sollen. Das Format der Einträge in Objekt 0x6444 entspricht dem Format von Objekt 0x6411.

Als kleine Erweiterung (die der Programmierer in dieser Form in DS401 nicht finden konnte) dient Objekt 0x6444 auch als **Default-Wert nach Einschalten der Versorgungsspannung** des E/A-Moduls. In der „Werkseinstellung“ liefern alle Ausgänge nach dem Einschalten solange den Wert Null, bis sie per SDO oder PDO angesteuert werden.

Mit Objekt 0x6444 ist es dagegen möglich, die analogen Ausgänge (fast⁵) sofort nach dem Einschalten auf beliebige „Anfangswerte“ zu setzen. Programmieren Sie die gewünschten Anfangswerte der analogen Ausgänge 1..4 in Objekt 0x6444.1..4 und speichern Sie die Konfiguration dauerhaft ab.

⁵ Eine gewisse Verzögerung im Millisekundenbereich ist nicht zu vermeiden, weil die Konfiguration des Moduls nach dem Einschalten erst aus einem seriell angesteuerten EEPROM gelesen wird. In den ersten Millisekunden nach dem Einschalten liefern die analogen Ausgänge daher immer den Wert Null.

Zusatzinformation „4 analoge Ausgänge“

5. Programmierung "wichtiger" Codestellen mit dem DIP-Schalter

5.1 Bedienung

Um die wichtigsten Parameter der E/A-Baugruppe auch ohne SDO-Zugriff ändern zu können, wurde eine Prozedur zum Umprogrammieren einiger Codestellen per DIP-Schalter (Grundplatine) implementiert. Der Verfahrensablauf für den Programmiermodus wurde dafür etwas abgewandelt und entspricht beim EA515 mit 4*Analog-Input daher **nicht mehr der Beschreibung im Systemhandbuch (V0.xx)**.

Wie gehabt wird der Programmiermodus durch kurzes Schließen von S12 aktiviert. Im weiteren Verlauf hat S12 im Programmiermodus etwa die Funktion einer "Enter-Taste", mit der die per DIP-Schalter eingestellten Daten in eine Codestelle geschrieben werden. (Hinweis: bei der vorliegenden Hardware ist 1="ON"=unten und 0="OFF"=oben.) Mit Schalter 9,10,11 wird die zu programmierende Codestelle ausgewählt. Mit Schalter 1-8 werden die (maximal) 8 Datenbits eingestellt, die in die Codestelle programmiert werden sollen.

Da sich mit den Schaltern 9,10,11 acht unterschiedliche Kombinationen einstellen lassen, können auf diese Weise maximal 8 verschiedene Codestellen programmiert werden. Der mit S9,10,11 eingestellte Code entspricht einer Codenummer nach folgender Tabelle:

Schalter 11	Schalter 10	Schalter 9	Codestelle	Bedeutung
0	0	0	C10	Portrichtungen
0	0	1	C19	Options-Flags
0	1	0	C25	abgekürzte Sync-Konfiguration
0	1	1	C26	Kommunikations-Zyklus
1	0	0	???	bei dieser Variante ungenutzt
1	0	1	???	bei dieser Variante ungenutzt
1	1	0	C2048	Default-Einstellungen (für Hersteller)
1	1	1	reserviert	

Die Nummer der Codestelle wird im Programmiermodus mit der Mode-LED (grün, Hunderter-Stelle), Modul-Fehler-LED (rot, Zehner-Stelle) und Ausgangs-Fehler-LED (rot, Einer-Stelle) angezeigt. Dazu blinkt jede LED jeweils entsprechend der Dezimalstelle. Für die Ziffer "0" blinkt die entsprechende LED nicht, für die Ziffer "9" entsprechend neun mal. Der gesamte "Blink-Zyklus" wiederholt sich alle 7 Sekunden, so daß die aktivierte Codestelle durch "Mitzählen" der Blinkimpulse erkennbar ist.

Zur dauerhaften Übernahme der eingestellten Daten wird der Programmierschalter (S12) kurz nach "ON" und dann wieder nach "OFF" geschaltet. Beim Wechsel von "ON" nach "OFF" leuchten alle grünen und roten LEDs für ca. 0.5 Sekunden gemeinsam auf, um eine erfolgreiche Programmierung anzuzeigen.

Der Programmiermodus kann nur durch Abschalten der Versorgungsspannung verlassen werden. Vor dem Wiedereinschalten der Spannung sollte wieder der passende Modul-ID und die korrekte Baudrate eingestellt werden.

Zusatzinformation „4 analoge Ausgänge“

5.2 Besonderheiten einiger Codestellen bei Programmierung per DIP-Schalter

5.2.1 Options-Flags (C19)

In der Beschreibung dieser Codestelle in Anhang A finden Sie ein Bit zur Umschaltung zwischen „Lenze“- und „DS401“-Modus. Das Setzen/Löschen dieses Bits hat allerdings erst einen Effekt, wenn Sie anschließend die „Default-Einstellungen“ für das Kommunikationsprofil laden. Ein spezieller „Lenze-Modus“ ist für das EA515-4Aout nicht vorgesehen (er würde sich nur durch den verwendeten PDO-CAN-Identifizier unterscheiden, und dessen Einstellung erfolgt besser per SDO-Zugriff auf das standardisierte CANopen-Object-Dictionary)

5.2.2 abgekürzte Sync-Konfiguration (C25)

Der Inhalt dieser Codestelle ist identisch mit dem höchstwertigen Byte aus Objekt 1005h nach CiA Draft Standard 301. Objekt 1005h wird dort (etwas irreführend) als "COB-ID Sync Message" bezeichnet. In der Tat wird aber mit dem höchstwertigen Byte dieses CANopen-Objektes die Verarbeitung bzw. Erzeugung des SYNC-Telegramms definiert. Das dort beschriebene Bit 31 ("1: Device consumes SYNC message) entspricht S8 des DIP-Schalters, wenn C25 programmiert wird.

Bit 30 aus DS301 ("1: Device generates SYNC message") entspricht S7 des DIP-Schalters.

Das Beschreiben dieser Codestelle wirkt sich auch auf den Inhalt von Objekt 1005h aus !

Beispiele zur Parametrierung des Synchron-Telegramms finden sich in Kapitel 5.3.

5.2.3 Kommunikationszyklus (C26)

Der Inhalt dieser Codestelle entspricht dem in CiA Draft Standard 301 beschriebenen Objekt 1006h (Communication Cycle Period), allerdings die hier enthaltene Parameter eine Zeit in **Millisekunden** (nicht Mikrosekunden). Per DIP-Schalter können maximal 255 Millisekunden programmiert werden, per SDO maximal 32767.

Wenn ein Modul als SYNC-Sender konfiguriert ist (sh. C25), dann ist diese Zeit das Intervall zwischen zwei SYNC-Sendungen.

Das Beschreiben dieser Codestelle wirkt sich auch auf den Inhalt von Objekt 1006h aus !

Zusatzinformation „4 analoge Ausgänge“

5.3 Beispiele zur Manipulation von Codestellen per DIP-Schalter

5.3.1 Einstellen der PDO-Übertragungsart auf "Synchron-Empfang"

- 1) Wechsel in den Programmiermodus durch Schließen von S12.
Alle roten und grünen LEDs blinken "kurz-lang-lang-kurz" um den Programmiermodus anzuzeigen.
- 2) S12 wieder öffnen.
Die roten und grünen LEDs zeigen ab jetzt die per S9..S11 gewählte Codestelle an.
- 3) S9,10,11 für die "abgekürzte SYNC-Konfiguration" per C25 einstellen:
S9 off, S10 on, S11 off
- 4) Mit S1..S8 den Wert 0x80 (hex) einstellen:
S1..S7 off, S8 on
- 5) Programmieren:
S12 kurz "on" und wieder "off" -> rote und grüne LEDs gehen kurz an.

Anmerkung: bei dieser Einstellung wird auch automatisch der "PDO Transmission Type" (CANopen-Objekt 1800h) auf "Synchronbetrieb" geschaltet, falls dies noch nicht der Fall ist. Bei der Programmierung per SDO entfällt diese Automatik⁶.

Um den Programmiermodus zu verlassen, schalten Sie erst die Modulversorgung ab. Vergessen Sie nicht, vor dem Wiedereinschalten die DIP-Schalter wieder passend für den „Normalbetrieb“ einzustellen (Modul-ID und CAN-Baudrate).

⁶ Wie Sie sehen, ist das Umprogrammieren von Codestellen per DIP-Schalter sehr arbeits- und zeitaufwendig. Die Programmierung per SDO (CAN-Bus) ist der Programmierung per DIP-Schalter daher vorzuziehen. Sie benötigen dazu allerdings ein CANopen-fähiges „Konfigurationstool“, das z.B. aus einem Laptop mit CAN-Schnittstelle und geeigneter Software bestehen kann.

Zusatzinformation „4 analoge Ausgänge“

5.3.2 Einstellen der PDO-Übertragungsart auf "SYNC-Senden"

Für spezielle Anwendungen kann das Modul auch als SYNC-Telegramm-Sender arbeiten. In diesem Fall wird auch der eigene Sende-PDO "synchronisiert".

- 1) Wechsel in den Programmiermodus durch Schließen von S12.
Alle roten und grünen LEDs blinken "kurz-lang-lang-kurz" um den Programmiermodus anzuzeigen.
- 2) S12 wieder öffnen.
Die roten und grünen LEDs zeigen ab jetzt die per S9..S11 gewählte Codestelle an.
- 3) S9,10,11 für die "abgekürzte SYNC-Konfiguration" per C25 einstellen:
S9 off, S10 on, S11 off
- 4) Mit S1..S8 den Wert 0x40 (hex) einstellen:
S1..S6 off, S7 on, S8 off.
- 5) Programmieren:
S12 kurz "on" und wieder "off" -> rote und grüne LEDs gehen kurz an.

Anmerkung: bei dieser Einstellung wird auch automatisch der "PDO Transmission Type" (CANopen-Objekt 1800h) auf "Synchronbetrieb" geschaltet, falls dies noch nicht der Fall ist. Ferner wird der "Communication Cycle" (CANopen-Objekt 1006h) auf einen gültigen Wert gesetzt, wenn dies noch nicht der Fall ist (mindestens 5 Millisekunden).
Bei der Programmierung per SDO entfällt diese Automatik.

Um einen definierten Kommunikationszyklus einstellen, sollten Sie auf jeden Fall auch den Inhalt von C26 per DIP-Schalter auf den benötigten Wert setzen (siehe 5.2.3). Als Beispiel hier noch die Programmierung eines 10ms-Zyklus', wenn sich das Modul bereits im Programmiermodus befindet (d.h. nach Durchführen der Schritte 1 bis 5 aus dem oben beschriebenen Programmierung von C25):

- 6) S9,10,11 für die "abgekürzte Kommunikationszyklus-Einstellung" per C26 einstellen:
S9 on, S10 on, S11 off (nach Tabelle in Kapitel 5.1) .
- 7) Mit S1..S8 den Wert 10 Millisekunden (dezimal) einstellen:
10 dezimal = 00001010 binär, daher (LSB=Schalter 1, MSB=Schalter 8) :
S1 off, S2 on, S3 off, S4 on, S5...S8 off.
- 8) Programmieren („Programmierimpuls“):
S12 kurz "on" und wieder "off" -> rote und grüne LEDs gehen kurz an.

Um den Programmiermodus zu verlassen, schalten Sie erst die Modulversorgung ab. Vergessen Sie nicht, vor dem Wiedereinschalten die DIP-Schalter wieder passend für den „Normalbetrieb“ einzustellen (Modul-ID und CAN-Baudrate).

Anhang A: Zusätzliche Codestellen "4 D/A"

6.Anhang A: zusätzliche Codestellen des Moduls „4 D/A“

Code-nr.	Inhalt	Datentyp, [Einheit]	Zugriff r = read w= write	Einstell- / Anzeige-möglichkeiten	Werks-einstellung	Bemerkungen, Verweise
Codestellen für die elementare Modul-Konfiguration (zusätzlich zu den im Systemhandbuch beschriebenen Codestellen)						
18	Hardware-Modell	8 Bit	r ..	0x40 = „D/A4 installiert“ 0x00 = „keine Erweiterung“	0x40	n.f. Hersteller
19	Modul-Optionen	8 Bit	r ..	0x00 = „Lenze-Modus“ 0x01 = „DS401-Modus“	je nach Bestellung , i.A. 0x01	n.f. Hersteller
Codestellen für EA515 mit Erweiterungsplatinen für 4 analoge Ausgänge (nur verfügbar wenn Hardware-Erweiterung installiert ist) Zum Ansteuern der analogen Ausgänge sollten die im CANopen DS401 spezifizierten „Objekte“ verwendet werden !						
640.0	Anzahl "DA4"-Register	8 Bit	r	0 (nicht installiert) .. <Anzahl Register>	4	
640.1	aktueller Wert von Kanal 1	16 Bit	rw	-32767 ... +32767	0, wird von Obj. 0x6444 überschrieben	nur für Diagnose- Zwecke,
640.2	aktueller Wert von Kanal 2	16 Bit	rw	-32767 ... +32767	„	
640.3	aktueller Wert von Kanal 3	16 Bit	rw	-32767 ... +32767	„	
640.4	aktueller Wert von Kanal 4	16 Bit	rw	-32767 ... +32767	„	
640.x	eventuelle Erweiterungen		?			

Anhang B: Objekte nach DS401 bei "4 D/A"

7.Anhang B: Objekte nach DS401 zum Ansteuern der analogen Ausgänge

Index (hex)	Bezeichnung nach DS401, Anmerkungen und Hinweise	Subindizes	Typ	Zugriff
6411	<p>„Writes value of the output channel (not converted)“ Schreibzugriffe auf dieses Objekt steuern den Kanal (entsprechend Subindex) an. Subindex 0 ist „read-only“ und enthält die Anzahl Kanäle, hier: 4. Die Subindizes 1..4 dieses Objektes können (als 16-Bit-Parameter) in den PDO-Kanal eingeblendet werden (siehe „PDO-Mapping“ im Systemhandbuch). Skalierung: -32767 = -10 V (Min.) 0 = 0 V +32767 = +10 V (Max.)</p>	0 1 ... 4	Unsigned 8 Signed 16	ro rw
6443	<p>„Output Fault Mode“ Dieses Objekt definiert, wie sich die analogen Ausgänge im Fehlerfall verhalten sollen. „Werkseinstellung“ ist für Subindex 1..4 der Wert Null.</p>	0 1 ... 4	Unsigned 8 Unsigned 16	ro rw
6444	<p>„Default Output Fault value (not converted)“ Dieses Objekt definiert, welchen Pegel die analogen Ausgänge im Fehlerfall annehmen sollen, wenn Objekt 6443 entsprechende Aktionen auslöst. Beim EA515 definieren sie darüberhinaus den Zustand der analogen Ausgänge nach dem Einschalten (vor CAN-Bus-Aktivität). „Werkseinstellung“ ist für Subindex 1..4 der Wert Null.</p>	0 1 ... 4	Unsigned 8 Signed 16	ro rw