

EASY204-DP PROFIBUS-DP-Slave-Anschaltung



Powering Business Worldwide

Alle Marken- und Produktnamen sind Warenzeichen oder eingetragene Warenzeichen der jeweiligen Titelhalter.

Störfallservice

Bitte rufen Sie Ihre lokale Vertretung an:

<http://www.eaton.com/moeller/aftersales>

oder

Hotline After Sales Service:

+49 (0) 180 5 223822 (de, en)

AfterSalesEGBonn@eaton.com

Originalbetriebsanleitung

Die deutsche Ausführung dieses Dokuments ist die Originalbetriebsanleitung.

Übersetzung der Originalbetriebsanleitung

Alle nicht deutschen Sprachausgaben dieses Dokuments sind Übersetzungen der Originalbetriebsanleitung.

1. Auflage 2001, Redaktionsdatum 04/01
 2. Auflage 2002, Redaktionsdatum 08/02
 3. Auflage 2003, Redaktionsdatum 05/03
 4. Auflage 2004, Redaktionsdatum 07/04
 5. Auflage 2005, Redaktionsdatum 03/05
 6. Auflage 2008, Redaktionsdatum 02/08
 7. Auflage 2010, Redaktionsdatum 09/10
- siehe Änderungsprotokoll im Kapitel „Zu diesem Handbuch“
© 2001 by Eaton Industries GmbH, 53105 Bonn

Autor: Peter Thiessmeier, Ronny Happ, Peter Roersch

Redaktion: Thomas Kracht, Barbara Petrick

Alle Rechte, auch die der Übersetzung, vorbehalten.

Kein Teil dieses Handbuches darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder einem anderen Verfahren) ohne schriftliche Zustimmung der Firma Eaton Industries GmbH, Bonn, reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

Änderungen vorbehalten.



Gefahr! Gefährliche elektrische Spannung!

Vor Beginn der Installationsarbeiten

- Gerät spannungsfrei schalten
- Gegen Wiedereinschalten sichern
- Spannungsfreiheit feststellen
- Erden und kurzschließen
- Benachbarte, unter Spannung stehende Teile abdecken oder abschränken.
- Die für das Gerät angegebenen Montagehinweise (AWA) sind zu beachten.
- Nur entsprechend qualifiziertes Personal gemäß EN 50110-1/-2 (VDE 0105 Teil 100) darf Eingriffe an diesem Gerät/System vornehmen.
- Achten Sie bei Installationsarbeiten darauf, dass Sie sich statisch entladen, bevor Sie das Gerät berühren.
- Die Funktionserde (FE) muss an die Schutzerde (PE) oder den Potentialausgleich angeschlossen werden. Die Ausführung dieser Verbindung liegt in der Verantwortung des Errichters.
- Anschluss- und Signalleitungen sind so zu installieren, dass induktive und kapazitive Einstreuungen keine Beeinträchtigung der Automatisierungsfunktionen verursachen.
- Einrichtungen der Automatisierungstechnik und deren Bedienelemente sind so einzubauen, dass sie gegen unbeabsichtigte Betätigung geschützt sind.
- Damit ein Leitungs- oder Aderbruch auf der Signalseite nicht zu undefinierten Zuständen in der Automatisierungseinrichtung führen kann, sind bei der E/A-Kopplung hard- und softwareseitig entsprechende Sicherheitsvorkehrungen zu treffen.
- Bei 24-Volt-Versorgung ist auf eine sichere elektrische Trennung der Kleinspannung zu achten. Es dürfen nur Netzgeräte verwendet werden, die die Forderungen der IEC 60364-4-41 bzw. HD 384.4.41 S2 (VDE 0100 Teil 410) erfüllen.
- Schwankungen bzw. Abweichungen der Netzspannung vom Nennwert dürfen die in den technischen Daten angegebenen Toleranzgrenzen nicht überschreiten, andernfalls sind Funktionsausfälle und Gefahrenzustände nicht auszuschließen.
- NOT-AUS-Einrichtungen nach IEC/EN 60204-1 müssen in allen Betriebsarten der Automatisierungseinrichtung wirksam bleiben. Entriegeln der NOT-AUS-Einrichtungen darf keinen Wiederanlauf bewirken.
- Einbaugeräte für Gehäuse oder Schränke dürfen nur im eingebauten Zustand, Tischgeräte oder Portables nur bei geschlossenem Gehäuse betrieben und bedient werden.

- Es sind Vorkehrungen zu treffen, dass nach Spannungseinbrüchen und -ausfällen ein unterbrochenes Programm ordnungsgemäß wieder aufgenommen werden kann. Dabei dürfen auch kurzzeitig keine gefährlichen Betriebszustände auftreten. Ggf. ist NOT-AUS zu erzwingen.
- An Orten, an denen in der Automatisierungseinrichtung auftretende Fehler Personen- oder Sachschäden verursachen können, müssen externe Vorkehrungen getroffen werden, die auch im Fehler- oder Störfall einen sicheren Betriebszustand gewährleisten beziehungsweise erzwingen (z. B. durch unabhängige Grenzwertschalter, mechanische Verriegelungen usw.).

Inhalt

<hr/>	
	Zu diesem Handbuch 11
	Änderungsprotokoll 11
	Zielgruppe 11
	Weitere Handbücher zum Gerät 12
	Gerätebezeichnung 12
	Abkürzungen 13
	Lesekonventionen 14
<hr/>	
1	Zu EASY204-DP 15
	Systemübersicht 16
	Aufbau des Gerätes 17
	Gerätfunktionsbeschreibung 18
	– easy600/700/800, MFD-CP8... 18
	– easy800/MFD-CP8... 19
	Hardware- und Betriebssystem-Voraussetzungen 19
	Änderungshistorie 20
	Sachwidriger Einsatz 20
<hr/>	
2	Installation 21
	EASY204-DP an das Grundgerät anschließen 21
	Versorgungsspannung anschließen 22
	PROFIBUS-DP anschließen 23
	Anschlussbelegung PROFIBUS-DP 23
	Abschlusswiderstände 24
	EMV-gerechte Verdrahtung 24
	Potentialtrennungen 25
	Übertragungsraten – automatische 26
	Baudratenerkennung 26
	Maximale Entfernungen/Busleitungslängen 26

3	Gerät betreiben	29
	Erstes Einschalten	29
	PROFIBUS-DP Teilnehmeradresse einstellen	30
	– Adresse am Basisgerät mit Anzeige einstellen	30
	– Adresse mittels easySoft einstellen	32
	LED-Statusanzeigen	33
	– POW-LED, Funktion	33
	– BUS-LED, Funktion	33
	Zykluszeit EASY-Basisgerät	34

4	PROFIBUS-DP-Funktionen	35
	Konfiguration des DP-Masters, Klasse 1	35
	Slave-Module	36
	Diagnosedaten	38
	– Format der Diagnose-Informationen	38
	– Bedeutung der Diagnose-Informationen	41
	GSD-Datei	43
	Anwendermodule	44
	PROFIBUS-Zertifizierung	45

5	Ein-/Ausgänge, Betriebsart easy600/700/ 800/MFD	47
	Modul „Eingaenge 3 Byte“: Betriebsart, S1 - S8	48
	Modul „Eingaenge 1 Byte“: S1 – S8	50
	Modul „Ausgaenge 3 Byte“: Betriebsart, R9 - R16, R1 - R8	50
	Modul „Ausgaenge 1 Byte“: R1 – R8	54

6	Zusätzliche Ein-/Ausgangsdaten easy800/ MFD	57
----------	--	----

7 Steuerbefehle für easy600 (DPV0)	59
Datenaustauschverfahren	59
Datum und Uhrzeit, Sommer-/Winterzeit	61
Abbild lesen/schreiben	64
– Generelles zum Arbeiten mit Abbilddaten	64
– Übersicht	64
– Zustand Hilfsrelais, digitale Ausgänge und Textanzeige lesen	65
– Zustand P-Tasten und Bedientasten lesen	68
– Zustand Zeitrelais, Zählerrelais, Zeitschaltuhren und Analogwertvergleicher lesen	69
Funktionsbausteine lesen/schreiben	72
– Übersicht	72
– Analogwertvergleicher schreiben (Funktion, Vergleichswerte)	73
– Zählerrelais Istwert lesen	76
– Zählerrelais Sollwert schreiben	78
– Analog- und Digital-Eingänge lesen (I7, I8, I1 bis I16)	81
– Zeitrelais Istwert lesen (Zeitbereich, Istwert, Schaltfunktion)	84
– Zeitrelais Sollwert schreiben (Zeitbereich, Sollwert, Schaltfunktion)	87
– Zeitschaltuhr lesen (Kanal, ON-Zeit, OFF-Zeit)	94
– Zeitschaltuhr schreiben (Kanal, ON-Zeit, OFF-Zeit)	98
8 Steuerbefehle easy700 (DPV0)	101
Datenaustauschverfahren	101
Datum und Uhrzeit lesen/schreiben	104
Abbilddaten lesen/schreiben	108
– Übersicht	109
– Analogwertvergleicher/ Schwellwertvergleicher: A1 – A16	110
– Zähler: C1 – C16	111
– Textbausteine: D1 – D16	112
– Lokale Eingänge: I1 - I16	113
– Lokale Analog-Eingänge: IA1 - IA4	115

– Merker: M1 – M16/N1 – N16	117
– Merker: M1 – M16/N1 – N16	119
– Betriebsstundenzähler: O1 – O4	121
– Lokale P-Tasten: P1 - P4	122
– Lokale Ausgänge: Q1 - Q8	124
– Ein-/Ausgänge von easyLink: R1 – R16/S1 – S8	125
– Zeitglieder: T1 - T16	127
– Jahresschaltuhr: Y1 - Y8	128
– Masterreset: Z1 - Z3	129
– Wochenschaltuhr: 71 - 78	130
Funktionsbaustein-Daten lesen/schreiben	131
– Allgemeine Hinweise	131
– Übersicht	131
– Analogwertvergleicher/Schwellwertschalter: A1 - A16	132
– Zählerrelais: C1 - C16	135
– Betriebsstundenzähler: O1 - O4	138
– Zeitrelais: T1 - T16	140
– Jahresschaltuhr: Y1 - Y8	144
– Wochenschaltuhr: 71 - 78	147
Analyse – Fehlercodes über easyLink	150
<hr/>	
9 Steuerbefehle easy800/MFD (DPV0)	153
Datenaustauschverfahren	153
Versionsgeschichte	155
Datum und Uhrzeit lesen/schreiben	156
– Winter-/Sommerzeit, Zeitumstellung	157
Abbilddaten lesen/schreiben	160
– Übersicht	160
– Lokale Eingänge IW0 lesen	161
– Eingänge der Netzwerkteilnehmer IW1 bis IW8 lesen	163
– Lokale Analog-Eingänge IA1 bis IA4 lesen	164
– Lokale Diagnose ID1 bis ID16 lesen	166
– Lokale Ausgänge QW0/Ausgänge der Netzwerkteilnehmer QW1 bis QW8 lesen und schreiben	168
– Lokalen Analog-Ausgang QA1 lesen und schreiben	170

– Lokale P-Tasten lesen	171
– Eingänge RW.../Ausgänge SW... von easyLink lesen	173
– Receive-Data Netz RN1...RN32/ Send-Data Netz SN1...SN32 lesen	175
– Merker lesen und schreiben	177
Funktionsbaustein-Daten lesen/schreiben	181
– Allgemeine Hinweise	181
– Übersicht	182
– Analogwertvergleichler A01...A32	184
– Arithmetikbaustein AR01...AR32	186
– Block Compare BC01...BC32	188
– Block Transfer BT01...BT32	190
– Boolesche Verknüpfung BV01...BV32	192
– Zähler C01...C32	194
– Frequenzzähler CF01...CF04	196
– Schneller Zähler CH01...CH04	198
– Inkrementalzähler CI01...CI02	200
– Vergleichler CP01...CP32	202
– Textausgabe-Baustein D01...D32	204
– Datenbaustein DB01...DB32	207
– PID-Regler DC01...DC32	209
– Diagnose DG01...DG16	212
– Signalglättungsfilter FT01...FT32	214
– Empfang von Netzdaten GT01...GT32	216
– Wochenzeitschaltuhr HW01...HW32	218
– Jahreszeitschaltuhr HY01...HY32	221
– Bedingter Sprung JC01...JC32	224
– Wertskalierung LS01...LS32	226
– Masterreset MR01...MR32	228
– Datenmultiplexer MX01...MX32	230
– Zahlenwandler NC01...NC32	232
– Betriebsstundenzähler OT01...OT04	234
– Impulsausgabe PO01...PO02	236
– Senden von Netzdaten PT01...PT32	239
– Pulsweitenmodulation PW01...PW02	241
– Uhr synchronisieren SC01	243
– Serielle Ausgabe SP01...SP32	244
– Schieberegister SR01...SR32	246
– Sollzykluszeit ST01	248

– Zeitrelais T01...T32	250
– Tabellenfunktion TB01...TB32	253
– Wertbegrenzung VC01...VC32	255
<hr/>	
10 Prozessdaten easy600 (DPV1)	259
Objektübersicht	260
Zugriff auf Objekte	262
Identifikation lesen	263
– Objekt „Easy Identifikation“	263
Betriebsart lesen/schreiben	265
– Objekt „Easy Betriebsart“	265
Abbild lesen/schreiben	267
– Objekt „Easy 600 Eingänge“	267
– Objekt „Easy 600 Ausgänge und Merker“	269
– Objekt „Easy 600 Funktionsrelais“	272
– Objekt „Easy 600 Tasten“	275
– Objekt „Easy Link Eingangsdaten“	277
– Objekt „Easy Link Ausgangsdaten“	279
Funktionsbaustein-Daten lesen/schreiben	281
– Objekte „Easy 600 Parameter Zeitrelais T1 bis Zeitrelais T8“	281
– Objekte „Easy 600 Parameter Zähler C1 bis Zähler C8“	286
– Objekte „Easy 600 Parameter Analogwertvergleichler A1 bis A8“	289
– Objekte „Easy 600 Parameter Wochenschaltuhr 71 Kanal A bis Wochenschaltuhr 74 Kanal D“	291
Datum und Uhrzeit lesen/schreiben	294
– Objekt „Easy Uhr“	294
Sommer-/Winterzeit lesen/schreiben	297
– Objekte Easy 600 Sommer-/Winterzeit Einstellung	297

11 Prozessdaten easy700 (DPV1)	299
Objektübersicht	300
Zugriff auf Objekte	302
Identifikation lesen	303
– Objekt „Easy Identifikation“	303
Betriebsart lesen/schreiben	303
– Objekt „Easy Betriebsart“	303
Abbild lesen/schreiben	304
– Objekt „Easy 700/800 Eingänge“	304
– Objekte „Easy 700/800 Analogeingänge IA1 bis IA4“	306
– Objekt „Easy 700/800 Ausgänge“	308
– Objekt „Easy 700/800 Tasten“	310
– Objekt „Easy 700 Analogwertvergleicher“	312
– Objekt „Easy 700 Wochenschaltuhren“	314
– Objekt „Easy 700 Jahresschaltuhren“	316
– Objekt „Easy 700 Masterreset“	318
– Objekt „Easy 700 Textbausteine“	320
– Objekt „Easy 700 Zeitglieder“	322
– Objekt „Easy 700 Zähler“	324
– Objekt „Easy 700 Betriebsstundenzähler“	326
– Objekt „Easy 700 Aktualwert Merker M“	328
– Objekt „Easy 700 Aktualwert Merker N“	330
– Objekte „Easy 700 Neuer Wert Merker M1 bis M16“	332
– Objekte „Easy 700 Neuer Wert Merker N1 bis N16“	334
– Objekt „Easy Link Eingangsdaten“	335
– Objekt „Easy Link Ausgangsdaten“	335
Funktionsbaustein-Daten lesen/schreiben	336
– Verfahrensweise	336
– Objekt „Easy 700/800 Auswahl Daten Funktionsbaustein“	337
– Objekt „Easy 700/800 Lesen Daten Funktionsbaustein“	339
– Objekt „Easy 700/800 Schreiben Daten Funktionsbaustein“	341
– Analogwertvergleicher/Schwellwertschalter:	
A1 - A16	343
– Zähler C1 bis C16	345

– Betriebsstundenzähler O1 bis O4	347
– Zeitrelais T1 bis T16	349
– Jahresschaltuhren Y1 bis Y8	352
– Wochenschaltuhren 71 bis 78	354
Datum und Uhrzeit lesen/schreiben	356
– Objekt „Easy Uhr“	356
Sommer-/Winterzeit lesen/schreiben	356
– Objekte „Easy 700 Sommer-/Winterzeit Einstellung“	356
<hr/>	
12 Prozessdaten easy800/MFD (DPV1)	363
Objektübersicht	364
Zugriff auf Objekte	366
Identifikation lesen	367
– Objekt „Easy Identifikation“	367
Betriebsart lesen/schreiben	367
– Objekt „Easy Betriebsart“	367
Abbild lesen/schreiben	367
– Objekt „Easy 700/800 Eingänge“	367
– Objekte „Easy 700/800 Analogeingänge IA1 bis IA4“	367
– Objekt „Easy 700/800 Ausgänge“	367
– Objekt „Easy 700/800 Tasten“	367
– Objekt „Easy 800 Analogausgang“	367
– Objekt „Easy 800 Lokale Diagnose“	370
– Objekte „Easy 800 Eingänge Netzwerkteilnehmer IW1 bis IW8“	372
– Objekte „Easy 800 Link-Eingänge Netzwerkteilnehmer RW1 bis RW8“	374
– Objekte „Easy 800 Ausgänge Netzwerkteilnehmer SW1 bis SW8“	376
– Objekte „Easy 800 Link-Ausgänge Netzwerkteilnehmer SW1 bis SW8“	378
– Objekte „Easy 800 Receive-Data Netzwerkteilnehmer RNW1 bis RNW8“	380
– Objekte „Easy 800 Send-Data Netzwerkteilnehmer SNW1 bis SNW8“	382
– Objekte „Easy 800 Bit Merker M1 bis M96“	384
– Objekte „Easy 800 Byte Merker MB1 bis MB96“	386

– Objekte „Easy 800 Wort Merker MW1 bis MW96“	388
– Objekte „Easy 800 Doppelwort Merker MD1 bis MD96“	390
– Objekt „Easy 800 8 Byte Daten“	392
– Objekt „Easy 800 16 Byte Daten“	394
– Objekt „Easy 800 32 Byte Daten“	396
– Objekt „Easy 800 64 Byte Daten“	399
– Objekt „Easy Link Eingangsdaten“	402
– Objekt „Easy Link Ausgangsdaten“	402
Funktionsbaustein-Daten lesen/schreiben	403
– Verfahrensweise	403
– Analogwertvergleicher A1 bis A32	404
– Arithmetikbausteine AR1 bis AR32	406
– Block Compare Bausteine BC1 bis BC32	408
– Block Transfer Bausteine BT1 bis BT32	410
– Boolesche Verknüpfung Bausteine BV1 bis BV32	412
– Zähler C1 bis C32	414
– Frequenzzähler CF1 bis CF4	416
– Schneller Zähler CH1 bis CH4	417
– Inkrementalzähler CI1 bis CI4	419
– Vergleichler CP1 bis CP32	420
– Textausgabe Bausteine D1 bis D32	421
– Datenbausteine DB1 bis DB32	423
– PID-Regler DC1 bis DC32	424
– Diagnose DG01...DG16	426
– Signalglättungsfilter FT1 bis FT32	427
– Bausteine Empfang Netzdaten GT1 bis GT32	428
– Wochenschaltuhren HW1 bis HW32	430
– Jahresschaltuhren HY1 bis HY32	432
– Bedingter Sprung JC01...JC32	434
– Wertskalierungen LS1 bis LS32	435
– Masterreset MR1 bis MR32	437
– Datenmultiplexer MX01...MX32	439
– Zahlenwandler NC1 bis NC32	441
– Betriebsstundenzähler OT1 bis OT4	443
– Impulsausgabe PO01...PO02	444
– Bausteine Senden Netzdaten PT1 bis PT32	446
– Pulsweitenmodulation PW1 bis PW2	447

– Baustein Uhr synchronisieren SC1	448
– Serielle Ausgabe SP01...SP32	449
– Schieberegister SR01...SR32	450
– Baustein Sollzykluszeit ST1	452
– Zeitrelais T1 bis T32	453
– Tabellenfunktion TB01...TB32	455
– Wertbegrenzung VC1 bis VC32	457
Datum und Uhrzeit lesen/schreiben	458
– Objekt „Easy Uhr“	458
Sommer-/Winterzeit lesen/schreiben	458
– Objekte „Easy 800 Sommer-/Winterzeit Einstellung“	458
DPV1-Fehlermeldungen	464

Anhang	467
Was ist wenn...?	467
Übersicht der Befehle	468
– easy600	468
– easy800/MFD	471
Technische Daten	473
– Allgemeines	473
– Klimatische Umgebungstemperaturen	473
– Mechanische Umgebungsbedingungen	474
– Elektromagnetische Verträglichkeit (EMV)	474
– Isolationsfestigkeit	475
– Werkzeug und Anschlussquerschnitte	475
– Stromversorgung	475
– LED-Anzeigen	475
– PROFIBUS-DP	476
Abmessungen	477

Glossar	479
----------------	-----

Stichwortverzeichnis	487
-----------------------------	-----

Zu diesem Handbuch

Änderungsprotokoll

Gegenüber der letzten Ausgabe haben sich folgende wesentliche Änderungen ergeben:

Redaktionsdatum	Seite/ Kapitel	Stichwort	neu	Änderung
08/02	45	Abschnitt „PROFIBUS-Zertifizierung“	✓	
05/03	allg.	easy800/MFD	✓	
07/04	allg.	easy700	✓	
03/05	Kapitel 10, 11, 12	easy600/700/800, MFD mit DPV1-Protokoll	✓	
02/08	155	„Auswirkung auf easyLink“		✓
	48	Tabelle 3		✓
	Kapitel 9	neue Funktionsbausteine DG, JC, MX, PO, SP, SR, TB	✓	
	Kapitel 12		✓	
09/10	allg.	Umstellung auf Eaton-Bezeichnungen	✓	

Zielgruppe

Dieses Handbuch richtet sich an Automatisierungstechniker und Ingenieure. Es werden fundierte Kenntnisse zum Feldbus PROFIBUS-DP und zur Programmierung eines PROFIBUS-DP-Masters vorausgesetzt. Außerdem sollten Sie mit der Handhabung des Steuerrelais „easy“ bzw. des Multi-Funktions-Displays MFD vertraut sein.

**Weitere Handbücher
zum Gerät**

Grundsätzlich gelten die Bedienhandbücher zu:

- „Steuerrelais easy412, easy600“ (MN05013004Z-DE; frühere Bezeichnung AWB2528-1304-D)
- „Steuerrelais easy700“ (MN05013003Z-DE; frühere Bezeichnung AWB2528-1508D)
- „Steuerrelais easy800“ (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D)
- „Multi-Funktions-Display MFD-Titan“ (MN05002001Z-DE; frühere Bezeichnung AWB2528-1480D).

Alle Handbücher stehen im Internet als PDF-Datei zum Download zur Verfügung. Für ein schnelles Auffinden geben Sie unter <http://www.eaton.com/moeller> als Suchbegriff die Dokumentationsnummer ein.

Gerätebezeichnung

In diesem Handbuch werden für die Gerätetypen folgende Kurzbezeichnungen genommen, sofern die Beschreibung auf alle diese Typen zutrifft:

- easy600 für
 - EASY6...-AC-RC(X)
 - EASY6...-DC-.C(X)
- easy700 für
 - EASY719-AB...
 - EASY719-AC...
 - EASY719-DA...
 - EASY719-DC...
 - EASY721-DC...
- easy800 für
 - EASY819-...
 - EASY820-...
 - EASY821-...
 - EASY822-...

- MFD-CP8... für
 - MFD-CP8-ME
 - MFD-CP8-NT
- MFD-...-CP8/CP10 für
 - MFD-CP8...
 - MFD-CP10...
- MFD-CP10... für
 - MFD-CP10-ME
 - MFD-CP10-NT
- easy-AB für
 - EASY719-AB...
- easy-AC für
 - EASY6...-AC-RC(X),
 - EASY719-AC
 - EASY8...-AC-...
- easy-DC für
 - EASY6...-DC-...
 - EASY719-DC-...
 - EASY8...-DC-...
- easy-DA für
 - EASY719-DA...

Abkürzungen

In diesem Handbuch werden Abkürzungen eingesetzt, die folgende Bedeutung haben:

hex: Hexadezimal (Zahlensystem mit der Basis 16)

dez: Dezimal (Zahlensystem mit der Basis 10)

bcd: Binär codierter Dezimalcode

WB: **Wertebereich**

PC: **Personal Computer**

len: **length** (Länge)

Lesekonventionen

Für eine gute Übersichtlichkeit finden Sie auf den linken Seiten im Kopf die Kapitelüberschrift und auf den rechten Seiten den aktuellen Abschnitt, Ausnahmen sind Kapitelanfangsseiten und leere Seiten am Kapitelende.

► zeigt Handlungsanweisungen an.



Achtung!

warnet vor leichten Sachschäden.



Warnung!

warnet vor schweren Sachschäden und leichten Verletzungen.



Gefahr!

warnet vor schweren Sachschäden und schweren Verletzungen oder Tod.



macht Sie aufmerksam auf interessante Tipps und Zusatzinformationen

1 Zu EASY204-DP

Das Kommunikationsmodul EASY204-DP wurde für Automatisierungsaufgaben mit dem Feldbus PROFIBUS-DP entwickelt. Das EASY204-DP stellt ein Gateway dar und kann nur in Verbindung mit den Steuerrelais easy600, easy700, easy800 oder MFD-Basisgeräten betrieben werden. Das Steuergerät „easy“ bzw. MFD mit PROFIBUS-DP-Gateway arbeitet immer als ein Slave-Teilnehmer im Netzwerk.

Systemübersicht

Die „easy“-PROFIBUS-DP-Slaves werden innerhalb eines PROFIBUS-DP-Systems integriert.

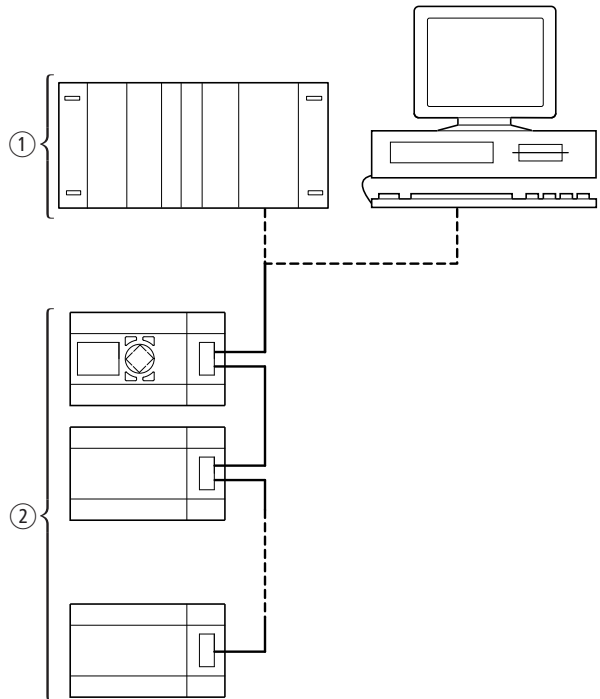


Abbildung 1: Einbindung EASY204-DP im DP-Netzwerk

- ① Masterbereich, SPS oder PC
- ② Slavebereich z. B.: Steuerrelais „easy“/MFD mit DP-Anschaltung

Aufbau des Gerätes

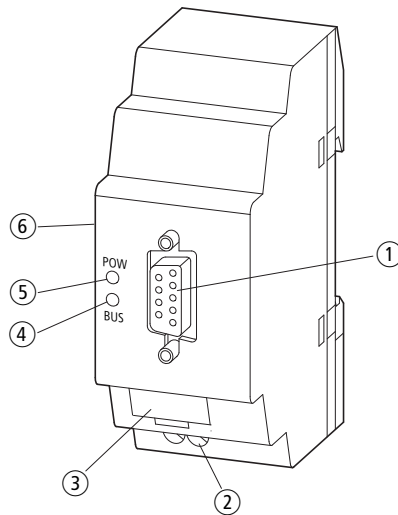


Abbildung 2: Geräteansicht

- ① PROFIBUS-DP-Anschluss, 9-polige SUB-D-Buchse
- ② Versorgungsspannung 24 V DC
- ③ Gerätekenzeichnungsschild
- ④ BUS-Kommunikations-LED
- ⑤ POW-Betriebsanzeige-LED
- ⑥ easyLink-Buchse

**Gerätefunktions-
beschreibung**

Das Modul EASY204-DP ermöglicht der easy- bzw. MDF-Familie die Anbindung an ein PROFIBUS-DP-Kommunikationsnetzwerk. Durch Auswahl der entsprechenden Ein-/Ausgabe-Module bzw. mit Hilfe der DPV1-Dienste „Read“ und „Write“ (ab Geräteversion 07) können die nachfolgenden Daten übertragen werden:

easy600/700/800, MFD-CP8...

- S1 bis S8
Ausgangsdaten des Basisgerätes, RUN/STOP
(Lesen, vom PROFIBUS-DP-Master aus gesehen)
- R1 bis R16
Eingangsdaten des Basisgerätes, RUN/STOP
(Schreiben, vom PROFIBUS-DP-Master aus gesehen)
- Alle Daten der Funktionsrelais
(Lesen, vom PROFIBUS-DP-Master aus gesehen)
 - Zeitrelais
 - Zählerrelais
 - Schaltuhr
 - Analogwertvergleicher
 - Wochentag, Uhrzeit, Sommer-/Winterzeit
 - Alle Zustände der easy600-Kontakte.
- Die Sollwerte der Funktionsrelais
(Schreiben, vom PROFIBUS-DP-Master aus gesehen)
 - Zeitrelais
 - Zählerrelais
 - Schaltuhr
 - Analogwertvergleicher
 - Wochentag, Uhrzeit, Sommer-/Winterzeit.

easy800/MFD-CP8...

- sämtliche Merker und easyNet-Daten
- Funktionsbausteine (Lesen/Schreiben aus Sicht des Masters)
 - Arithmetikbausteine
 - Frequenzzähler, schnelle Zähler, Inkrementalzähler
 - Wochen- und Jahreszeitschaltuhr
 - Betriebsstundenzähler
 - PID-Regler
 - PWM (Pulsweitenmodulation)
 - Echtzeituhr.

Hardware- und Betriebssystem-Voraussetzungen

Das Erweiterungsgerät EASY204-DP arbeitet mit den Basisgeräten easy600, easy700, easy800 und MFD ab den nachfolgenden Betriebssystemversionen zusammen:

Basisgerät		Erweiterungsgerät EASY204-DP	
Geräteversion	BTS-Version	Geräteversion \geq 04	Geräteversion \geq 05
easy600			
\geq 04	ab 2.4	×	×
easy700			
\geq 01	ab 1.10.xxx	–	×
easy800			
\geq 04	ab 1.10.xxx	–	×
MFD-CP8...			
\geq 01	ab 1.10.xxx	–	×
MFD-CP10...			
\geq 01	ab 1.00	–	×

Die Geräteversion des entsprechenden Basis- oder Erweiterungsgerätes ist auf der rechten Gehäuseseite angegeben.

Beispiel: EASY204-DP: 03-228xxxxxxx (03 = Geräteversion)

Die Betriebssystemversion (BTS) des entsprechenden Basisgerätes können Sie über die easySoft auslesen. Bei den Geräten easy700, easy800 und MFD-CP8... besteht auch die Möglichkeit, die Information direkt am Gerät auszulesen. Lesen Sie hierzu bitte im entsprechenden Handbuch nach.

Änderungshistorie

Änderungen von Version 04 zu Version 05:

- Erweiterung des easyLink-Protokolls zur Anbindung an das easy800.
- Zyklischer Datentransfer um 1-Byte-Modul erweitert.

Änderungen von Version 05 zu Version 06:

- Erweiterung des easyLink-Protokolls zur Anbindung an das easy700.
- Zyklischer Datentransfer (3-Byte-Modul) im Byte 0 angepasst (siehe auch Kapitel „Was ist wenn...?“ auf Seite 467).

Änderungen von Version 06 zu Version 07:

- Erweiterung um die DPV1-Dienste „Read“ und „Write“.
- Erweiterung um Module für zusätzliche Ein-/Ausgangsdaten für das easy800.

Sachwidriger Einsatz

„easy“ darf nicht eingesetzt werden als Ersatz für sicherheitsrelevante Steuerungen wie

- Brenner-,
- NOT-AUS-,
- Kran- oder
- Zweihand-Sicherheitssteuerungen.

2 Installation

Es gelten die gleichen Richtlinien wie für die easy600, easy700, easy800 und MFD-Basisgeräte mit Erweiterung.

EASY204-DP an das Grundgerät anschließen

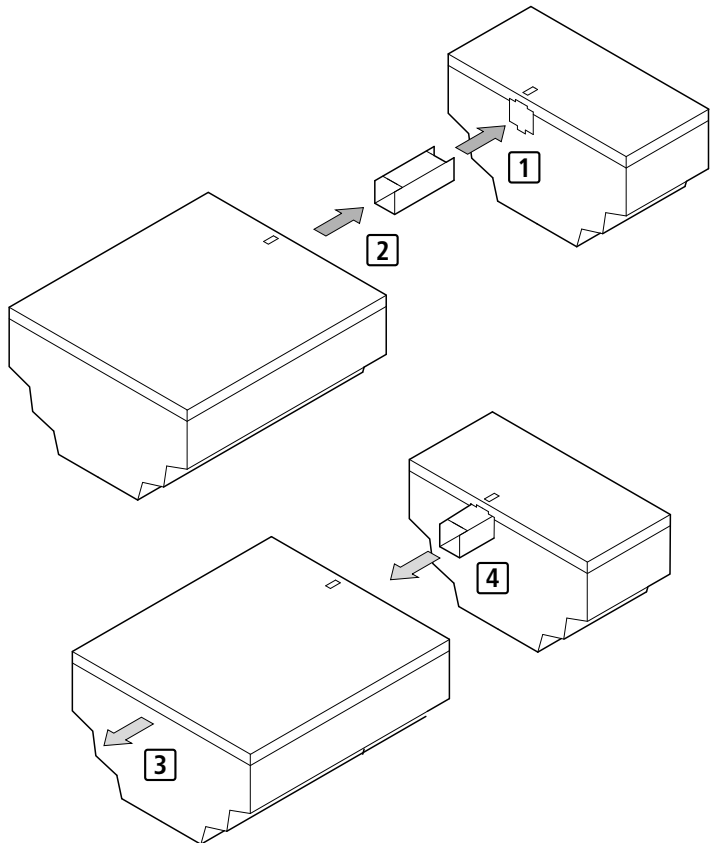


Abbildung 3: EASY204-DP an Grundgerät montieren **1** + **2**,
bzw. demontieren **3** + **4**

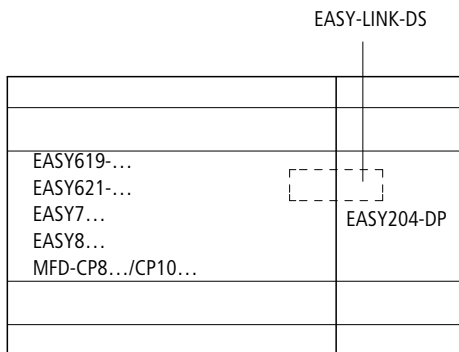


Abbildung 4: Verbindung zwischen Basisgerät und EASY204-DP

Versorgungsspannung anschließen

Das Gerät EASY204-DP wird mit einer 24-V-DC-Versorgungsspannung betrieben (→ Abschnitt „Technische Daten“ ab Seite 473).



Gefahr!

Bei der 24-V-Versorgung ist auf eine sichere elektrische Trennung der Kleinspannung (SELV) zu achten.

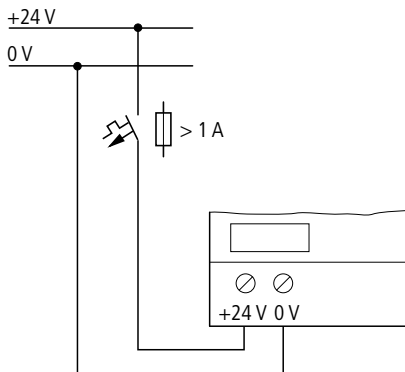
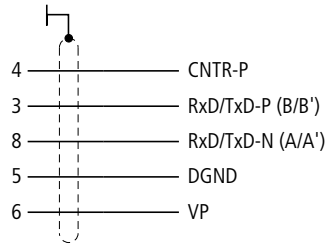
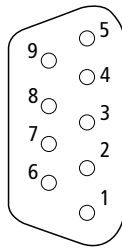


Abbildung 5: Standardanschluss

PROFIBUS-DP anschließen Ein 9-poliger SUB-D-Stecker verbindet den PROFIBUS-DP-Anschluss des Gerätes mit dem Feldbus PROFIBUS-DP. Zum Anschluss nehmen Sie bitte einen speziellen PROFIBUS-DP-Stecker und die spezielle PROFIBUS-DP-Leitung aus dem Zubehörangebot von Eaton. Die Art des Kabels hat Einfluss auf die verfügbare maximale Busleitungslänge und somit auch auf die Übertragungsrate.

**Anschlussbelegung
PROFIBUS-DP**



Pin	Signalname	Bezeichnung
1	nicht belegt	-
2	nicht belegt	-
3	RxD/TxD-P (B-Line)	Empfangs-/Sende-Daten-P
4	CNTR-P/RTS	Request to Send
5	DGND	Datenbezugspotential
6	VP	+5V DC für externen Busabschluss
7	nicht belegt	-
8	RxD/TxD-N (A-Line)	Empfangs-/Sende-Daten-N
9	nicht belegt	-

Für den Datenverkehr reichen die Anschlüsse 3, 8 und Schirm.

Abschlusswiderstände

Der erste und der letzte Teilnehmer in einem Bussegment muss den Bus mit eingeschaltetem Abschlusswiderstand abschließen. Der Busabschlusswiderstand wird extern aufgeschaltet. Diese externe Aufschaltung kann entweder als separater Abschlusswiderstand oder durch einen speziellen SUB-D-Stecker mit integriertem Busabschluss realisiert werden.

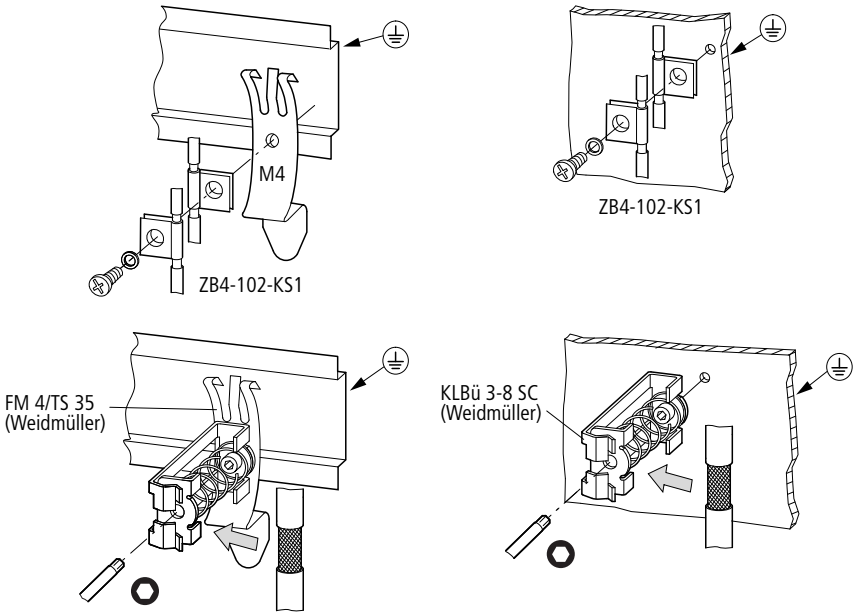
Die PROFIBUS-DP-Datenstecker von Eaton erlauben am Stecker das Ein- und Ausschalten der beiden Abschlusswiderstände.

**EMV-gerechte
Verdrahtung**

Durch eventuelle elektromagnetische Beeinflussung des Kommunikationsfeldbusses können unerwünschte Störungen auftreten. Diese lassen sich durch geeignete EMV-Maßnahmen schon im Vorfeld minimieren. Hierzu zählen:

- EMV-gerechter Systemaufbau der Anlage,
- EMV-gerechte Leitungsführung und
- Maßnahmen, die keine großen Potentialunterschiede aufkommen lassen,
- richtige Installation des PROFIBUS-Systems (Kabel, Anschluss des Bussteckers,...).

Die elektromagnetische Beeinflussung lässt sich durch das Auflegen des Schirms wesentlich verringern. Die folgenden zwei Abbildungen stellen das richtige Auflegen der Abschirmung dar.



Potentialtrennungen

Für die Schnittstellen des EASY204-DP gelten folgende Potentialtrennungen:

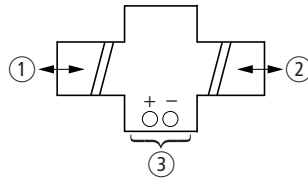


Abbildung 6: Potentialtrennung zwischen Versorgungsspannung und Ausgängen

- ① sichere Trennung easyLink 240 V AC
- ② einfache Trennung PROFIBUS-DP
- ③ Versorgungsspannung 24 V DC

Übertragungsraten – automatische Baudratenerkennung

Nach dem Einschalten erkennt das Modul EASY204-DP automatisch die im Kommunikationsnetzwerk verwendete Übertragungsrate. Hierzu muss allerdings mindestens ein Teilnehmer im Netzwerk gültige Nachrichten verschicken.

Nachfolgende Übertragungsraten werden unterstützt:

- 9,6 kBit/s bis 12000 kBit/s

Maximale Entfernungen/ Busleitungslängen

Zwei Varianten der Busleitung sind in der IEC 61158 spezifiziert. Der Leitungstyp B sollte bei neuen Anwendungen nicht mehr verwendet werden, da er veraltet ist. Mit dem Leitungstyp A können alle Übertragungsraten bis 12000 kBit/s genutzt werden. Neben der Standardleitung sind auch Leitungen für Erdverlegung, Girlandenaufhängung und Schleppkabel verfügbar.

Die Leitungsparameter sind wie folgt:

Parameter	Leitungstyp A
Wellenwiderstand in Ω	135...165 bei 3...20 MHz
Betriebskapazität (pF/m)	< 30
Schleifenwiderstand (Ω /km)	< 110
Aderndurchmesser (mm)	> 0,64
Aderquerschnitt (mm ²)	> 0,34

Mit den spezifizierten Leitungsparametern ergeben sich nachfolgende Leitungsausdehnungen eines Bussegments.

Entfernung zwischen zwei Teilnehmern, wenn Leitungstyp „A“ nach IEC 61158 verwendet wird:

Baudrate [kBit/s]	max. Leitungslänge Leitung Typ „A“ [m]
9,6	1200
19,2	1200
93,75	1200
187,5	1000
500	400
1500	200
3000	100
6000	100
12000	100

Entfernung zwischen zwei Teilnehmern, wenn Leitungstyp „B“ nach IEC 61158 verwendet wird:

Baudrate [kBit/s]	max. Leitungslänge Leitung Typ „B“ [m]
9,6	1200
19,2	1200
93,75	1200
187,5	1000
500	400
1500	–

3 Gerät betreiben

Erstes Einschalten

- ▶ Prüfen Sie vor dem Einschalten, ob die Stromversorgung, der Busanschluss und die Verbindung zum Basisgerät richtig angeschlossen sind.
- ▶ Schalten Sie die Versorgungsspannung des Basisgerätes und der PROFIBUS-DP-Erweiterung ein.

Die Power-LED des EASY204-DP leuchtet. Die BUS-LED ist aus (keine Kommunikation über PROFIBUS-DP).

Am Basisgerät wird die GW-Information (intelligenter Teilnehmer angekoppelt) angezeigt.

Basisgerät	Gerätrevision	GW-Anzeige
easy600	04	statisch
easy700	ab 01	blinkend
easy800	04	statisch
	ab 05	blinkend
MFD-CP8...	01	statisch
	ab 02	blinkend
MFD-CP10...	01	blinkend

Sobald das Gerät im Netzwerkverbund PROFIDUS-DP eingebunden ist, schaltet die BUS-LED auf „statisch“ um und der Zustand des GW wechselt auch bei den Geräten mit blinkenden GW auf statisch.



Gültige Daten über PROFIBUS-DP an das Grundgerät werden erst übertragen, wenn das GW statisch angezeigt wird.

Besitzt das PROFIBUS-DP-Gerät die Werkseinstellungen, muss die PROFIBUS-DP-Teilnehmeradresse eingestellt werden.

**PROFIBUS-DP Teilnehmer-
adresse einstellen**

Jeder PROFIBUSDP-Teilnehmer benötigt eine eindeutige Adresse in der PROFIBUS-DP-Struktur. Die PROFIBUS-DP-Adresse können Sie bei EASY204-DP auf zwei Arten einstellen:

- Mit der integrierten Anzeige und Tastatur am „easy“- oder MFD-Titan-Basisgerät
- Mit easySoft Version 3.01 oder höher am PC.

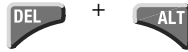
Adressbereich: 001 bis 126

Adresse am Basisgerät mit Anzeige einstellen

Voraussetzungen:

- Basisgerät (easy600, easy700, easy800 oder MFD-Titan) und Erweiterungsgerät EASY204-DP sind mit Spannung versorgt.
- Das Basisgerät ist aufgeschlossen (kein Passwortschutz aktiviert).
- Das Basisgerät besitzt eine gültige Betriebssystemversion (→ Seite 19).
- Das Basisgerät muss im STOP-Modus sein.
- Das Gerät EASY204-DP kommuniziert nicht mit dem PROFIBUS-DP-Master (Bus-LED ist aus).

- ▶ Wechseln Sie in durch gleichzeitiges Betätigen von DEL + ALT in das Sondermenü.



```
PASSWORT...  
SYSTEM  
GB D F E  
I..
```

- ▶ Wählen Sie mit den Cursor-Tasten ^ oder v zu KONFIGURATOR.

```
PASSWORT...  
SYSTEM  
GB D F E  
I..
```

- ▶ Bestätigen Sie mit „OK“



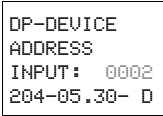
- ▶ Wählen Sie bei den Geräten easy800/MFD den Menüpunkt LINK...

```
NET...  
LINK...
```

- ▶ Bestätigen Sie mit „OK“.



Die easy600- und easy700-Geräte zeigen direkt den nachfolgenden Dialog an:



- Stellen Sie die Adresse ein:
 - ➔ Den Wert der aktuellen Ziffer mit ^ oder v.
 - Zur nächsten Ziffer mit < oder >.

2 9 0 1



0 0 0 1 ←

→ 0 0 0 1

1 0 9 2



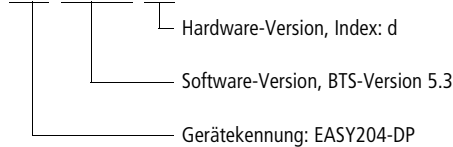
- Übernehmen Sie die Adresse mit OK oder



- brechen Sie die Adresseingabe ab.

Informationen zur 4. Displayzeile:

xxx - x x . x x - x x
 204 - 05 30 - d



Adresse mittels easySoft einstellen

Bei easySoft Version 3.1

⟨Menü → Online → Erweiterungsgeräte konfigurieren⟩

Ab easySoft Version 4.01 und höher

⟨Menü → Kommunikation → Konfiguration → Erweiterungsgeräte → EASY204-DP⟩.



Das Menü ist nur in der Kommunikationsansicht verfügbar; aktivieren Sie deshalb die Registerkarte „Kommunikation“.

LED-Statusanzeigen

Das Erweiterungsgerät EASY204-DP besitzt zwei Anzeigeleuchten.

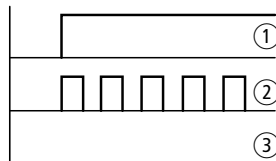
POW-LED, Funktion

Abbildung 7: Anzeigefunktion POW-LED

- ① LED Dauerlicht:
 - Versorgungsspannung vorhanden
 - Kommunikation mit dem Basisgerät unterbrochen
- ② LED blinkt:
 - Versorgungsspannung vorhanden
 - Kommunikation mit dem Basisgerät in Ordnung
- ③ LED keine Anzeige:
 - Keine Versorgungsspannung vorhanden
 - Kommunikation mit dem Basisgerät unterbrochen

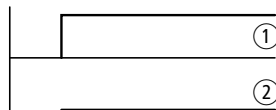
BUS-LED, Funktion

Abbildung 8: Anzeigefunktion BUS-LED

- ① LED Dauerlicht:
 - PROFIBUS-DP-Kommunikation ist in Ordnung
- ② LED keine Anzeige:
 - Keine PROFIBUS-DP-Kommunikation vorhanden

Zykluszeit easy-Basisgerät Die Kommunikation zwischen dem Basisgerät und EASY204-DP über den easyLink verlängert die Zykluszeit des Basisgerätes.

Im Extremfall kann sich die Zykluszeit um 40 ms verlängern.

Bitte beachten Sie dieses bei den Reaktionszeiten des Basisgerätes.

4 PROFIBUS-DP-Funktionen

Konfiguration des DP-Masters, Klasse 1

Für die Konfiguration des übergeordneten DP-Masters führen Sie prinzipiell folgende Schritte durch:

- ▶ Fügen Sie die GSD-Datei „Moe4d10.gsd“ (ab Geräteversion 07) bzw. „Moel4d10.gsd“ (bis Geräteversion 06) in die GSD-Datenbasis des Konfigurationstools Ihres DP-Masters ein.
- ▶ Fügen Sie einen Teilnehmer EASY204-DP in die Topologie des zu konfigurierenden Bussegmentes ein.
- ▶ Weisen Sie diesem Teilnehmer die vorgesehene Teilnehmeradresse zu.
- ▶ Wählen Sie für diesen Teilnehmer bis zu 5 der vorgeschlagenen Module für den zyklischen Datentransfer aus, → Abschnitt „Slave-Module“.
- ▶ Wiederholen Sie die Schritte 2 bis 4 für jeden in die Topologie einzufügenden EASY204-DP-Teilnehmer.
- ▶ Speichern Sie die Konfiguration ab.
- ▶ Übertragen Sie die Konfiguration in den DP-Master.



Beachten Sie die Dokumentation des DP-Masters, wenn Sie die Konfiguration vornehmen.

Slave-Module

Das Erweiterungsmodul EASY204-DP ist ein PROFIBUS-DP-Slave nach IEC 61158.

Im PROFIBUS-DP-Konfigurator der Mastersteuerung können Sie unter Verwendung der entsprechenden GSD-Datei nachfolgende Module des Slaves EASY204-DP auswählen. Die Module werden in den nachfolgenden Kapiteln detailliert beschrieben.

Modul-Bezeichnung	Eingänge (Byte)	Ausgänge (Byte)	Ein-/Ausgänge (Byte)	Dienst	Unterstützte Geräte	Kenennung in GSD-Datei	Seite ↑
Steuerebene							
1: Easy 600 Steuerbefehle	–	–	7	<ul style="list-style-type: none"> • Uhrzeit • Abbild • Funktionsbausteine 	easy600	0xB6	59
2: Easy700/800 Steuerbefehle	–	–	9	<ul style="list-style-type: none"> • Echtzeituhr • Abbild • Funktionsbausteine 	easy700, easy800, MFD-CP8..., MFD-CP10...	0xB8	101
Ein-/Ausgangsebene							
3: Eingänge 3 Byte	3	–	–	<ul style="list-style-type: none"> • Daten lesen: S1 - S8 • Betriebsart 	easy600, easy700, easy800, MFD-CP8..., MFD-CP10...	0x92	48
4: Ausgänge 3 Byte	–	3	–	<ul style="list-style-type: none"> • Daten schreiben: R1 - R8, R9 - R16 • Betriebsart 		0xA2	50
5: Eingänge 1 Byte	1	–	–	<ul style="list-style-type: none"> • Daten lesen: S1 - S8 		0x90	50
6: Ausgänge 1 Byte	–	1	–	<ul style="list-style-type: none"> • Daten schreiben: R1 - R8 		0xA0	54

Modul-Bezeichnung	Eingänge (Byte)	Ausgänge (Byte)	Ein-/Ausgänge (Byte)	Dienst	Unterstützte Geräte	Kennung in GSD-Datei	Seite ↑
Zusätzliche Ein-/Ausgangsdaten (ab Geräteversion 07)							
7: Easy 800 Zusatzeingänge 4 Byte	4	–	–	• Daten lesen: MD63	easy800	0x13	57
8: Easy 800 Zusatzeingänge 8 Byte	8	–	–	• Daten lesen: MD63 - MD64		0x17	57
9: Easy 800 Zusatzeingänge 12 Byte	12	–	–	• Daten lesen: MD63 - MD65		0x1B	57
10: Easy 800 Zusatzeingänge 16 Byte	16	–	–	• Daten lesen: MD63 - MD66		0x1F	57
11: Easy 800 Zusatzausgänge 4 Byte	–	4	–	• Daten schreiben: MD59		0x23	58
12: Easy 800 Zusatzausgänge 8 Byte	–	8	–	• Daten schreiben: MD59 - MD60		0x27	58
13: Easy 800 Zusatzausgänge 12 Byte	–	12	–	• Daten schreiben: MD59 - MD61		0x2B	58
14: Easy 800 Zusatzausgänge 16 Byte	–	16	–	• Daten schreiben: MD59 - MD62		0x2F	58
Ohne Funktion							
15: Leerplatz ¹⁾	–	–	–	–	easy600, easy700, easy800, MFD-CP8..., MFD-CP10...	0x00	–

1) In einem PROFIBUS-DP-Konfigurator bietet das EASY204-DP Steckplätze an, die Sie mit den für Ihre Applikation benötigten Modulen belegen können. Nicht benötigte Steckplätze können Sie mit dem Modul „Leerplatz“ belegen.

Beachten Sie die Betriebssystemvoraussetzungen auf Seite 19!

Für die Konfiguration der Module gelten folgende Regeln:

- Verwenden Sie aus dem Bereich „Steuerebene“ maximal ein Modul.
- Verwenden Sie aus dem Bereich „Ein-/Ausgangsebene“ maximal jeweils ein Eingangs- und ein Ausgangsmodul.
- Verwenden Sie aus dem Bereich „zusätzliche Ein-/Ausgangsdaten“ maximal jeweils ein Eingangs- und ein Ausgangsmodul (ab Geräteversion 07).

Diagnosedaten

Das EASY204-DP liefert außer der genormten DP-Standard-Diagnose zusätzliche Diagnose-Informationen, die in einem DPV0-Master der Klasse 1 als „Gerätespezifische Diagnose“ und – ab Geräteversion 07 – in einem DPV1-Master der Klasse 1 als „Statusdiagnose“ erscheinen.

Format der Diagnose-Informationen

Das Lesen der Diagnose erfolgt direkt über die DP-Diagnosebefehle oder über die bei der PROFIBUS-DP-Konfiguration definierten Diagnosebytes des DP-Masters. Beachten Sie hierzu die Dokumentation des Master-Gerätes.

Bei einem DPV0-Master werden 8, bei einem DPV1-Master 11 Octets als Diagnose eingelesen. Sie beinhalten folgende Informationen:

Tabelle 1: Adresslage der Diagnoseinformation in einem DPV0- bzw. DPV1-Master

Diagnose- position	Bezeichnung	
	DPV0-Master	DPV1-Master
Octet 1		
Bit 0	Station nicht existent	
Bit 1	Station nicht bereit	
Bit 2	Konfigurations-Fehler	
Bit 3	Zusätzliche Diagnose-Informationen	
Bit 4	Funktion nicht unterstützt	
Bit 5	Ungültige Antwort DP-Slave	
Bit 6	Parametrierungs-Fehler	
Bit 7	Master bereits vorhanden	
Octet 2		
Bit 0	Parametrierungs-Anfrage	
Bit 1	Statische Diagnose	
Bit 2	Nicht genutzt	
Bit 3	Ansprechüberwachung aktiviert	
Bit 4	Freeze-Modus aktiv	
Bit 5	Sync-Modus aktiv	
Bit 6	Nicht genutzt	
Bit 7	Slave deaktiviert	
Octet 3		
Bit 0 - 6	Nicht genutzt	
Bit 7	Überlauf zusätzliche Diagnose-Information	
Octet 4	Stationsadresse DP-Master	
Octet 5 und 6	Ident-Nummer DP-Slave	
Octet 7	Länge zusätzliche Diagnose-Information	

Diagnose- position	Bezeichnung	
	DPV0-Master	DPV1-Master
Octet 8		Status Type
Bit 0	Kommunikations- fehler Easy Link	–
Bit 1 - 7	Nicht genutzt	–
Octet 9	–	Slot Number
Octet 10	–	Status Specifier
Octet 11	–	
Bit 0	–	Kommunikations- fehler easyLink
Bit 1	–	Ungültiger Befehl für easy-Betriebsart
Bit 2	–	Zusätzliche E/A-Daten noch nicht verfügbar
Bit 3 - 7	–	Nicht genutzt



Beachten Sie beim Zugriff auf die Diagnose-Information „Ident Nummer“ das im PROFIBUS-DP verwendete Motorola-Kodierungsformat (Octet N: High-Byte, Octet N+1: Low-Byte) für Daten im WORD-Format. Sollte das Datenverarbeitungsformat in Ihrem DP-Master-System hiervon abweichen und die DP-Zugriffsbefehle keine automatische Konvertierung beinhalten, müssen Sie die notwendige Konvertierung in Ihrem Anwendungsprogramm selber vornehmen. Beachten Sie diesbezüglich die Dokumentation Ihres DP-Master-Systems.

Bedeutung der Diagnose-Informationen

Die eingelesenen Diagnose-Informationen haben folgende Bedeutung:

Tabelle 2: Dateninhalte der Diangnoseinformationen

Bezeichnung	Bedeutung	Erläuterung/Abhilfe
Ansprechüberwachung aktiviert	Die Ansprechüberwachung im EASY204-DP wurde ordnungsgemäß aktiviert.	Sollzustand
Freeze-Modus aktiv	Der DP-Master hat das synchrone Lesen von Dateneingängen mehrerer Teilnehmer aktiviert.	Gewollte Anwender-Aktion
Funktion nicht unterstützt	Der DP-Master hat eine vom EASY204-DP nicht unterstützte Funktion angefordert.	Überprüfen Sie die Konfiguration des DP-Masters.
Ident-Nummer DP-Slave	Enthält die Ident-Nummer des EASY204-DP: 4D10 _{hex}	
Kommunikationsfehler Easy Link	Die Kommunikation zwischen EASY204-DP und easy/MFD ist unterbrochen	Überprüfen Sie die Verbindung zwischen EASY204-DP und easy/MFD.
Konfigurations-Fehler	Der DP-Master hat ein ungültiges Konfigurations-Telegramm an das EASY204-DP gesendet (z.B. falsche Länge der Dateneingänge und/oder Datenausgänge)	Überprüfen Sie die Konfiguration des DP-Masters.
Länge zusätzliche Diagnose-Information	Enthält die Länge der zusätzlichen Diagnose-Informationen: 02 _{hex} bei DPV0 05 _{hex} bei DPV1	
Master bereits vorhanden	Das EASY204-DP ist von einem anderen DP-Master belegt.	
Nicht genutzt	Enthält keine auszuwertende Information.	
Parametrierungs-Anfrage	Das EASY204-DP wartet auf das Parametrierungs-Telegramm des DP-Masters.	Temporärer Zustand

Bezeichnung	Bedeutung	Erläuterung/Abhilfe
Parametrierungs-Fehler	Der DP-Master hat ein ungültiges Parametrierungs-Telegramm an das EASY204-DP gesendet.	Überprüfen Sie die Konfiguration des DP-Masters
Slave deaktiviert	Der DP-Master hat das EASY204-DP aus seiner zyklischen Bearbeitung herausgenommen (deaktiviert).	Gewollte Anwender-Aktion
Slot Number	Enthält die Slot-Number, aus der die zusätzlichen Diagnose-Informationen (Status) stammen: 00 _{hex}	
Station nicht bereit	Das EASY204-DP ist für die Kommunikation noch nicht bereit (Initialisierungsphase).	Temporärer Zustand
Station nicht existent	Unter der verwendeten Stationsadresse meldet sich kein Teilnehmer.	Überprüfen Sie die Konfiguration des DP-Masters. Überprüfen Sie die Adresseinstellung am EASY204-DP.
Stationsadresse DP-Master	Enthält die Stationsadresse des DP-Masters.	
Statische Diagnose	Die Kommunikation zwischen EASY204-DP und EASY/MFD ist unterbrochen.	Überprüfen Sie die Verbindung zwischen EASY/MFD und EASY204-DP
Status Specifier	Das EASY204-DP gibt keine „Kommend/Gehend“-Meldung zu den gesendeten Diagnose-Informationen (Status): 00 _{hex}	
Status Type	Das EASY204-DP verwendet den Status-Typ „Status-Meldung“: 01 _{hex}	
Sync-Modus aktiv	Der DP-Master hat das synchrone Ausgeben von Datenausgängen an mehrere Teilnehmer aktiviert.	Gewollte Anwender-Aktion
Überlauf zusätzliche Diagnose-Information	Die zusätzlichen Diagnosedaten (Status) sind größer als der dafür reservierte Speicherplatz im DP-Master.	Überprüfen Sie die Konfiguration des DP-Masters

Bezeichnung	Bedeutung	Erläuterung/Abhilfe
Ungültige Antwort DP-Slave	Das EASY204-DP hat eine ungültige Antwort gesendet.	Überprüfen Sie die Verkabelung und die Störsicherheitsmaßnahmen.
Ungültiger Befehl für easy-Betriebsart	In den zyklischen Ausgangsdaten wurde eine unzulässige Bitkombination für die easy-Betriebsart gewählt.	Beachten Sie die Angaben in der Dokumentation und ändern Sie Ihr Anwendungsprogramm entsprechend.
Zusätzliche Diagnose-Informationen	Das EASY204-DP hat zusätzliche Diagnose-Informationen (Status) gesendet.	Sollzustand
Zusätzliche E/A Daten noch nicht verfügbar	Das EASY204-DP befindet sich noch in der Initialisierung und hat noch keine gültigen zusätzlichen Eingangsdaten vom easy800 empfangen.	Temporärer Zustand im Anlauf

GSD-Datei

Für die Auswahl des Gerätes und den Betrieb am Kommunikationsbus PROFIBUS-DP ist eine Gerätestammdatei (GSD-Datei) notwendig. Die GSD-Datei enthält vereinheitlichte PROFIBUS-Teilnehmer-Beschreibungen und ist in diesem Handbuch im Anhang zu finden.

Sie können die Datei „Moe4d10.gsd“ (ab Geräteversion 07) bzw. „Moel4d10.gsd“ (bis Geräteversion 06) unter folgenden Internetadressen erreichen:

- <http://www.eaton.com/moeller> → Support → ...
- <http://www.eaton.com/moeller> → easy Software → Downloadcenter → ...
- <ftp://ftp.moeller.net/EASY/GSD-FILES/>

Folgen Sie den Links auf diesen Seiten.

Anwendermodule

Alle easy600-Funktionen, die über das EASY204-DP verfügbar sind, werden von den Eaton-Steuerungen PS4-341, PS416, XControl und der Siemens-Steuerung S7 unterstützt. Die nachfolgenden SPS-Anwendermodule bieten eine komfortable Möglichkeit, den Datenaustausch zwischen den Steuerrelais EASY und der Mastersteuerung zu gewährleisten.

Folgende Anwendermodule stehen Ihnen zur Verfügung:

Steuerung	Anwendermodul/ Anwendungshinweis	Datei	
		deutsch	englisch
Eaton-Steuerungen			
PS4-341 und PS416			
easy600	S40-AM-K6-D/GB	s40amk6d.exe	s40amk6g.exe
easy800/MFD	AN2700K21D/GB	an27k21d.exe	an27k21g.exe
XC-Steuerungen			
easy600	S40-AM-K6-D/GB	xs-easydp_d.exe	xs-easydp_g.exe
Siemens-Steuerung			
SIMATIC S7-300			
easy600	S7-AM-K6-D/GB	s7amk6d.exe	s7amk6g.exe

Sowohl die in der Tabelle genannten Anwendermodule bzw. Anwendungshinweise, als auch weitere Module zur komfortablen Bereitstellung der easy800- und MFD-Titan-Funktionalität finden Sie unter folgenden Internetadressen:

<ftp://ftp.moeller.net/AUTOMATION>

ftp://ftp.moeller.net/AUTOMATION/APPLICATION_MODULES

ftp://ftp.moeller.net/AUTOMATION/APPLICATION_NOTES

PROFIBUS-Zertifizierung

EASY204-DP wurde von der PROFIBUS-Nutzer-Organisation als PROFIBUS-DP-Slave zertifiziert. EASY204-DP beinhaltet die PROFIBUS-Anschaltung SPC3 (ab Geräteversion 07) bzw. VPC3+ (bis Geräteversion 06)



Beim Betrieb des EASY204-DP bis Geräteversion 04 kann es unter folgenden Umständen zu nonkonformen Verhalten kommen:

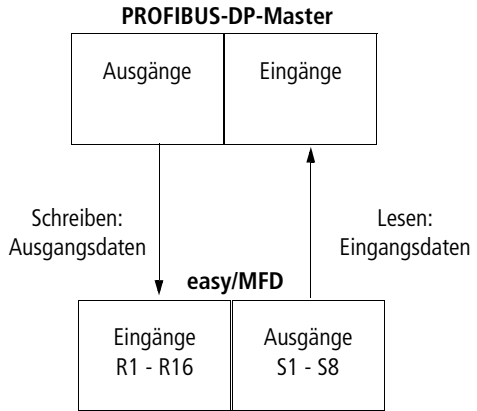
- Wenn in einem Multi-Master-System gleichzeitig DP-Master der Klasse I und II mit Parametrier- oder Konfigurationsdaten auf den Slave zugreifen (sehr geringe Wahrscheinlichkeit).
- Oder wenn andere PROFIBUS-Schicht 2 basierte Master beteiligt sind.

5 Ein-/Ausgänge, Betriebsart easy600/700/800/MFD

Um Ein- oder Ausgangsdaten zwischen dem Slave EASY204-DP und einem PROFIBUS-DP-Master übertragen zu können, müssen Sie das entsprechende Modul in der Slave-Konfiguration wählen.



Die Begriffe „Eingangsdaten“ und „Ausgangsdaten“ werden aus Sicht des PROFIBUS-DP-Masters verwendet.



Modul „Eingaenge 3 Byte“: Betriebsart, S1 - S8

Der normale zyklische Datenaustausch des PROFIBUS-DP-Masters mit dem Slave EASY204-DP wird über die Eingangsdaten-Byte 0, 1, 2 gewährleistet.

Byte	Bedeutung	Wert
0	Betriebsart abfragen	→ Tabelle 3
1	Zustand der easy-Ausgänge S1 bis S8 abfragen	→ Tabelle 4
2	Nicht belegt	00 _{hex}

Voraussetzung:

Das Modul „Eingaenge 3 Byte“ ist ausgewählt.



Die Daten der Ausgänge und Steuerbefehle können Sie nur nutzen, wenn Sie zusätzlich die entsprechenden Module ausgewählt haben.

Der Master liest in den Bytes 0, 1, 2 folgende Daten:

Tabelle 3: Byte 0: Betriebsart

„easy“-Betriebsart	Bit							
	7	6	5	4	3	2	1	0
ohne Eingangsverzögerung	0	0	0	1	0	0	0	0/1
mit Eingangsverzögerung	0	0	1	0	0	0	0	0/1

Beispiel:

Wert 21_{hex}

„easy“/MFD ist im Status „Run“ und arbeitet mit Eingangsverzögerung.

Tabelle 4: Byte 1: Status S1 bis S8 des Basisgerätes

Ausgang	Bit							
	7	6	5	4	3	2	1	0
S1								0/1
S2							0/1	
S3						0/1		
S4					0/1			
S5				0/1				
S6			0/1					
S7		0/1						
S8	0/1							

Beispiel:

Wert 19_{hex} S5, S4 und S1 sind aktiv.



Achtung!

Werden Steuerbefehle mit Ein-/Ausgangsdaten gleichzeitig benutzt, gilt:

- Während der Steuerbefehl ausgeführt wird, bleiben die Eingänge in dem Zustand, bevor der Steuerbefehl aufgerufen wurde.
- Nach Beenden des Datenaustausches „Steuerbefehle“ werden die Bytes der Eingänge wieder aktualisiert.

**Modul „Eingaenge
1 Byte“: S1 – S8**

Bei der Auswahl dieses Moduls erhält der Master lediglich 1 Byte (Spulen-Ausgangsdaten S1 bis S8) über den PROFIBUS.

Byte	Bedeutung	Wert
0	Zustand der easy-Ausgänge S1 bis S8	→ Tabelle 4 auf Seite 49

Voraussetzung:
Das Modul „Eingaenge 1 Byte“ ist ausgewählt.



Die Daten der Ausgänge und Steuerbefehle können Sie nur nutzen, wenn Sie zusätzlich die entsprechenden Module ausgewählt haben.

**Modul „Ausgaenge
3 Byte“: Betriebsart,
R9 - R16, R1 - R8**

Der normale Datenaustausch des PROFIBUS-DP-Masters mit dem Slave EASY204-DP wird über die Ausgangsdaten-Byte 0, 1, 2 gewährleistet.

Byte	Bedeutung	Wert
0	Betriebsart festlegen	→ Tabelle 5
1	Setzen/Rücksetzen der easy/MFD-Eingänge R9 bis R16	→ Tabelle 6
2	Setzen/Rücksetzen der easy/MFD-Eingänge R1 bis R8	→ Tabelle 7

Voraussetzung:
Das Modul „Ausgaenge 3 Byte“ ist ausgewählt.



Die Daten der Ausgänge und Steuerbefehle können Sie nur nutzen, wenn Sie zusätzlich die entsprechenden Module ausgewählt haben.

Der Master liest in den Bytes 0, 1, 2 folgende Daten:

Tabelle 5: Byte 0: Betriebsart

easy-Betriebsart	Bit							
	7	6	5	4	3	2	1	0
Index, um das Basisgerät in den sicheren Zustand zu setzen	0	0	0	0	0	0	0	0
Index zur Übertragung gültiger Daten	0	0	0	1	0	1	0	0
Kommando RUN	0	0	1	1	0	1	0	0
Kommando STOP	0	1	0	0	0	1	0	0

0 = Zustand „0“, 1 = Zustand „1“

Erläuterung

Wert 34_{hex} = 00110100_{bin}:

Dieser Wert setzt den easy/MFD-Status von STOP auf RUN. Er wird nur als Kommando verstanden und lässt somit auch kein zusätzliches Senden von Daten zu. Hierzu muss der Index-Wert 14_{hex} verwendet werden.

Wert 44_{hex} = 01000100_{bin}:

Dieser Wert setzt den easy/MFD-Status von RUN auf STOP. Auch er wird nur als Kommandobefehl verwendet und unterliegt somit derselben Arbeitsweise, wie auch das Kommando RUN.

Bis zur Geräteversion 05 bitte unbedingt beachten!:

Wert 14_{hex} = 00010100_{bin}:

Dieser Wert muss immer im Byte 0 enthalten sein, wenn Daten über das Gateway EASY204-DP zum Basisgerät easy/MFD geschrieben werden sollen.



Auch wenn die Ein-/Ausgänge eines Steuerrelais direkt einem bestimmten Speicherbereich der Master-SPS zugewiesen werden können, muss dennoch das richtige Format für die Datenzusammensetzung (z. B. Eingangsdaten Byte 0 = 14_{hex}) eingehalten werden.

Ab Version 06 ist dies nicht mehr erforderlich.

Tabelle 6: Byte 1: Status von R9 bis R16 schreiben

EASY6... Eingang	Bit							
	7	6	5	4	3	2	1	0
R9								0/1
R10							0/1	
R11						0/1		
R12					0/1			
R13				0/1				
R14			0/1					
R15		0/1						
R16	0/1							

Beispiel:

Wert 19_{hex} R13, R12 und R9 sollen aktiv werden.

Tabelle 7: Byte 2: Status von R1 bis R8 schreiben

EASY6... Eingang	Bit							
	7	6	5	4	3	2	1	0
R1								0/1
R2							0/1	
R3						0/1		
R4					0/1			
R5				0/1				
R6			0/1					
R7		0/1						
R8	0/1							

Beispiel:

Wert 2B_{hex} R6, R4, R2 und R1 sollen aktiv werden.



Achtung!

Werden Steuerbefehle mit Ein-/Ausgangsdaten gleichzeitig benutzt, gilt:

- Während der Steuerbefehl ausgeführt wird, bleiben die Eingänge in dem Zustand, bevor der Steuerbefehl aufgerufen wurde.
- Nach Beenden des Datenaustausches „Steuerbefehle“ werden die Bytes der Ausgänge wieder aktualisiert.

**Modul „Ausgaenge
1 Byte“: R1 – R8**

Bei der Auswahl dieses Moduls sendet der Master lediglich 1 Byte (Spulen-Ausgangsdaten S1 bis S8) über den PROFIBUS.

Byte	Bedeutung	Wert
0	Zustand R1 bis R8	→ Tabelle 7 auf Seite 53

Voraussetzung:

Das Modul „Ausgaenge 1 Byte“ ist ausgewählt.



Die Daten der Eingänge und Steuerbefehle können Sie nur nutzen, wenn Sie zusätzlich die entsprechenden Module ausgewählt haben.

Hinweis zur Verwendung der 1-Byte-Module

Die 1-Byte-Module sind nicht in allen Gerätekombinationen verfügbar. Sollten Probleme in der Handhabung auftreten, prüfen Sie zuerst den GW-Status in der Statusanzeige des Grundgerätes:

GW statisch: der 1-Byte-Modus kann verwendet werden

GW blinkend: Überprüfen Sie die Geräteversionen des EASY204-DP und des Grundgerätes. Sollten diese gültig sein, überprüfen Sie die Konfiguration im PROFIBUS-Netzwerk bzw. dem Konfigurator.

Tabelle 8: Mögliche Gerätekombinationen für das Benutzen des 1-Byte-Modus

Grundgerät	Easy204-DP
Easy600	Version 05 oder 06
Easy700	Version 06
Easy800	
Version \cong 04:	Version 05 oder 06
Version \cong 05:	Version 06
MFD-..-CP8	
Version \cong 02:	Version 05 oder 06
Version \cong 03:	Version 06
MFD-CP10...	
Version \cong 01:	Version 06

Ist der 1-Byte-Modus nicht nutzbar, verwenden Sie bitte den 3-Byte-Modus und achten Sie darauf, dass im Byte 0 der Wert = 0x14 eingetragen ist. Ohne diesen zusätzlichen Eintrag werden durch das Grundgerät keine gültigen Werte erkannt. Sobald das Gerät betriebsbereit ist und die Datenkommunikation läuft, wird dies durch ein statisches GW in der Anzeige des Grundgerätes dargestellt.

6 Zusätzliche Ein-/Ausgangsdaten easy800/MFD

Für die zyklische Übertragung größerer Datenmengen bietet das EASY204-DP ab Geräteversion 07 bei angeschlossenem easy800/MFD Ein-/Ausgangsmodule für 4, 8, 12 und 16 Byte Daten an. Diese Daten sind dem Merkerbereich des easy800/MFD zugeordnet.



Die Übertragung dieser Daten erfolgt Byte-konsistent.

Die nachfolgenden Tabellen zeigen die Adresslage der Ein- und Ausgangsdaten dieser Module auf.

Tabelle 9: Adresslage der Modul-Eingangsdaten in einem DP-Master

		Datenposition	Bezeichnung	
Modul „EASY800 Zusatzzeigänge 16 Byte“	Modul „EASY800 Zusatzzeigänge 12 Byte“	Octet 1	MD63 erstes Byte (low)	
		Octet 2	MD63 zweites Byte	
		Octet 3	MD63 drittes Byte	
		Octet 4	MD63 viertes Byte (high)	
	Modul „EASY800 Zusatzzeigänge 8 Byte“	Modul „EASY800 Zusatzzeigänge 4 Byte“	Octet 5	MD64 erstes Byte (low)
			Octet 6	MD64 zweites Byte
			Octet 7	MD64 drittes Byte
			Octet 8	MD64 viertes Byte (high)
			Octet 9	MD65 erstes Byte (low)
			Octet 10	MD65 zweites Byte
			Octet 11	MD65 drittes Byte
			Octet 12	MD65 viertes Byte (high)
			Octet 13	MD66 erstes Byte (low)
			Octet 14	MD66 zweites Byte
			Octet 15	MD66 drittes Byte
			Octet 16	MD66 viertes Byte (high)

Tabelle 10: Adresslage der Modul-Ausgangsdaten in einem DP-Master

			Datenposition	Bezeichnung
Modul „EASY800 Zusatzeingänge 16 Byte“	Modul „EASY800 Zusatzeingänge 12 Byte“	Modul „EASY800 Zusatzeingänge 8 Byte“	Octet 1	MD59 erstes Byte (low)
			Octet 2	MD59 zweites Byte
			Octet 3	MD59 drittes Byte
			Octet 4	MD59 viertes Byte (high)
		Modul „EASY800 Zusatzeingänge 4 Byte“	Octet 5	MD60 erstes Byte (low)
			Octet 6	MD60 zweites Byte
			Octet 7	MD60 drittes Byte
			Octet 8	MD60 viertes Byte (high)
			Octet 9	MD61 erstes Byte (low)
			Octet 10	MD61 zweites Byte
			Octet 11	MD61 drittes Byte
			Octet 12	MD61 viertes Byte (high)
			Octet 13	MD62 erstes Byte (low)
			Octet 14	MD62 zweites Byte
			Octet 15	MD62 drittes Byte
			Octet 16	MD62 viertes Byte (high)

7 Steuerbefehle für easy600 (DPV0)

Datenaustauschverfahren

Das Modul „Steuerbefehle 7 Byte“ erlaubt den erweiterten Datenaustausch des easy600 am Kommunikationsbus PROFIBUS-DP. Damit können Sie Dienste aus folgenden Bereichen übertragen:

- „Datum und Uhrzeit, Sommer-/Winterzeit“,
- „Funktionsbausteine lesen/schreiben“ und
- „Abbild lesen/schreiben“.

Damit die Daten sicher vom Master zum Slave und umgekehrt über PROFIBUS-DP übertragen werden, ist ein Datenaustauschverfahren notwendig.

Über einen speziellen Befehlscode im Byte 0 werden die entsprechenden Dienste aufgerufen. In den nachfolgenden Datenbytes 1 bis 6 können die entsprechenden Werte geschrieben oder gelesen werden.



Achtung!

Während der Benutzung eines Steuerbefehls behalten die Ein- und Ausgangsdaten den Zustand vor Aufruf des Steuerbefehls. Erst wenn der Datenaustausch des Steuerbefehls beendet ist, sind die Ein-/Ausgangsdaten wieder aktuell.



Warnung!

Es dürfen nur die für den Befehlscode angegebenen Werte verwendet werden.

Überprüfen Sie die Werte, die Sie schreiben, um Fehlfunktionen zu vermeiden.

Voraussetzung:

Das entsprechende Modul „Steuerefehle 7 Byte“ wurde gewählt.



Daten können nur geschrieben werden, wenn das „easy“-Basisgerät mit LCD-Anzeige die Statusanzeige anzeigt.

Der Master löst den Datenaustausch der Steuerbefehle aus und der angesprochene Slave antwortet.

Bei der Kommunikation werden jeweils 7 Datenbytes auf dem PROFIBUS übergeben.

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
--------	--------	--------	--------	--------	--------	--------

Byte 0, Toggelbyte

Das Byte 0 beinhaltet das Toggelbit und den Befehl. Es dient dazu, das Senden eines Steuerbefehls zu aktivieren.

Byte 0, Toggelbyte								
Bit	7	6	5	4	3	2	1	0
	0/1	x	x	x	x	x	x	x
	Toggelbit	Befehl/Kommando						

Vorgehensweise

- ▶ Um einen Befehl abzusetzen, wird das Bit 7 getoggelt; d. h. von 1 auf 0 oder von 0 auf 1 gesetzt.
- ▶ Pollen Sie anschließend so lange auf das Togglebit der Antwort des Koppelmoduls, bis es den gleichen Zustand wie das gesendete Togglebit angenommen hat. Dieser Zustand zeigt der Mastersteuerung an, dass die Antwort auf den abgesetzten Befehl gültig ist.
- ▶ Senden Sie einen neuen Befehl erst ab, wenn Sie eine Antwort erhalten haben (Ändern des Togglebits); sonst wird eventuell die Antwort des vorherigen Befehls überschrieben, bevor diese gelesen werden kann.



Wenn Sie Ein-/Ausgangsdaten sowie Steuerbefehle gleichzeitig benutzen, gilt:

Wenn der Datenaustausch von Steuerbefehlen beendet ist, werden die Bytes der Ein-Ausgänge wieder aktualisiert.

Datum und Uhrzeit, Sommer-/Winterzeit		Telegrammaufbau											
Byte	Bedeutung	Wert (hex), sendet		Bit									
		Master	Slave	7	6	5	4	3	2	1	0		
0	Befehl ¹⁾												
	Lesen	3C	–	0/1	0	1	1	1	1	1	0	0	
	Schreiben	2A	–	1/0	0	1	0	1	0	1	0		
	Antwort ¹⁾												
	Lesen erfolgreich	–	C2/42	1/0	1	0	0	0	0	0	1	0	
	Schreiben erfolgreich	–	C1/41	1/0	1	0	0	0	0	0	0	1	
	Befehl zurückgewiesen	–	C0/40	1/0	1	0	0	0	0	0	0	0	
1	Wochentag												
	Beim Lesen	00	→ Tabelle 11										
	Beim Schreiben	→ Tabelle 11	00										
2	Stunde												
	Beim Lesen	00	→ Tabelle 12										
	Beim Schreiben	→ Tabelle 12	00										
3	Minute												
	Beim Lesen	00	→ Tabelle 13										
	Beim Schreiben	→ Tabelle 13	00										
4	Sommer-/Winterzeit												
	Beim Lesen	00	→ Tabelle 14										
	Beim Schreiben	→ Tabelle 14	00										

1) Beachten Sie das Datenaustauschverfahren Byte 0 Bit 7, → Seite 59.

Tabelle 11: Byte 1: Wochentag (Wertebereich 00 bis 06)

Wochentag	Bit							
	7	6	5	4	3	2	1	0
Montag = 0	0	0	0	0	0	0	0	0
Dienstag = 1	0	0	0	0	0	0	0	1
Mittwoch = 2	0	0	0	0	0	0	1	0
Donnerstag = 3	0	0	0	0	0	0	1	1
Freitag = 4	0	0	0	0	0	1	0	0
Samstag = 5	0	0	0	0	0	1	0	1
Sonntag = 6	0	0	0	0	0	1	1	0

Tabelle 12: Byte 2: Stunde (Wertebereich 00 bis 23)

Wert (bcd)	Wert 10 Bit				Wert 1 Bit			
	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1
...								
9	0	0	0	0	1	0	0	1
...								
14	0	0	0	1	0	1	0	0
...								
23	0	0	1	0	0	0	1	1

Tabella 13: Byte 3: Minute (Wertebereich 00 bis 59)

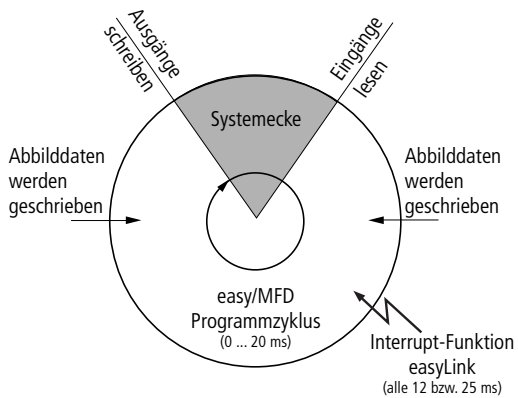
Wert (bcd)	Wert 10				Wert 1			
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
00	0	0	0	0	0	0	0	0
...								
10	0	0	0	1	0	0	0	0
...								
21	0	0	1	0	0	0	0	1
...								
42	0	1	0	0	0	0	1	0
...								
59	0	1	0	1	1	0	0	1

Tabella 14: Byte 4: Winter-/Sommerzeit (Wertebereich 00 bis 01)

Wert (bcd)	Wert 10				Wert 1			
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Funktion								
Winterzeit	0	0	0	0	0	0	0	0
Sommerzeit	0	0	0	0	0	0	0	1

Abbild lesen/schreiben

Generelles zum Arbeiten mit Abbilddaten



Beim Schreiben auf die Abbilddaten muss beachtet werden, dass ein im easy800/MFD-Programm verwendetes Abbild (z. B. Eingänge, Ausgänge, ...) ebenfalls durch das eigentliche Programm zyklisch beschrieben wird. Es bleiben lediglich die Abbilddaten unverändert, die im Programm nicht verwendet und somit im Programmzyklus nicht beschrieben werden. Aus dieser Arbeitsweise geht auch hervor, dass ein über den easyLink beschriebenes Abbild, z. B. Ausgangsdaten nur dann an die physikalischen Ausgänge des easy800/MFD ausgegeben werden, wenn das Steuerrelais im „Run“-Modus arbeitet.

Übersicht

Operanden	Bedeutung	Lesen/schreiben	Befehl	Seite
M1 - M16, Q1 - Q8, D1 - D8	„Zustand Hilfsrelais, digitale Ausgänge und Textanzeige lesen“	lesen	40	65
P1 - P4, ESC/OK/DEL/ALT	„Zustand P-Tasten und Bedientasten lesen“	lesen	3E	68
T1 - T8, C1 - C8, Q1 - Q4, A1 - A8,	„Zustand Zeitrelais, Zählerrelais, Zeitschaltuhren und Analogwertvergleichler lesen“	lesen	3F	69

Zustand Hilfsrelais, digitale Ausgänge und Textanzeige lesen

Mit dem folgenden Befehl wird der logische Status aller Hilfsrelais (Merker) M1 bis M16, digitale Ausgänge Q1 bis Q7, Textmerker D1 bis D8 ausgelesen.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet		Bit									
		Master	Slave	7	6	5	4	3	2	1	0		
0	Befehl ¹⁾												
	Lesen	40	–	1/0	1	0	0	0	0	0	0	0	0
	Antwort ¹⁾												
	Lesen erfolgreich	–	C2/42	1/0	1	0	0	0	0	0	1	0	
	Befehl zurückgewiesen	–	C0/40	1/0	1	0	0	0	0	0	0	0	
1	Zustand Hilfsrelais (Merker) M1 bis M8	00	→ Tabelle 15										
2	Zustand Hilfsrelais (Merker) M9 bis M16	00	→ Tabelle 16										
3	Zustand digitale Ausgänge Q1 bis Q8	00	→ Tabelle 17										
4	Zustand Textmerker D1 bis D8	00	→ Tabelle 18										

1) Beachten Sie das Datenaustauschverfahren Byte 0 Bit 7, → Seite 59.

Tabelle 15: Byte 1: Zustand der Hilfsrelais/Merker 1 bis 8

	Bit							
	7	6	5	4	3	2	1	0
M1								0/1
M2							0/1	
M3						0/1		
M4					0/1			
M5				0/1				
M6			0/1					
M7		0/1						
M8	0/1							

Tabelle 16: Byte 2: Zustand der Hilfsrelais/Merker 9 bis 16

	Bit							
	7	6	5	4	3	2	1	0
M9								0/1
M10							0/1	
M11						0/1		
M12					0/1			
M13				0/1				
M14			0/1					
M15		0/1						
M16	0/1							

Tabelle 17: Byte 3: Zustand der digitalen Ausgänge Q1 bis Q8

	Bit							
	7	6	5	4	3	2	1	0
Q1								0/1
Q2							0/1	
Q3						0/1		
Q4					0/1			
Q5				0/1				
Q6			0/1					
Q7		0/1						
Q8	0/1							

Tabelle 18: Byte 4: Zustand der Textmerker D1 bis D8

	Bit							
	7	6	5	4	3	2	1	0
D1								0/1
D2							0/1	
D3						0/1		
D4					0/1			
D5				0/1				
D6			0/1					
D7		0/1						
D8	0/1							

Zustand P-Tasten und Bedientasten lesen

Mit dem folgenden Befehl wird der logische Zustand der digitalen Tastereingänge P1 bis P4 ausgelesen.

Der Zustand der Tasten wird nur angezeigt, wenn

- eine P-Taste im Schaltplan verwendet ist und
- die Tasten am Gerät aktiviert sind.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet		Bit									
		Master	Slave	7	6	5	4	3	2	1	0		
0	Befehl ¹⁾												
	Lesen	3E	–	1/0	0	1	1	1	1	1	1	0	
	Antwort ¹⁾												
	Lesen erfolgreich	–	C2/42	1/0	1	0	0	0	0	0	1	0	
	Befehl zurückgewiesen	–	C0/40	1/0	1	0	0	0	0	0	0	0	
1	Zustand Zeitrelais	00	→ Tabelle 19										

1) Beachten Sie das Datenaustauschverfahren Byte 0 Bit 7, → Seite 59.

Tabelle 19: Byte 1: Zustand der Tasten

Bedeutung	Bit							
	7	6	5	4	3	2	1	0
Status P1								0/1
Status P2							0/1	
Status P3						0/1		
Status P4					0/1			
ESC unbetätigt/betätigt				0/1				
OK unbetätigt/betätigt			0/1					
DEL unbetätigt/betätigt		0/1						
ALT unbetätigt/betätigt	0/1							

Zustand Zeitrelais, Zählerrelais, Zeitschaltuhren und Analogwertvergleichler lesen

Mit dem folgenden Befehl wird der logische Status aller Zeitrelais, Zähler, Zeitschaltuhren und Analogwertvergleichlern ausgelesen.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet		Bit									
		Master	Slave	7	6	5	4	3	2	1	0		
0	Befehl ¹⁾												
	Lesen	3F	–	1/0	0	1	1	1	1	1	1	1	1
	Antwort ¹⁾												
	Lesen erfolgreich	–	C2/42	1/0	1	0	0	0	0	0	1	0	0
	Befehl zurückgewiesen	–	C0/40	1/0	1	0	0	0	0	0	0	0	0
1	Zustand Zeitrelais	00	→ Tabelle 20										
2	Zustand Zählerrelais	00	→ Tabelle 21										
3	Zustand Zeitschaltuhr	00	→ Tabelle 22										
4	Zustand Analogwertvergleichler	00	→ Tabelle 23										

1) Beachten Sie das Datenaustauschverfahren Byte 0 Bit 7, → Seite 59.

Tabelle 20: Byte 1: Zustand der Zeitrelais

	Bit							
	7	6	5	4	3	2	1	0
T1								0/1
T2							0/1	
T3						0/1		
T4					0/1			
T5				0/1				
T6			0/1					
T7		0/1						
T8	0/1							

Tabelle 21: Byte 2: Zustand der Zählerrelais

	Bit							
	7	6	5	4	3	2	1	0
C1								0/1
C2							0/1	
C3						0/1		
C4					0/1			
C5				0/1				
C6			0/1					
C7		0/1						
C8	0/1							

Tabelle 22: Byte 3: Zustand der Zeitschaltuhren

	Bit							
	7	6	5	4	3	2	1	0
Q1								0/1
Q2							0/1	
Q3						0/1		
Q4					0/1			
				0				
			0					
		0						
	0							

Tabelle 23: Byte 4: Zustand der Analogwertvergleichler

	Bit							
	7	6	5	4	3	2	1	0
A1								0/1
A2							0/1	
A3						0/1		
A4					0/1			
A5				0/1				
A6			0/1					
A7		0/1						
A8	0/1							

Funktionsbausteine **Übersicht**
lesen/schreiben

Operanden	Bedeutung	Befehl	Seite
A1 - A8	„Analogwertvergleichler schreiben (Funktion, Vergleichswerte)“	22 _{hex} - 29 _{hex}	73
C1 - C8	„Zählerrelais Istwert lesen“	49 _{hex} - 50 _{hex}	76
	„Zählerrelais Sollwert schreiben“	09 _{hex} - 10 _{hex}	78
I1 - I16	„Analog- und Digital-Eingänge lesen (I7, I8, I1 bis I16)“	3D _{hex}	81
T1 - T8	„Zeitrelais Istwert lesen (Zeitbereich, Istwert, Schaltfunktion)“	41 _{hex} - 48 _{hex}	84
	„Zeitrelais Sollwert schreiben (Zeitbereich, Sollwert, Schaltfunktion)“	01 _{hex} - 08 _{hex}	87
Q1 - Q4	„Zeitschaltuhr lesen (Kanal, ON-Zeit, OFF-Zeit)“	2B _{hex} - 3A _{hex}	94
	„Zeitschaltuhr schreiben (Kanal, ON-Zeit, OFF-Zeit)“	12 _{hex} - 21 _{hex}	98

**Analogwertvergleichers schreiben
(Funktion, Vergleichswerte)**

Byte	Bedeutung	Wert (hex), sendet		Bit									
		Master	Slave	7	6	5	4	3	2	1	0		
0	Befehl ¹⁾												
	A1	22	–	0	0	1	0	0	0	1	0		
	A2	23	–	0	0	1	0	0	0	1	1		
	A3	24	–	0	0	1	0	0	1	0	0		
	A4	25	–	0	0	1	0	0	1	0	1		
	A5	26	–	0	0	1	0	0	1	1	0		
	A6	27	–	0	0	1	0	0	1	1	1		
	A7	28	–	0	0	1	0	1	0	0	0		
	A8	29	–	0	0	1	0	1	0	0	1		
	Antwort ¹⁾												
	Schreiben erfolgreich	–	C1/41	1/0	1	0	0	0	0	0	1		
	Befehl zurückgewiesen	–	C0/40	1/0	1	0	0	0	0	0	0		
1	Vergleicher	→ Tabelle 39	ungültig										
2	Vergleichswert bei Vergleich mit Konstante	→ Tabelle 40											

1) Beachten Sie das Datenaustauschverfahren Byte 0 Bit 7, → Seite 59.

Die Vergleichswerte sowie die Funktion sind Bestandteil einer „*.eas-Datei“. Werden diese Werte verändert, stimmt die Original „*.eas-Datei“ mit der in EASY6... befindlichen nicht mehr überein.

Beachten Sie diese Eigenschaft beim Hoch- und Runterladen und Vergleich von „easy“-Schaltplänen mit der easySoft.

Beim Runterladen vom PC wird der aktuelle Stand der „*.eas“ überschrieben.

Beim Vergleich sind die Schaltpläne ungleich.

Tabelle 24: Byte 1: Steuerbyte Analogwertvergleicher: Vergleicher

Bedeutung	Bit							
	7	6	5	4	3	2	1	0
Vergleich auf „ \geq “								0
Vergleich auf „ \leq “								1
I7 zu I8						0	0	
I7 zu Konstante						0	1	
I8 zu Konstante						1	0	
Fest			0	0	0			
Erscheint nicht im Parametermenü		1						
Erscheint im Parametermenü		0						
Bearbeiten	1							

Tabelle 25: Byte 2: Vergleichswert bei Vergleich mit Konstante

Wert (hex)	Bit							
	7	6	5	4	3	2	1	0
00	0	0	0	0	0	0	0	0
63	0	1	1	0	0	0	1	1

Beispiel

Der Analogwertvergleicher A8 besitzt folgende Eigenschaften:

- Vergleich $I7 < 4,7\text{ V}$

Der Master erteilt den Befehl, dass der Vergleichswert auf 4,2 V gesenkt werden soll.

Byte	Bedeutung	Wert (hex)	Bit							
			7	6	5	4	3	2	1	0
0	Befehl: A8	29	0	0	1	0	1	0	0	1
	Antwort: Schreiben erfolgreich	–	0	1	0	0	0	0	0	1
1	Vergleicher	→	1	0	0	0	0	0	1	1
2	Vergleichswert bei Vergleich mit Konstante	2A	0	0	1	0	1	0	1	0

Der Slave antwortet mit folgendem Telegramm:

Byte	Bedeutung	Wert (hex)	Bit							
			7	6	5	4	3	2	1	0
0	Antwort: Schreiben erfolgreich	41	0	1	0	0	0	0	0	1
1	Vergleicher	ungültig								
2	Vergleichswert bei Vergleich mit Konstante	00								

Zählerrelais Istwert lesen

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet		Bit										
		Master	Slave	7	6	5	4	3	2	1	0			
0	Befehl ¹⁾													
	C1	49	–	1/0	1	0	0	1	0	0	1			
	C2	4A	–	1/0	1	0	0	1	0	1	0			
	C3	4B	–	1/0	1	0	0	1	0	1	1			
	C4	4C	–	1/0	1	0	0	1	1	0	0			
	C5	4D	–	1/0	1	0	0	1	1	0	1			
	C6	4E	–	1/0	1	0	0	1	1	1	0			
	C7	4F	–	1/0	1	0	0	1	1	1	1			
	C8	50	–	1/0	1	0	1	0	0	0	0			
	Antwort ¹⁾													
	Lesen erfolgreich	–	C2/42	1/0	1	0	0	0	0	1	0			
	Befehl zurückgewiesen	–	C0/40	1/0	1	0	0	0	0	0	0			
1	ungültig	00	→	x	x	x	x	x	x	x	x	x	x	x
2	Zählerrelais-Istwert (Low Byte)	00	→ Tabelle 26											
3	Zählerrelais-Istwert (High Byte)	00	→ Tabelle 27											

1) Beachten Sie das Datenaustauschverfahren Byte 0 Bit 7, → Seite 59.

Tabelle 26: Byte 2: Zählerrelais Istwert (Low-Byte)

Wert (hex)	Bit							
	7	6	5	4	3	2	1	0
00	0	0	0	0	0	0	0	0
FF	1	1	1	1	1	1	1	1

Tabelle 27: Byte 3: Zähler Istwert (High-Byte)

Wert (hex)	Bit							
	7	6	5	4	3	2	1	0
00	0	0	0	0	0	0	0	0
FF	1	1	1	1	1	1	1	1

Beispiel

Der Master erteilt den Befehl zum Lesen des Zählers C5:

Byte	Bedeutung	Wert (hex)	Bit							
			7	6	5	4	3	2	1	0
0	Befehl: C5	4D	1	1	0	0	1	1	0	1
1 - 6		00								

Der Slave antwortet mit folgenden Werten:

Byte	Bedeutung	Wert (hex)	Bit							
			7	6	5	4	3	2	1	0
0	Antwort: Lesen erfolgreich	42	1	1	0	0	0	0	1	0
1	ungültig		x	x	x	x	x	x	x	x
2	Zählerrelais-Istwert (Low Byte)	67	0	1	1	0	0	1	1	1
3	Zählerrelais-Istwert (High Byte)	12	0	0	0	1	0	0	1	0

Istwert des Zählers C5: 4711 (Wert $1267_{hex} = 4711_{dez}$)

Zählerrelais Sollwert schreiben

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet		Bit										
		Master	Slave	7	6	5	4	3	2	1	0			
0	Befehl ¹⁾													
	C1	09	–	1/0	0	0	0	1	0	0	1			
	C2	0A	–	1/0	0	0	0	1	0	1	0			
	C3	0B	–	1/0	0	0	0	1	0	1	1			
	C4	0C	–	1/0	0	0	0	1	1	0	0			
	C5	0D	–	1/0	0	0	0	1	1	0	1			
	C6	0E	–	1/0	0	0	0	1	1	1	0			
	C7	0F	–	1/0	0	0	0	1	1	1	1			
	C8	10	–	1/0	0	0	1	0	0	0	0			
	Antwort ¹⁾													
	Schreiben erfolgreich	–	C1/41	1/0	1	0	0	0	0	0	1			
	Befehl zurückgewiesen	–	C0/40	1/0	1	0	0	0	0	0	0			
1	Parametermenü	→ Tabelle 28	00											
2	Sollwert (Low Byte)	→ Tabelle 29	00											
3	Sollwert (High Byte)	→ Tabelle 30	00											

1) Beachten Sie das Datenaustauschverfahren Byte 0 Bit 7, → Seite 59.

Wertebereich der Zählerwerte: 0000 bis 9999



Halten Sie den Wertebereich ein.

Der Wert ist Bestandteil einer „*.eas-Datei“. Werden diese Werte verändert, stimmt die Original „*.eas-Datei“ mit der in EASY6... befindlichen nicht mehr überein.

Beachten Sie diese Eigenschaft beim Hoch- und Runterladen und Vergleich von „easy“-Schaltplänen mit der easySoft.

Beim Runterladen vom PC wird der aktuelle Stand der „*.eas“ überschrieben.

Beim Vergleich sind die Schaltpläne ungleich.

Tabelle 28: Byte 1: Steuerbyte Zählerrelais

Bedeutung	Bit							
	7	6	5	4	3	2	1	0
Nicht belegt			0	0	0	0	0	0
Erscheint nicht im Parametermenü		1						
Erscheint im Parametermenü		0						
Bearbeiten	1							

Tabelle 29: Byte 2: Zählerwert (Low-Byte)

Wert (hex)	Bit							
	7	6	5	4	3	2	1	0
00	0	0	0	0	0	0	0	0
FF	1	1	1	1	1	1	1	1

Tabelle 30: Byte 3: Zählerwert (High-Byte)

Wert (hex)	Bit							
	7	6	5	4	3	2	1	0
00	0	0	0	0	0	0	0	0
FF	1	1	1	1	1	1	1	1

Beispiel: Zählerrelais Sollwert verändern

Der Master erteilt den Befehl, den Sollwert des Zählerrelais C8 auf 1542 zu ändern:

Byte	Bedeutung	Wert (hex)	Bit							
			7	6	5	4	3	2	1	0
0	Befehl: C8	10	0	0	0	1	0	0	0	0
1	erscheint im Parametermenü	-> Bit-Wert	1	0	0	0	0	0	0	0
2	Sollwert (Low Byte)	06	0	0	0	0	0	1	1	0
3	Sollwert (High Byte)	06	0	0	0	0	0	1	1	0

$0606_{\text{hex}} = 1542_{\text{dez}}$

Der Slave antwortet mit folgendem Telegramm:

Byte	Bedeutung	Slave	Bit							
			7	6	5	4	3	2	1	0
0	Antwort: Schreiben erfolgreich	41	0	1	0	0	0	0	0	1
1 - 3		00								

Analog- und Digital-Eingänge lesen (I7, I8, I1 bis I16)

Mit dem folgenden Befehl werden die Werte der beiden Analog-Eingänge I7, I8 (nur EASY...-DC-...) sowie die logischen Zustände der digitalen Eingänge I1 bis I16 gelesen.

Byte	Bedeutung	Wert (hex), sendet		Bit									
		Master	Slave	7	6	5	4	3	2	1	0		
0	Befehl ¹⁾												
	Lesen	3D	–	0	0	1	1	1	1	0	1		
	Antwort ¹⁾												
	Lesen erfolgreich	–	C2/42	1/0	1	0	0	0	0	0	1	0	
	Befehl zurückgewiesen	–	C0/40	1/0	1	0	0	0	0	0	0	0	
1	Analogwert von I7	00	→ Tabelle 31										
2	Analogwert von I8	00	→ Tabelle 32										
3	Zustand der Eingänge I1 bis I8	00	→ Tabelle 33										
4	Zustand der Eingänge I9 bis I12, I15, I16	00	→ Tabelle 34										

1) Beachten Sie das Datenaustauschverfahren Byte 0 Bit 7, → Seite 59.

Tabelle 31: Byte 1: Analogwert I7

Analogwert I7 (hex)	Bit							
	7	6	5	4	3	2	1	0
00	0	0	0	0	0	0	0	0
...
64	0	1	1	0	0	1	0	0

Beispiele: 0A_{hex} = 1 V, 32_{hex} = 5 V, 64_{hex} = 10 V

Tabelle 32: Byte 2: Analogwert I8

Analogwert I8 (hex)	Bit							
	7	6	5	4	3	2	1	0
00 _{hex}	0	0	0	0	0	0	0	0
...
64 _{hex}	0	1	1	0	0	1	0	0

Beispiele: 0A_{hex} = 1 V, 3C_{hex} = 6 V, 5A_{hex} = 9 V

Tabelle 33: Byte 3: Status Eingänge I1 bis I8

Wert	Bit							
	7	6	5	4	3	2	1	0
I1								0/1
I2							0/1	
I3						0/1		
I4					0/1			
I5				0/1				
I6			0/1					
I7		0/1						
I8	0/1							

Wert 0 = ausgeschaltet, Wert 1 = eingeschaltet

Tabelle 34: Byte 4: Status Eingänge I9 bis I12, I15, I16

Wert	Bit							
	7	6	5	4	3	2	1	0
I9								0/1
I10							0/1	
I11						0/1		
I12					0/1			
I13				0				
I14			0					
I15		0/1						
I16	0/1							

Wert 0 = ausgeschaltet, Wert 1 = eingeschaltet



I13 = 0, I14 = 0

Wenn I14 = 1, hat sich EASY204-DP vom Basisgerät abgekoppelt.

I15, I16 sind die Kurzschlussmelder bei den EASY...-DC-... Transistorvarianten.

**Zeitrelais Istwert lesen
(Zeitbereich, Istwert, Schaltfunktion)**

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet		Bit									
		Master	Slave	7	6	5	4	3	2	1	0		
0	Befehl ¹⁾												
	T1	41	–	1/0	1	0	0	0	0	0	0	1	
	T2	42	–	1/0	1	0	0	0	0	0	1	0	
	T3	43	–	1/0	1	0	0	0	0	0	1	1	
	T4	44	–	1/0	1	0	0	0	0	1	0	0	
	T5	45	–	1/0	1	0	0	0	0	1	0	1	
	T6	46	–	1/0	1	0	0	0	0	1	1	0	
	T7	47	–	1/0	1	0	0	0	0	1	1	1	
	T8	48	–	1/0	1	0	0	1	0	0	0	0	
		Antwort ¹⁾											
	Lesen erfolgreich	–	C2/42	1/0	1	0	0	0	0	0	1	0	
	Befehl zurückgewiesen	–	C0/40	1/0	1	0	0	0	0	0	0	0	
1	Zeitrelaisfunktion, Zeitbereich, Steuerzustand	00	→ Tabelle 35										
2	Zeit-Istwert (Low Byte)	00	→ Tabelle 36										
3	Zeit-Istwert (High Byte)	00	→ Tabelle 37										

1) Beachten Sie das Datenaustauschverfahren Byte 0 Bit 7, → Seite 59.

Tabelle 35: Byte 1: Zeitrelaisfunktion, Zeitbereich, Steuerzustand

Bedeutung	Bit							
	7	6	5	4	3	2	1	0
Ansprechverzögert						0	0	0
Rückfallverzögert						0	0	1
Ansprechverzögert mit Zufallsschalten						0	1	0
Rückfallverzögert mit Zufallsschalten						0	1	1
Impulsformend						1	0	0
Blinkend						1	0	1
Zeitbasis „S“				0	0			
Zeitbasis „M:S“				0	1			
Zeitbasis „H:M“				1	0			
Nicht belegt			0					
Erscheint im Parametermenü		0						
Erscheint nicht im Parametermenü		1						
Zeitrelais nicht in Bearbeitung vom Betriebssystem	0							
Zeitrelais in Bearbeitung vom Betriebssystem	1							

Tabelle 36: Byte 2: Zeit-Istwert (Low-Byte)

Wert (hex)	Bit							
	7	6	5	4	3	2	1	0
00	0	0	0	0	0	0	0	0
FF	1	1	1	1	1	1	1	1

Tabelle 37: Byte 3: Zeitistwert (High-Byte)

Wert (hex)	Bit							
	7	6	5	4	3	2	1	0
00 _{hex}	0	0	0	0	0	0	0	0
FF _{hex}	1	1	1	1	1	1	1	1

Beispiel

Der Master erteilt den Befehl zum Lesen des Zeitrelais T1:

Byte	Bedeutung	Wert (hex)	Bit								
			7	6	5	4	3	2	1	0	
0	Befehl: T1	41	1	1	0	0	0	0	0	0	1
1 - 3		00									

Der Slave antwortet mit folgenden Werten:

Byte	Bedeutung	Wert (hex)	Bit								
			7	6	5	4	3	2	1	0	
0	Antwort: Lesen erfolgreich	C2	1	1	0	0	0	0	0	1	0
1	Triggerspule angesteuert, Zeitbasis „M:S“, ansprechverzögert, Parameteranzeige +	→	1	0	0	0	1	0	0	0	0
2	Zeit-Istwert (Low Byte)	10	0	0	0	1	0	0	0	0	0
3	Zeit-Istwert (High Byte)	0E	0	0	0	0	1	1	1	0	0

Wert Sollzeit = 0E10_{hex} = 3600

3600 s = 60:00 M:S

Zeitrelais Sollwert schreiben
(Zeitbereich, Sollwert, Schaltfunktion)

Byte	Bedeutung	Wert (hex), sendet		Bit									
		Master	Slave	7	6	5	4	3	2	1	0		
0	Befehl ¹⁾												
	T1	01	–	1/0	0	0	0	0	0	0	0	1	
	T2	02	–	1/0	0	0	0	0	0	0	1	0	
	T3	03	–	1/0	0	0	0	0	0	0	1	1	
	T4	04	–	1/0	0	0	0	0	0	1	0	0	
	T5	05	–	1/0	0	0	0	0	0	1	0	1	
	T6	06	–	1/0	0	0	0	0	0	1	1	0	
	T7	07	–	1/0	0	0	0	0	0	1	1	1	
	T8	08	–	1/0	0	0	0	0	1	0	0	0	
		Antwort ¹⁾											
	Schreiben erfolgreich	–	C1/41	1/0	1	0	0	0	0	0	1		
	Befehl zurückgewiesen	–	C0/40	1/0	1	0	0	0	0	0	0		
1	Zeitrelaisfunktion, Zeitbasis, Parametermenü	→ Tabelle 38	ungültig										
2	Zeitwert „--.xx“ bei Zeitbasis „S“	→ Tabelle 39											
3	Zeitwert „xx.--“ bei Zeitbasis „S“ oder „--:xx“ bei „M:S“	→ Tabelle 40											
4	Zeitwert „xx.--“ bei Zeitbasis „M:S“ oder „--:xx“ bei „H:M“	→ Tabelle 41											
5	Zeitwert „xx.--“ bei Zeitbasis „H:M“	→ Tabelle 42											
6	Stundenwert in Tage	00	→ Tabelle 43										

1) Beachten Sie das Datenaustauschverfahren Byte 0 Bit 7, → Seite 59.



Zeitwerte über 60 s werden in Minuten umgewandelt.
Zeitwerte über 60 Min. werden in Stunden umgewandelt.
Zeitwerte über 24 h werden in Tage umgewandelt.

Der Wertebereich der Zeitwerte und der Sollwert des Zeitrelais sind Bestandteil einer „*.eas-Datei“. Werden diese Werte verändert, stimmt die Original „*.eas-Datei“ mit der in EASY6... befindlichen nicht mehr überein.

Beachten Sie diese Eigenschaft beim Hoch- und beim Runterladen und beim Vergleich von „easy“-Schaltplänen mit der easySoft.

Beim Runterladen vom PC wird der aktuelle Stand der „*.eas“ überschrieben.

Beim Vergleich sind die Schaltpläne ungleich.

Wertebereich der Zeitwerte

- „S“ 00.00 bis 99.99
- „M:S“ 00:00 bis 99:59 (M = 00 bis 99, S = 00 bis 59)
- „H:M“ 00:00 bis 99:59 (H = 00 bis 99, M = 00 bis 59)



Es dürfen nur die Bytes benutzt werden, die für die Zeitbasis gelten.

Tabelle 38: Byte 1: Zeitrelais Steuerbyte

Bedeutung	Bit							
	7	6	5	4	3	2	1	0
Ansprechverzögert						0	0	0
Rückfallverzögert						0	0	1
Ansprechverzögert mit Zufallsschalten						0	1	0
Rückfallverzögert mit Zufallsschalten						0	1	1
Impulsformend						1	0	0
Blinkend						1	0	1
Zeitbasis „S“				0	0			
Zeitbasis „M:S“				0	1			
Zeitbasis „H:M“				1	0			
Nicht belegt			0					
Erscheint nicht im Parametermenü		1						
Erscheint im Parametermenü		0						
Bearbeiten	1							

Tabelle 39: Byte 2: Zeitwert „--.xx“ Zeitbasis „S“

Wert (bcd)	Wert --.x-				Wert --.-x			
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
00	0	0	0	0	0	0	0	0
05	0	0	0	0	0	1	0	1
17	0	0	0	1	0	1	1	1
42	0	1	0	0	0	0	1	0
99	1	0	0	1	1	0	0	1

Tabelle 40: Byte 3: Zeitwert „xx.--“ Zeitbasis „S“
Zeitwert „--:xx“ Zeitbasis „M:S“

Wert (bcd)	Wert x.-- „S“				Wert -x.-- „S“			
	Wert --:x- „:S“				Wert --:x- „:S“			
	Bit				Bit			
	7	6	5	4	3	2	1	0
00	0	0	0	0	0	0	0	0
05	0	0	0	0	0	1	0	1
17	0	0	0	1	0	1	1	1
42	0	1	0	0	0	0	1	0
99	1	0	0	1	1	0	0	1

Tabelle 41: Byte 4: Zeitwert „xx.--“ Zeitbasis „M:S“
Zeitwert „--:xx“ Zeitbasis „H:M“

Wert (bcd)	Wert x.-- „M:-“				Wert -x.-- „M:-“			
	Wert --:x- „:M“				Wert --:x- „:M“			
	Bit				Bit			
	7	6	5	4	3	2	1	0
00	0	0	0	0	0	0	0	0
05	0	0	0	0	0	1	0	1
17	0	0	0	1	0	1	1	1
42	0	1	0	0	0	0	1	0

Tabelle 42: Byte 5: Zeitwert „xx:--“ Zeitbasis „H:M“

Wert (bcd)	Wert x:-:- „H:-“				Wert -x:-:- „H:-“			
	Bit				Bit			
	7	6	5	4	3	2	1	0
00	0	0	0	0	0	0	0	0
05	0	0	0	0	0	1	0	1
17	0	0	0	1	0	1	1	1

Tabelle 43: Byte 6: Zeitwert in Tage

Wert (bcd)	Wert				Wert Tage			
	Bit				Bit			
	7	6	5	4	3	2	1	0
00	0	0	0	0	0	0	0	0
04	0	0	0	0	0	1	0	0

Beispiel 1: Zeitrelais

Der Master erteilt den Befehl zum Schreiben des Zeitrelais T8 mit folgenden Werten:

Schaltfunktion	Ansprechverzögert
Zeitbereich	„S“
Sollzeit	50

T8 soll die Zeitbasis „M:S“ und die Sollzeit 30 Minuten, 25 Sekunden, erhalten.

Byte	Bedeutung	Wert	Bit								
			7	6	5	4	3	2	1	0	
0	Befehl: T8	08 _{hex}	0	0	0	0	0	1	0	0	0
1	Zeitrelaisfunktion, Zeitbasis, Parametermenü	→ Bit-Wert	1	0	0	0	0	1	0	0	0
2	Zeitwert „--.xx“ bei Zeitbasis „S“	→ Bit-Wert	0	0	0	0	0	0	0	0	0
3	Zeitwert „xx.--“ bei Zeitbasis „S“ oder „--.xx“ bei „M:S“	25 _{bcd}	0	0	1	0	0	0	1	0	1
4	Zeitwert „xx.--“ bei Zeitbasis „M:S“ oder „--.xx“ bei „H:M“	30 _{bcd}	0	0	1	1	0	0	0	0	0
5	Zeitwert „xx.--“ bei Zeitbasis „H:M“	→ Bit-Wert	0	0	0	0	0	0	0	0	0
6	Stundenwert in Tage	→ Bit-Wert	0	0	0	0	0	0	0	0	0

Der Slave antwortet mit folgenden Werten:

Byte	Bedeutung	Wert	Bit								
			7	6	5	4	3	2	1	0	
0	Antwort: Schreiben erfolgreich	C1 _{hex}	0	1	0	0	0	0	0	0	1
1	ungültig	→ Bit-Wert	x	x	x	x	x	x	x	x	x
2 - 6	Gleicher Inhalt wie beim Master										

Beispiel 2: Zeitrelais

Der Master erteilt den Befehl zum Schreiben des Zeitrelais T1 mit folgenden Werten:

Schaltfunktion	Rückschaltverzögert
Zeitbereich	„M:S“
Sollzeit	10:30

T1 soll den Zeitbereich „H:M“ und die Sollzeit 95 Stunden, 53 Minuten, erhalten. 95 Stunden = 3 Tage, 19 Stunden

Byte	Bedeutung	Wert	Bit								
			7	6	5	4	3	2	1	0	
0	Befehl: T1	01 _{hex}	1	0	0	0	0	0	0	0	1
1	Zeitrelaisfunktion, Zeitbasis, Parametermenü	→ Bit-Wert	1	0	0	1	0	0	0	0	1
2	Zeitwert „--.xx“ bei Zeitbasis „S“	→ Bit-Wert	0	0	0	0	0	0	0	0	0
3	Zeitwert „xx.--“ bei Zeitbasis „S“ oder „--:xx“ bei „M:S“	25 _{bcd}	0	0	0	0	0	0	0	0	0
4	Zeitwert „xx.--“ bei Zeitbasis „M:S“ oder „--:xx“ bei „H:M“	53 _{bcd}	0	1	0	1	0	0	1	1	
5	Zeitwert „xx.--“ bei Zeitbasis „H:M“	23 _{bcd}	0	0	1	0	0	0	1	1	
6	Stundenwert in Tage	03 _{bcd}	0	0	0	0	0	0	1	1	

Der Slave antwortet mit folgenden Werten:

Byte	Bedeutung	Wert	Bit								
			7	6	5	4	3	2	1	0	
0	Antwort: Schreiben erfolgreich	C1 _{hex}	1	1	0	0	0	0	0	0	1
1	ungültig	→ Bit-Wert	x	x	x	x	x	x	x	x	x
2 - 6	Gleicher Inhalt wie beim Master										

Zeitschaltuhr lesen (Kanal, ON-Zeit, OFF-Zeit)

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet		Bit										
		Master	Slave	7	6	5	4	3	2	1	0			
0	Befehl ¹⁾													
	☐1 Kanal A	2B	–	1/0	0	1	0	1	0	1	1			
	☐1 Kanal B	2C	–	1/0	0	1	0	1	1	0	0			
	☐1 Kanal C	2D	–	1/0	0	1	0	1	1	0	1			
	☐1 Kanal D	2E	–	1/0	0	1	0	1	1	1	0			
	☐2 Kanal A	2F	–	1/0	0	1	0	1	1	1	1			
	☐2 Kanal B	30	–	1/0	0	1	1	0	0	0	0			
	☐2 Kanal C	31	–	1/0	0	1	1	0	0	0	1			
	☐2 Kanal D	32	–	1/0	0	1	1	0	0	1	0			
	☐3 Kanal A	33	–	1/0	0	1	1	0	0	1	1			
	☐3 Kanal B	34	–	1/0	0	1	1	0	1	0	0			
	☐3 Kanal C	35	–	1/0	0	1	1	0	1	0	1			
	☐3 Kanal D	36	–	1/0	0	1	1	0	1	1	0			
	☐4 Kanal A	37	–	1/0	0	1	1	0	1	1	1			
	☐4 Kanal B	38	–	1/0	0	1	1	1	0	0	0			
	☐4 Kanal C	39	–	1/0	0	1	1	1	0	0	1			
	☐4 Kanal D	3A	–	1/0	0	1	1	1	0	1	0			
	Antwort ¹⁾													
	Lesen erfolgreich	–	C2/42	1/0	1	0	0	0	0	0	1	0		
	Befehl zurückgewiesen	–	C0/40	1/0	1	0	0	0	0	0	0	0		
1	ungültig	00	→ Bit-Wert	x	x	x	x	x	x	x	x	x		
2	Wochentag, Parametermenü-Anzeige	00	→ Tabelle 44											
3	Minute (Schaltpunkt ON)	00	→ Tabelle 45											
4	Stunde (Schaltpunkt ON)	00	→ Tabelle 46											
5	Minute (Schaltpunkt OFF)	00	→ Tabelle 47											
6	Stunde (Schaltpunkt OFF)	00	→ Tabelle 48											

1) Beachten Sie das Datenaustauschverfahren Byte 0 Bit 7, → Seite 59.

Tabelle 44: Byte 2: Wochentag, beginnend, endend, Parametermenüanzeige

	Bit							
	7	6	5	4	3	2	1	0
Tag ON								
keiner eingestellt						0	0	0
Montag						0	0	1
Dienstag						0	1	0
Mittwoch						0	1	1
Donnerstag						1	0	0
Freitag						1	0	1
Samstag						1	1	0
Sonntag						1	1	1
Tag OFF								
keiner eingestellt			0	0	0			
Montag			0	0	1			
Dienstag			0	1	0			
Mittwoch			0	1	1			
Donnerstag			1	0	0			
Freitag			1	0	1			
Samstag			1	1	0			
Sonntag			1	1	1			
Schaltzeitpunkt								
ON > OFF		1						
ON < OFF		0						
Erscheint imParametermenü								
nein	1							
ja	0							

Tabelle 45: Byte 3: Minute (Schaltpunkt ON)

Wert (bcd)	Bit							
	7	6	5	4	3	2	1	0
00	0	0	0	0	0	0	0	0
...
59	0	1	0	1	1	0	0	1

Tabelle 46: Byte 4: Stunde (Schaltpunkt ON)

Wert (bcd)	Bit							
	7	6	5	4	3	2	1	0
00	0	0	0	0	0	0	0	0
...
23	0	0	1	0	0	0	1	1

Tabelle 47: Byte 5: Minute (Schaltpunkt OFF)

Wert (bcd)	Bit							
	7	6	5	4	3	2	1	0
00	0	0	0	0	0	0	0	0
...
59	0	1	0	1	1	0	0	1

Tabelle 48: Byte 6: Stunde (Schaltpunkt OFF)

Wert (bcd)	Bit							
	7	6	5	4	3	2	1	0
00	0	0	0	0	0	0	0	0
...
23	0	0	1	0	0	0	1	1

Beispiel

Der Master erteilt den Befehl, die Werte des Kanal „A“ der ④4 zu lesen:

Byte	Bedeutung	Wert (hex)	Bit							
			7	6	5	4	3	2	1	0
0	Befehl: ④4 Kanal A	37	0	0	1	1	0	1	1	1
1 - 6		00								

Der Slave antwortet mit folgenden Werten:

- Tag: Montag (001) bis Freitag (101)
- ON: 19:00
- OFF: 06:30
- Schalterpunkt ON > OFF (1)
- Kanal erscheint im Parametermenü

Byte	Bedeutung	Wert	Bit							
			7	6	5	4	3	2	1	0
0	Antwort: Lesen erfolgreich	42	0	1	0	0	0	0	1	0
1	ungültig		x	x	x	x	x	x	x	x
2	Wochentag, Parametermenü-Anzeige	69 _{bcd}	0	1	1	0	1	0	0	1
3	Minute (Schalterpunkt ON)	00 _{bcd}	0	0	0	0	0	0	0	0
4	Stunde (Schalterpunkt ON)	19 _{bcd}	0	0	0	1	1	0	0	1
5	Minute (Schalterpunkt OFF)	30 _{bcd}	0	0	1	1	0	0	0	0
6	Stunde (Schalterpunkt OFF)	06 _{bcd}	0	0	0	0	0	1	1	0

Zeitschaltuhr schreiben (Kanal, ON-Zeit, OFF-Zeit)

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet		Bit										
		Master	Slave	7	6	5	4	3	2	1	0			
0	Befehl ¹⁾													
	ⓐ1 Kanal A	12	–	1/0	0	0	1	0	0	1	0			
	ⓐ1 Kanal B	13	–	1/0	0	0	1	0	0	1	1			
	ⓐ1 Kanal C	14	–	1/0	0	0	1	0	1	0	0			
	ⓐ1 Kanal D	15	–	1/0	0	0	1	0	1	0	1			
	ⓐ2 Kanal A	16	–	1/0	0	0	1	0	1	1	0			
	ⓐ2 Kanal B	17	–	1/0	0	0	1	0	1	1	1			
	ⓐ2 Kanal C	18	–	1/0	0	0	1	1	0	0	0			
	ⓐ2 Kanal D	19	–	1/0	0	0	1	1	0	0	1			
	ⓐ3 Kanal A	1A	–	1/0	0	0	1	1	0	1	0			
	ⓐ3 Kanal B	1B	–	1/0	0	0	1	1	0	1	1			
	ⓐ3 Kanal C	1C	–	1/0	0	0	1	1	1	0	0			
	ⓐ3 Kanal D	1D	–	1/0	0	0	1	1	1	0	1			
	ⓐ4 Kanal A	1E	–	1/0	0	0	1	1	1	1	0			
	ⓐ4 Kanal B	1F	–	1/0	0	0	1	1	1	1	1			
	ⓐ4 Kanal C	20	–	1/0	0	1	0	0	0	0	0			
	ⓐ4 Kanal D	21	–	1/0	0	1	0	0	0	0	1			
	Antwort ¹⁾													
	Schreiben erfolgreich	–	C1/41	1/0	1	0	0	0	0	0	0	1		
	Befehl zurückgewiesen	–	C0/40	1/0	1	0	0	0	0	0	0	0		
1	Wochentag, Parametermenü-Anzeige	00	→ Tabelle 44											
2	Minute (Schaltpunkt ON)	00	→ Tabelle 45											
3	Stunde (Schaltpunkt ON)	00	→ Tabelle 46											
4	Minute (Schaltpunkt OFF)	00	→ Tabelle 47											
5	Stunde (Schaltpunkt OFF)	00	→ Tabelle 48											
6	nicht benutzt													

1) Beachten Sie das Datenaustauschverfahren Byte 0 Bit 7, → Seite 59.

Die Werte von Minute und Stunde der jeweiligen Schaltpunkte sind Bestandteil einer „*.eas-Datei“. Werden diese Werte verändert, stimmt die Original „*.eas-Datei“ mit der in EASY6... befindlichen nicht mehr überein.

Beachten Sie diese Eigenschaft beim Hoch- und beim Runterladen und Vergleich von „easy“-Schaltplänen mit der easy-Soft.

Beim Runterladen vom PC wird der aktuelle Stand der „*.eas“ überschrieben.

Beim Vergleich sind die Schaltpläne ungleich.

Beispiel

Der Master erteilt den Befehl, in den Kanal „C“ der 02 folgende Daten zu schreiben:

- Tag: Dienstag (010) bis Samstag (110)
- ON: 10:00
- OFF: 17:30
- Schaltpunkt ON < OFF (0)
- Kanal erscheint nicht im Parametermenü (1)

Byte	Bedeutung	Wert	Bit							
			7	6	5	4	3	2	1	0
0	Befehl: 02 Kanal C	18 _{hex}	0	0	0	1	1	0	0	0
1	Wochentag, Parametermenü-Anzeige	B2 _{hex}	1	0	1	1	0	0	1	0
2	Minute (Schaltpunkt ON)	00 _{bcd}	0	0	0	0	0	0	0	0
3	Stunde (Schaltpunkt ON)	10 _{bcd}	0	0	0	1	0	0	0	0
4	Minute (Schaltpunkt OFF)	30 _{bcd}	0	0	1	1	0	0	0	0
5	Stunde (Schaltpunkt OFF)	17 _{bcd}	0	0	0	1	0	1	1	1
6	nicht benutzt									

Der Slave antwortet mit folgendem Telegramm:

Byte	Bedeutung	Wert	Bit							
			7	6	5	4	3	2	1	0
0	Antwort: Schreiben erfolgreich	41 _{hex}	0	1	0	0	0	0	0	1
1 - 6		00								

8 Steuerbefehle easy700 (DPV0)

Datenaustauschverfahren Das Modul „Steuerbefehle 9 Byte“ erlaubt den erweiterten Datenaustausch des easy700 am Kommunikationsbus PROFIBUS-DP. Damit können Sie Dienste aus folgenden Bereichen übertragen:

- „Datum und Uhrzeit lesen/schreiben“ (Seite 104)
- „Abbilddaten lesen/schreiben“ (Seite 108) und
- „Funktionsbaustein-Daten lesen/schreiben“ (Seite 131).

Damit die Daten sicher vom Master zum Slave und umgekehrt über PROFIBUS-DP übertragen werden, ist ein Datenaustauschverfahren notwendig.



Achtung!

Während der Benutzung eines Steuerbefehls behalten die Ein- und Ausgangsdaten den Zustand vor Aufruf des Steuerbefehls. Erst wenn der Datenaustausch des Steuerbefehls beendet ist, sind die Ein-/Ausgangsdaten wieder aktuell.



Warnung!

Es dürfen nur die für den Befehlscode angegebenen Werte verwendet werden.

Überprüfen Sie die Werte, die Sie schreiben, um Fehlfunktionen zu vermeiden.

Voraussetzung:

Das entsprechende Modul „Steuerbefehle 9 Byte“ wurde gewählt.

Der Master löst den Datenaustausch der Steuerbefehle aus und der angesprochene Slave antwortet.

Bei der Kommunikation werden jeweils 9 Datenbytes (Byte 0 = Toggelbyte, Byte 1 bis 8 mit Informationsgehalt) auf dem PROFIBUS übergeben.

Der prinzipielle Aufbau des Telegramms ist in der nachfolgenden Abbildung dargestellt.

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
--------	--------	--------	--------	--------	--------	--------	--------	--------

Byte 0 – Toggelbyte

Das Byte 0 dient dazu, mit der Toggelfunktion das Senden eines Steuerbefehls zu aktivieren.

Bit	7	6	5	4	3	2	1	0
01 _{hex} /81 _{hex}	0/1	0	0	0	0	0	0	1
	Toggelbit	fix						

Vorgehensweise

- ▶ Um einen Befehl abzusetzen, wird das Bit 7 getoggelt; d. h. von 1 auf 0 oder von 0 auf 1 gesetzt.
- ▶ Pollen Sie anschließend so lange auf das Toggelbit der Antwort des Koppelmoduls, bis es den gleichen Zustand wie das gesendete Toggelbit angenommen hat. Dieser Zustand zeigt der Mastersteuerung an, dass die Antwort auf den abgesetzten Befehl gültig ist.
- ▶ Senden Sie einen neuen Befehl erst ab, wenn Sie eine Antwort erhalten haben (Ändern des Toggelbits); sonst wird eventuell die Antwort des vorherigen Befehls überschrieben, bevor diese gelesen werden kann.



Wenn Sie Ein-/Ausgangsdaten und Steuerbefehle gleichzeitig benutzen, gilt:

Wenn der Datenaustausch von Steuerbefehlen beendet ist, werden die Bytes der Ein-Ausgänge wieder aktualisiert.

Alle angegebenen Kommandos und Parameter müssen im hexadezimalen Zahlenformat übergeben werden.

In den nachfolgenden Tabellen sind die verschiedenen Steuerbefehle zusammengefasst. Hierbei werden drei wesentliche Rubriken unterschieden – Echtzeituhr, Abbild und Funktionsbausteine.

Datum und Uhrzeit lesen/schreiben



Beachten Sie bitte auch die entsprechende Beschreibung zur Echtzeituhr im easy700-Handbuch (MN05013003Z-DE; frühere Bezeichnung AWB2528-1508D).

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl		
	Lesen	93	–
	Schreiben	B3	–
	Antwort		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Len	05	05
3	Index	0 - 2 ¹⁾	0 - 2 ¹⁾
4 - 8	Data 1 - 4	abhängig vom Index, → Tabelle 49	

- 1) 0 = Uhrzeit/Datum, → Tabelle 49
 1 = Sommerzeit, → Tabelle 50
 2 = Winterzeit, → Tabelle 51

Tabelle 49: Index 0 – Datum und Zeit der Echtzeituhr

Byte	Inhalt	Operand		Wert (hex)
4	Data 1	Stunde	0 bis 23	0x00 bis 0x17h
5	Data 2	Minute	0 bis 59	0x00 bis 0x3Bh
6	Data 3	Tag	1 bis 28; 29, 30, 31 (abhängig von Monat und Jahr)	0x01 bis 0x1Fh
7	Data 4	Monat	1 bis 12	0x01 bis 0x0Ch
8	Data 5	Jahr	0 bis 99 (entspricht 2000-2099)	0x00 bis 0x63h

Tabelle 50: Index 1 – Sommerzeit

Byte	Inhalt		Wert (hex)
4	Data 1	Area	
		keine	00
		Regel	01
		automatische EU	02
		automatische GB	03
		automatische US	04
für „Area“ = „Regel“:			
5	Data 2	Schaltregel Sommerzeit	→ Tabelle 52
6	Data 3		
7	Data 4		
8	Data 5		

Tabelle 51: Index 2 – Winterzeit
(nur gültig, wenn Area = „Regel“ gewählt worden ist)

Byte	Inhalt		Wert (hex)
4	Data 1	Area = Regel	01
5 - 8	Data 2 - 5	Schaltregel Winterzeit	→ Tabelle 52

Bitfeld „Schaltregel“



Lesen Sie hierzu bitte die detaillierte Beschreibung im easy700-Handbuch (MN05013003Z-DE; frühere Bezeichnung AWB2528-1508D).

Die nachfolgende Tabelle gibt Ihnen die Vorschrift zum Zusammensetzen der entsprechenden Datenbytes an.

Tabelle 52: Bitfeld Schaltregel

Bit	Data 5				Data 4				Data 3				Data 2																		
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
	Differenz				Zeit der Umstellung				Monat				Tag				Regel_2		Tag		Regel_1										
0:	0:30h				Minute: 0 bis 59				Stunde: 0 bis 23				0 bis 11				0 bis 30				0: im		0: So		0: am						
1:	1:00h																1: nach dem		1: Mo		1: am ersten										
2:	1:30h																2: vor dem		2: Di		2: am zweiten										
3:	2:00h																3: Mi		3: am dritten												
4:	2:30h																4: Do		4: am vierten												
5:	3:00h																5: Fr		5: am letzten												
													6: Sa																		

**Abbilddaten lesen/
schreiben**

Beachten Sie bitte auch die entsprechende Beschreibung zu den möglichen Abbilddaten, die im easy700-Handbuch (MN05013003Z-DE; frühere Bezeichnung AWB2528-1508D) oder in der easySoft-Hilfe angeführt sind.

Die aktuelle Ausgabe des Handbuchs ist als PDF-Datei im Internet verfügbar:

<http://www.eaton.com/moeller> → Support

→ Suchbegriff: MN05013003Z-DE

Desweiteren gilt der Abschnitt „Generelles zum Arbeiten mit Abbilddaten“ auf Seite 64 auch für easy700.

Übersicht

Operanden	Bedeutung	Lesen/ schreiben	Type (hex)	Seite
A1 - A16	„Analogwertvergleich/Schwellwertvergleich: A1 – A16“	lesen	8B	110
C1 - C16	„Zähler: C1 – C16“	lesen	EE	111
D1 - D16	„Textbausteine: D1 – D16“	lesen	94	112
I1 - I16	„Lokale Eingänge: I1 - I16“	lesen	84	113
IA1 - IA4	„Lokale Analog-Eingänge: IA1 - IA4“	lesen	8C	115
M1 - M16, N1 - N16	„Merker: M1 – M16/N1 – N16“	schreiben	86/87	117
M1 - M16, N1 - N16	„Merker: M1 – M16/N1 – N16“	lesen	86/87	119
O1 - O4	„Betriebsstundenzähler: O1 – O4“	lesen	EF	121
P1 - P4	„Lokale P-Tasten: P1 - P4“	lesen	8A	122
Q1 - Q8	„Lokale Ausgänge: Q1 - Q8“	lesen	85	124
R1 - R16/ S1 - S8	„Ein-/Ausgänge von easyLink: R1 – R16/ S1 – S8“	lesen	88/89	125
T1 - T16	„Zeitglieder: T1 - T16“	lesen	ED	127
Y1 - Y4	„Jahresschaltuhr: Y1 - Y8“	lesen	91	128
Z1 - Z3	„Masterreset: Z1 - Z3“	lesen	93	129
H1 - H4	„Wochenschaltuhr: ☐1 - ☐8“	lesen	90	130

Analogwertvergleichler/Schwellwertvergleichler: A1 – A16

Mit den nachfolgenden Befehlen lesen Sie den logischen Status der einzelnen Analogwertvergleichler A1 bis A16 aus.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl: Lesen	88	–
	Antwort: Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Len	01	01
3	Type	8B	8B
4	Index	00	00
5	Data 1 (Low Byte)	00	→ Tabelle 53
6	Data 2 (Low Byte)	00	→ Tabelle 53
7 - 8	Data 3 - 4	00	00

1) Mögliche Ursachen → Seite 150

Tabelle 53: Byte 5 bis 6: Data 1 bis 2

Data 1	Bit	7	6	5	4	3	2	1	0
A1									0/1
A2								0/1	
...					...				
A8		0/1							
Data 2	Bit	7	6	5	4	3	2	1	0
A9									0/1
A10								0/1	
...					...				
A16		0/1							

Zähler: C1 – C16

Mit den nachfolgenden Befehlen lesen Sie den logischen Status der einzelnen Zähler C1 – C16 aus.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl: Lesen	88	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Len	01	01
3	Type	EE	EE
4	Index	00	00
5	Data 1 (Low Byte)	00	→ Tabelle 63
6	Data 2 (Low Byte)	00	→ Tabelle 63
7 - 8	Data 3 - 4	00	00

1) Mögliche Ursachen → Seite 150

Tabelle 54: Byte 5 bis 6: Data 1 bis 2

Data 1	Bit	7	6	5	4	3	2	1	0
C1									0/1
C2								0/1	
...					...				
C8		0/1							
Data 2	Bit	7	6	5	4	3	2	1	0
C9									0/1
C10								0/1	
...					...				
C16		0/1							

Textbausteine: D1 – D16

Mit den nachfolgenden Befehlen lesen Sie den logischen Status der einzelnen Textbausteine (D-Merker) aus.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl: Lesen	88	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Len	01	01
3	Type	94	94
4	Index	00	00
5	Data 1 (Low Byte)	00	→ Tabelle 55
6	Data 2 (High Byte)	00	→ Tabelle 55
7 - 8	Data 3 - 4	00	00

1) Mögliche Ursachen → Seite 150

Tabelle 55: Byte 5 bis 6: Data 1 bis 2

Data 1	Bit	7	6	5	4	3	2	1	0
D1									0/1
D2								0/1	
...				...					
D8		0/1							
Data 2	Bit	7	6	5	4	3	2	1	0
D9									0/1
D10								0/1	
...				...					
D16		0/1							

Lokale Eingänge: I1 - I16

Mit diesem Befehlsstring können Sie die lokalen Eingänge des Basisgerätes easy700 auslesen. Das entsprechende Eingangswort ist hierbei im Intel-Format abgelegt.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl: Lesen	88	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Len	02	02
3	Type	84	84
4	Index	00	00
5	Data 1 (Low Byte)	00	→ Tabelle 56
6	Data 2 (High Byte)	00	→ Tabelle 56
7 - 8	Data 3 - 4	00	00

1) Mögliche Ursachen → Seite 150

Tabelle 56: Byte 5 bis 6: Data 1 bis 2

Data 1	Bit	7	6	5	4	3	2	1	0
I1									0/1
I2									0/1
...					...				
I8			0/1						
Data 2	Bit	7	6	5	4	3	2	1	0
I9									0/1
I10									0/1
...					...				
I16			0/1						

Lokale Analog-Eingänge: IA1 - IA4

Die am Basisgerät easy700 vorhandenen Analog-Eingänge (I7, I8, I11, I12) können Sie direkt über den PROFIBUS-DP auslesen. Hierbei wird der 16-Bitwert im Intelformat übergeben (Low Byte first).

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl: Lesen	88	–
	Antwort: Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Len	02	02
3	Type	8C	8C
4	Index	00 - 03 ²⁾	00 - 03 ²⁾
5	Data 1 (Low Byte)	00	→ Tabelle 57
6	Data 2 (High Byte)	00	→ Tabelle 57
7 - 8	Data 3 - 4	00	00

1) Mögliche Ursachen → Seite 150

- 2) 00 = Analog-Eingang I7
- 01 = Analog-Eingang I8
- 02 = Analog-Eingang I11
- 03 = Analog-Eingang I12

Beispiel:

Es liegt ein Spannungspegel am Analog-Eingang „1“ an. Die entsprechenden Telegramme zum Lesen des Analogwertes sehen wie folgt aus:

Tabelle 57: Beispiel-Telegramm zum Lesen des Wertes am Analog-Eingang „1“

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl: Lesen	88	–
	Antwort: Lesen erfolgreich	–	C2
2	Len	02	02
3	Type	8C	8C
4	Index	02 ¹⁾	02 ¹⁾
5	Data 1	00	4B
6	Data 2	00	03
7	Data 3	00	00
8	Data 4	00	00

1) 02 = Analog-Eingang I11

Byte 5 – Data 1 (Low Byte): 4B_{hex}

Byte 6 – Data 2 (High Byte): 03_{hex}

→ entsprechender 16-Bitwert: 034B_{hex} = 843

Der Wert „843“ entspricht dem IO-Bit-Wert des Analogwandlers. Für den tatsächlichen Analogwert ist folgende Umrechnung notwendig:

$$\frac{10 \text{ V}}{1023} \times \text{IO-Bitwert} \Rightarrow \frac{10 \text{ V}}{1023} \times 843 = 8,24 \text{ V}$$

Merker: M1 – M16/N1 – N16

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl:Schreiben	8C	–
	Antwort: Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Len	01	01
3	Type ²⁾		
	bei Merker M	86	86
	bei Merker N	87	87
4	Index ²⁾	00 - 0F	00 - 0F
5	Data 1 (Low Byte) ³⁾	00/01	00/01
6 - 8	Data 2 - 4	00	00

- 1) Mögliche Ursachen → Seite 150
- 2) Es gibt 16 M-Merker und 16 N-Merker. Die Adressierung der Merker differenziert sich durch „Type“ und „Index“:
Mit „Type“ wählen Sie den Merker-Typ „M“ bzw. „N“.
Mit „Index“ bestimmen Sie die Merker-Nummer.
- 3) Der Merker wird gesetzt, wenn ein Wert ungleich Null auf das Datenbyte geschrieben wird. Wird auf das Datenbyte „Data 1“ der Wert „0“ geschrieben, wird der Merker entsprechend zurückgesetzt.

Beispiel:

Es wird der Merker M13 gesetzt.

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl:Schreiben	8C	–
	Antwort:		
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Len	01	01
3	Type		
	Merker M	86	86
4	Index	0C	0C
5	Data 1	01	00
6 - 8	Data 2 - 4	00	00

1) Mögliche Ursachen → Seite 150

Merker: M1 – M16/N1 – N16

Beim Lesen der Merker wird – im Gegensatz zum Schreiben – der gesamte Merkerdatenbereich eines Merker-Typs (M oder N) ausgelesen.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl: Lesen	88	–
	Antwort: Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Len	01	01
3	Type		
	Merker M	86	86
	Merker N	87	87
4	Index ²⁾	00	00
5	Data 1 (Low Byte)	00	→ Tabelle 58
6	Data 2 (Low Byte)	00	→ Tabelle 58
7 - 8	Data 3 - 4	00	00

- 1) Mögliche Ursachen → Seite 150
- 2) Es gibt 16 M-Merker und 16 N-Merker. Die Adressierung der Merker differenziert sich durch „Type“ und „Index“:
 Mit „Type“ wählen Sie den Merker-Typ „M“ bzw. „N“.
 Mit „Index“ bestimmen Sie die Merker-Nummer.

Tabelle 58: Byte 5 bis 6: Data 1 bis 2

Data 1		Bit	7	6	5	4	3	2	1	0
M	N									
M1	N1									0/1
M2	N2									0/1
...				
M8	N8		0/1							
Data 2		Bit	7	6	5	4	3	2	1	0
M9	N9									0/1
M10	N10									0/1
...	-					...				
M16	N16		0/1							

Beispiel: Es werden die „N“-Merker gelesen:

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl: Lesen	88	-
	Antwort:		
	Lesen erfolgreich	-	C2
	Befehl zurückgewiesen	-	C0 ¹⁾
2	Len	01	01
3	Type		
	Merker N	87	87
4	Index	00	00
5	Data 1 (Low Byte)	00	04
6	Data 2 (Low Byte)	00	84
7 - 8	Data 3 - 4	00	00

1) Mögliche Ursachen → Seite 150

Die Merker N3, N11 und N16 sind gesetzt.

Betriebsstundenzähler: 01 – 04

Mit den nachfolgenden Befehlen lesen Sie den logischen Status der einzelnen Betriebsstundenzähler 01 – 04 aus.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl: Lesen	88	–
	Antwort: Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Len	01	01
3	Type	EF	EF
4	Index	00	00
5	Data 1 (Low Byte)	00	→ Tabelle 59
6 - 8	Data 2 - 4	00	00

1) Mögliche Ursachen → Seite 150

Tabelle 59: Byte 5 : Data 1

Data 1	Bit 7	6	5	4	3	2	1	0
01								0/1
02								0/1
03								0/1
04								0/1
...

Lokale P-Tasten: P1 - P4

Die lokalen P-Tasten sind die Display-Courser-Tasten des easy700-Basisgerätes. Die Tasten können Sie sowohl im Modus RUN als auch im Modus STOP abfragen.



Achten Sie nur darauf, dass die P-Tasten auch unter dem Menüpunkt „System“ (im Basisgerät) aktiviert sind.

Bei den P-Tasten müssen Sie nur ein Byte übergeben.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl: Lesen	88	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Len	01	01
3	Type	8A	8A
4	Index	00	00
5	Data 1 (Low Byte)	00	→ Tabelle 60
6 - 8	Data 2 - 4	00	00

1) Mögliche Ursachen → Seite 150

Tabelle 60: Byte 5: Data 1

Data 1	Bit 7	6	5	4	3	2	1	0
P1								0/1
P2								0/1
P3								0/1
P4								0/1
–					0			
–				0				
–			0					
–	0							

Beispiel:

Data 1 = 2_{hex} → P3 ist aktiv.

Lokale Ausgänge: Q1 - Q8

Die lokalen Ausgänge können Sie über den PROFIBUS-DP-Feldbus direkt lesen.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl: Lesen	88	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Len	01	01
3	Type	85	85
4	Index	00	00
5	Data 1 (Low Byte)	00	→ Tabelle 61
6 - 8	Data 2 - 4	00	00

1) Mögliche Ursachen → Seite 150

Tabelle 61: Byte 5: Data 1

Data 1	Bit 7	6	5	4	3	2	1	0
Q1								0/1
Q2								0/1
...				...				
Q8	0/1							

Beispiel:

Data 1 = 52_{hex} → Q2, Q5 und Q7 sind aktiv.

Ein-/Ausgänge von easyLink: R1 – R16/S1 – S8

Mit diesem Dienst können Sie die lokalen R- und S-Daten und die der NET-Teilnehmer (1 – 8), die über easyLink transferiert werden, nochmals aus dem entsprechenden Abbild der easy700 lesen.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl: Lesen	88	–
	Antwort: Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Len	01	01
3	Type		
	bei R-Daten	88	88
	bei S-Daten	89	89
4	Index	00	00
5	Data 1 (Low Byte)	00	→ Tabelle 62
6	Data 2 (Low Byte)	00	→ Tabelle 62
7 - 8	Data 3 - 4	00	00

1) Mögliche Ursachen → Seite 150

Tabelle 62: Byte 5 bis 6: Data 1 bis 2

Data 1		Bit	7	6	5	4	3	2	1	0
RW	SW									
R1	S1									0/1
R2	S2									0/1
...				
R8	S8		0/1							
Data 2		Bit	7	6	5	4	3	2	1	0
R9	–									0/1
R10	–									0/1
...	–					...				
R16	–		0/1							

Zeitglieder: T1 - T16

Mit den nachfolgenden Befehlen lesen Sie den logischen Status der einzelnen Zeitglied T1-T16 aus.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl: Lesen	88	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Len	01	01
3	Type	ED	ED
4	Index	00	00
5	Data 1 (Low Byte)	00	→ Tabelle 63
6	Data 2 (Low Byte)	00	→ Tabelle 63
7 - 8	Data 3 - 4	00	00

1) Mögliche Ursachen → Seite 150

Tabelle 63: Byte 5 bis 6: Data 1 bis 2

Data 1	Bit	7	6	5	4	3	2	1	0
T1									0/1
T2								0/1	
...					...				
T8		0/1							
Data 2	Bit	7	6	5	4	3	2	1	0
T9									0/1
T10								0/1	
...					...				
T16		0/1							

Jahresschaltuhr: Y1 - Y8

Mit den nachfolgenden Befehlen können Sie den logischen Status der einzelnen Jahresschaltuhren auslesen.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl: Lesen	88	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Len	01	01
3	Type	91	91
4	Index	00	00
5	Data 1 (Low Byte)	00	→ Tabelle 64
6 - 8	Data 2 - 4	00	00

1) Mögliche Ursachen → Seite 150

Tabelle 64: Byte 5: Data 1

Data 1	Bit	7	6	5	4	3	2	1	0
HY1									0/1
HY2								0/1	
HY3							0/1		
HY4						0/1			
HY5				0					
HY6			0						
HY7		0							
HY8	0								

Beispiel:

Data 1 = 1_{hex} → HY2 ist aktiv

Masterreset: Z1 - Z3

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl: Lesen	88	–
	Antwort: Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Len	01	01
3	Type	93	93
4	Index	00	00
5	Data 1 (Low Byte)	00	→ Tabelle 65
6 - 8	Data 2 - 4	00	00

1) Mögliche Ursachen → Seite 150

Tabelle 65: Byte 5: Data 1

Data 1	Bit 7	6	5	4	3	2	1	0
Z1 für Ausgänge Q								0/1
Z2 für Merker M							0/1	
Z3 für Ausgänge und Merker						0/1		
...	0	0	0	0	0			

Wochenschaltuhr: 01 – 08

Mit den nachfolgenden Befehlen lesen Sie den logischen Status der einzelnen Wochenschaltuhren aus.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl: Lesen	88	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Len	01	01
3	Type	90	90
4	Index	00	00
5	Data 1 (Low Byte)	00	→ Tabelle 66
6 - 8	Data 2 - 4	00	00

1) Mögliche Ursachen → Seite 150

Tabelle 66: Byte 5: Data 1

Data 1	Bit 7	6	5	4	3	2	1	0
HW1								0/1
HW2								0/1
HW3								0/1
HW4								0/1
HW5								0
HW6								0
HW7								0
HW8								0

Beispiel:

Data 1 = 2_{hex} → 03 ist aktiv.

**Funktionsbaustein-Daten
lesen/schreiben**



Beachten Sie bitte auch die entsprechende Beschreibung zu den Funktionsbausteinen, die im easy700-Handbuch (MN05013003Z-DE; frühere Bezeichnung AWB2528-1508D) oder in der easySoft-Hilfe angeführt sind.

Allgemeine Hinweise

Grundsätzlich müssen Sie beim Arbeiten mit den Funktionsbausteinen Folgendes beachten:

- Die entsprechenden Daten werden im Intelformat übergeben: das erste Byte ist das Low Byte (Byte 5) und das letzte das High Byte (Byte 8).
- Die Datenlänge beträgt bis zu 4 Byte. Alle Werte müssen Sie im hexadezimalen Zahlenformat übergeben.

Übersicht

Operanden	Bedeutung	Lesen/ schreiben	Type	Seite
A1 - A16	„Analogwertvergleich/Schwellwertschalter: A1 - A16“	Lesen/ schreiben	8D	132
C1 - C16	„Zählerrelais: C1 - C16“	Lesen/ schreiben	8F	135
O1 - O4	„Betriebsstundenzähler: O1 - O4“	Lesen/ schreiben	92	138
T1 - T16	„Zeitrelais: T1 - T16“	Lesen/ schreiben	8E	140
Y1 - Y8	„Jahresschaltuhr: Y1 - Y8“	Lesen/ schreiben	A2	144
Q1 - Q8	„Wochenschaltuhr: Q1 - Q8“	Lesen/ schreiben	A1	147

Analogwertvergleichler/Schwellwertschalter: A1 - A16

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl:		
	Lesen	89	–
	Schreiben	8D	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Type	8D	8D
3	Instanz ²⁾	00 - 0F	00 - 0F
4	Index	→ Tabelle 67	
5 - 8	Data 1 - 4	abhängig vom Index, → Tabelle 68	

1) Mögliche Ursachen → Seite 150

2) easy stellt 16 Analogwertvergleichler A1 bis A16 zur Verfügung. Diese können über die Instanz (0 - F) angesprochen werden.

Tabelle 67: Operandenübersicht

Index (hex)	Operand		Lesen	Schreiben
00	Parameter → Tabelle 68		×	
01	Kontrollbyte → Tabelle 69		×	
02	Vergleichswert 1	I1 ²⁾	×	c ¹⁾
03	Vergleichswert 2	I2 ²⁾	×	c ¹⁾
04	Verstärkungsfaktor für I1 (I1 = F1 × I1)	F1 ²⁾	×	c ¹⁾
05	Verstärkungsfaktor für I2 (I2 = F2 × I2)	F2 ²⁾	×	c ¹⁾
06	Offset für den Wert I1 (I1 = OS + Istwert an I1)	OS ²⁾	×	c ¹⁾
07	Schalthyserese für den Wert I2	HY ²⁾	×	c ¹⁾

- 1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.
- 2) In den Datenbytes Data 1 – Data 2 wird ein 16-Bit-Wert übergeben. Dabei ist zu beachten, dass das Low-Byte in Data 1 (Byte 5) und das High-Byte in Data 2 (Byte 8) geführt wird.
Beispiel: 5327_{dez} = 14CF_{hex} → Data 1 = 0xCF, Data 2 = 0x14

Tabelle 68: Index 00 – Parameter

Bedeutung	Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Erscheint im Parametermenü																	
ja/nein																	0/1
Vergleich auf																	
FB nicht benutzt														0	0	0	
EQ (=)														0	0	1	
GE (\geq)														0	1	0	
LE (\leq)														0	1	1	
GT (>)														1	0	0	
LT (<)														1	0	1	
Verwendung als Konstante und somit beschreibbar																	
I1= Konstante														0/1			
F1= Konstante													0/1				
I2= Konstante											0/1						
F2 = Konstante										0/1							
OS = Konstante								0/1									
HY = Konstante							0/1										
Nicht verwendet		0	0	0	0	0	0										

Beispiel:

Data 1 (Byte 5) = 0xA3, Data 2 (Byte 6) = 0x03

→ Resultierender 16-Bit-Wert = 03A3_{hex}

Bedeutung: HY, OS, F2, F1 sind mit einer Konstanten beschaltet; I1, I2 sind mit einer Variablen z. B. I7, I8 C2... usw. beschaltet; erscheint im Parametermenü;

Der Ausgang des Analogwertvergleichers wird aktiv, solange die Gleichung $(I1 \times F1) + OS = (I2 \times F2) + HY$ erfüllt ist.

Tabelle 69: Index 01 – Kontrollbyte

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		-	-	-	-	-	-	-	Q1 ¹⁾

1) Zustand „1“, wenn die Vergleichsbedingung erfüllt ist.

Zählerrelais: C1 - C16

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl:		
	Lesen	89	–
	Schreiben	8D	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Type	8F	8F
3	Instanz ²⁾	00 - 0F	00 - 0F
4	Index	→ Tabelle 70	
5 - 8	Data 1 - 4	abhängig vom Index, → Tabelle 71	

1) Mögliche Ursachen → Seite 150

2) easy stellt 16 Zählerrelais C1 bis C16 zur Verfügung. Diese können über die Instanz (0 - F) angesprochen werden.

Tabelle 70: Operandenübersicht

Index (hex)	Operand		Lesen	Schreiben
00	Parameter → Tabelle 71		×	
01	Kontrollbyte → Tabelle 72		×	
02	Istwert	S1 ²⁾	×	c ¹⁾
03	Zählersollwert 2	S2 ²⁾	×	c ¹⁾

- 1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.
- 2) In den Datenbytes Data 1 - Data 2 wird ein 16-Bit-Wert übergeben. Dabei ist zu beachten, dass das Low-Byte in Data 1 und das High-Byte in Data 2 geführt wird.

Tabelle 71: Index 00 – Parameter

Bedeutung	Bit	7	6	5	4	3	2	1	0
Erscheint im Parametermenü									
ja/nein									0/1
Zählermodus									
FB nicht benutzt							0	0	
Vor-/Rückwärtszähler (N)							0	1	
Schneller Vor-/Rückwärtszähler (H)							1	0	
Frequenzmesser (F)							1	1	
Verwendung als Konstante und somit beschreibbar									
Zählersollwert S1						0/1			
Nicht benutzte Bits		–	–	–	–				

Beispiel:

Data 1 (Byte 5) = 0x07

Bedeutung:

Die Werte erscheinen im Parametermenü. Der Zähler wird im Modus des Frequenzmessers benutzt. Der Zählersollwert S1 ist nicht mit einer Konstanten beschaltet und somit auch nicht beschreibbar.

Tabelle 72: Index 01 – Kontrollbyte

Data 1	Bit	7	6	5	4	3	2	1	0
		FB-Ausgang	–	–	–	–	C ⁴⁾	RE ³⁾	D ²⁾

- 1) Schaltkontakt
- 2) Zählrichtungsangabe: Zustand „0“ = vorwärts zählen
Zustand „1“ = rückwärts zählen
- 3) Reset, das Zeitrelais wird zurückgesetzt (Resetspule)
- 4) Zählerspule, zählt bei jeder positiven Flanke

Beispiel:

Der Istwert von C3 soll gelesen werden:

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Befehl: Lesen	89	–
	Antwort: Lesen erfolgreich	–	C2
1	Type	8F	8F
2	Instanz	02	02
3	Index	02	02
4	Data1	00	12
5	Data 2	00	03
6	Data 3	00	00
7	Data 4	00	00

Erklärung:

Data 1 = 12

Data 2 = 03

→ resultierender 16-Bit-Wert = 0312_{hex} = 786_{dez}

Zählerstand = 786

Betriebsstundenzähler: 01 - 04**Telegrammaufbau**

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl:		
	Lesen	89	–
	Schreiben	8D	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Type	92	92
3	Instanz ²⁾	00 - 03	00 - 03
4	Index	→ Tabelle 73	
5 - 8	Data 1 - 4	abhängig vom Index, → Tabelle 74	

- 1) Mögliche Ursachen → Seite 150
- 2) easy stellt 4 Betriebsstundenzähler 01 bis 04 zur Verfügung. Diese können über die Instanz (0 - 3) angesprochen werden.

Tabelle 73: Operandenübersicht

Index (hex)	Operand		Lesen	Schreiben
00	Parameter → Tabelle 74		×	
01	Kontrollbyte → Tabelle 75		×	
02	Istwert	S1 ²⁾	×	c ¹⁾
03	Zählersollwert 2	S2 ²⁾	×	c ¹⁾

- 1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.
- 2) In den Datenbytes Data 1 - Data 4 wird ein 32-Bit-Wert übergeben. Dabei ist zu beachten, dass das Low-Byte in Data 1 und das High-Byte in Data 4 geführt wird.

Tabelle 74: Index 00 – Parameter

Bedeutung	Bit	7	6	5	4	3	2	1	0
Erscheint im Parametermenü									
ja/nein									0/1
Verwendung im Programm									
Sollwert S1								0/1	
Nicht benutzte Bits		–	–	–	–	–	–		

Beispiel:
Data 1 (Byte 5) = 0x01

Bedeutung:
Die Werte erscheinen im Parametermenü.

Tabelle 75: Index 01 – Kontrollbyte

Data 1	Bit	7	6	5	4	3	2	1	0
FB-Ausgang		–	–	–	–	–	RE ³⁾	EN ²⁾	Q1 ¹⁾

- 1) Schaltkontakt
- 2) Enable, das Zeitrelais wird gestartet (Triggerspule)
- 3) Reset, das Zeitrelais wird zurückgesetzt (Resetspule)

Beispiel:
Index 02/03

Übergebene Werte: Data 1 0x21
Data 2 0x23
Data 3 0x40
Data 4 0x00

Resultierender Wert: 00402321_{hex} = 4203297_{dez}

Zeitrelais: T1 - T16**Telegrammaufbau**

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl:		
	Lesen	89	–
	Schreiben	8D	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Type	8E	8E
3	Instanz ²⁾	00 - 0F	00 - 0F
4	Index	→ Tabelle 76	
5 - 8	Data 1 - 4	abhängig vom Index, → Tabelle 77	

1) Mögliche Ursachen → Seite 150

2) easy stellt 16 Zeitrelais T1 bis T16 zur Verfügung. Diese können über die Instanz (0 - F) angesprochen werden.

Tabelle 76: Operandenübersicht

Index (hex)	Operand		Lesen	Schreiben
00	Parameter → Tabelle 77		×	
01	Kontrollbyte → Tabelle 78		×	
02	Istwert 1	T	×	c ¹⁾
03	Zeitsollwert 1	S1 ²⁾	×	c ¹⁾
04	Zeitsollwert 2	S2 ²⁾	×	c ¹⁾

- 1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.
- 2) In den Datenbytes Data 1 - Data 2 wird ein 16-Bit-Wert übergeben. Dabei ist zu beachten, dass das Low-Byte in Data 1 und das High-Byte in Data 2 geführt wird.

Tabelle 77: Index 00 – Parameter

Bedeutung	Bit	7	6	5	4	3	2	1	0
Erscheint im Parametermenü									
ja/nein									0/1
Timer-Modus									
Ansprechverzögert						0	0	0	
Rückfallverzögert						0	0	1	
Ansprechverzögert mit Zufalls-Sollwert						0	1	0	
Rückfallverzögert mit Zufalls-Sollwert						0	1	1	
Ansprechverzögert und Rückfallverzögert (zwei Zeitsollwerte)						1	0	0	
Ansprechverzögert und Rückfallverzögert, jeweils mit Zufalls-Sollwert (zwei Zeitsollwerte)						1	0	1	
Impulsgeber						1	1	0	
Blink-Relais (zwei Zeitsollwerte)						1	1	1	
Zeitbasis									
FB nicht benutzt				0	0				
Millisekunde: S				0	1				
Sekunde: M:S				1	0				
Minute: H:M				1	1				
Verwendung als Konstante und somit beschreibbar									
Zeitsollwert S1			0/1						
Zeitsollwert S2		0/1							

Beispiel:

Data 1 (Byte 5) = 0xAC

Bedeutung:

Die Werte erscheinen im Parametermenü. Der Timer wird im Modus des Impulsgebers mit der Zeitbasis „Sekunde“ benutzt. Der Zeitsollwert S1 ist mit einer Konstanten beschaltet und der Zeitsollwert S2 ist mit einer Variablen z. B. I7, I8 C2... usw. beschaltet.

Tabelle 78: Index 01 – Kontrollbyte

	Bit	7	6	5	4	3	2	1	0
FB-Ein-/Ausgang Data 3		–	–	–	ST ⁴⁾	RE ³⁾	EN ²⁾	Q1 ¹⁾	

- 1) Schaltkontakt
- 2) Enable, das Zeitrelais wird gestartet (Triggerspule)
- 3) Reset, das Zeitrelais wird zurückgesetzt (Resetspule)
- 4) Stopp, das Zeitrelais wird gestoppt (Stoppspule)

Beispiel:

Der Zeitsollwert 1 soll gelesen werden:

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Befehl: Lesen	89	–
	Antwort: Lesen erfolgreich	–	C2
1	Type	8E	8E
2	Instanz	00	00
3	Index	03	03
4	Data 1	00	4C
5	Data 2	00	06
6	Data 3	00	00
7	Data 4	00	00

Erklärung:

Data 1 = 4C

Data 2 = 06

→ resultierender 16-Bit-Wert = 064C_{hex} = 1612_{dez}

Bedeutung je nach eingestellter Zeitbasis:

Millisekunde	S	16120 ms	16,120 s
Sekunde	M:S	1620 s	26:52 Minuten
Minute	H:M	1612 min	67:04 Stunden

Jahresschaltuhr: Y1 - Y8**Telegrammaufbau**

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl:		
	Lesen	89	–
	Schreiben	8D	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Type	A2	A2
3	Instanz ²⁾	00 - 07	00 - 07
4	Index	→ Tabelle 79	
5 - 8	Data 1 - 4	abhängig vom Index, → Tabelle 80	

1) Mögliche Ursachen → Seite 150

2) easy stellt 8 Jahresschaltuhren Y1 bis Y8 zur Verfügung. Diese können über die Instanz (0 - 7) angesprochen werden.

Tabelle 79: Operandenübersicht

Index (hex)	Operand	Lesen	Schreiben
00	Parameter → Tabelle 80	×	
01	Kontrollbyte → Tabelle 81	×	
	Kanal A	×	c ¹⁾
11	Zeitpunkt ein	×	c ¹⁾
12	Zeitpunkt aus	×	c ¹⁾
	Kanal B	×	c ¹⁾
21	Zeitpunkt ein	×	c ¹⁾
22	Zeitpunkt aus	×	c ¹⁾
	Kanal C	×	c ¹⁾
31	Zeitpunkt ein	×	c ¹⁾
32	Zeitpunkt aus	×	c ¹⁾
	Kanal D	×	c ¹⁾
41	Zeitpunkt ein	×	c ¹⁾
42	Zeitpunkt aus	×	c ¹⁾

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 80: Index 00 – Parameter

Bedeutung	Bit	7	6	5	4	3	2	1	0
Erscheint im Parametermenü									
Kanal A									0/1
Kanal B								0/1	
Kanal C							0/1		
Kanal D						0/1			
Nicht benutzte Bits		–	–	–	–				

Beispiel:

Data 1 (Byte 5) = 0x03 → Die Werte der Jahresschaltuhr von Kanal A und B erscheinen im Parametermenü.

Tabelle 81: Index 01 – Kontrollbyte

Data 1	Bit	7	6	5	4	3	2	1	0
		FB-Ausgang	–	–	–	–	–	–	–

1) Zustand „1“, wenn die Zählbedingung erfüllt ist.

Kanal A, Index 11/12

Index 0x11 Kanal A Zeitpunkt des Einschaltens

Index 0x12 Kanal A Zeitpunkt des Ausschaltens

Data 1 (Byte 5) – Tag

Data 2 (Byte 6) – Monat

Data 3 (Byte 7) – Jahr

Beispiel:

Die Jahresschaltuhr Kanal A soll am 21.04.2004 eingeschaltet werden.

Index = 0x11

Data 1 = 0x15

Data 2 = 0x04

Data 3 = 0x04

Die Jahresschaltuhr Kanal B soll am 05.11.2012 ausgeschaltet werden.

Index = 0x22

Data 1 = 0x05

Data 2 = 0x0B

Data 3 = 0x0C

Wochenschaltuhr: 01 - 08

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 102	
1	Befehl:		
	Lesen	89	–
	Schreiben	8D	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0 ¹⁾
2	Type	A1	A1
3	Instanz ²⁾	00 - 07	00 - 07
4	Index	→ Tabelle 82	→ Tabelle 82
5 - 8	Data 1 - 4	abhängig vom Index, → Tabelle 83	

1) Mögliche Ursachen → Seite 150

2) easy stellt 8 Wochenschaltuhren 01 bis 08 zur Verfügung. Diese können über die Instanz (0 - 7) angesprochen werden.

Tabelle 82: Operandenübersicht

Index (hex)	Operand	Lesen	Schreiben
00	Parameter → Tabelle 83	×	
01	Kontrollbyte → Tabelle 84	×	
11	Kanal A Tag an/aus	×	c ¹⁾
12	Zeit ein	×	c ¹⁾
13	Zeit aus	×	c ¹⁾
21	Kanal B Tag an/aus	×	c ¹⁾
22	Zeit ein	×	c ¹⁾
23	Zeit aus	×	c ¹⁾
31	Kanal C Tag an/aus	×	c ¹⁾
32	Zeit ein	×	c ¹⁾
33	Zeit aus	×	c ¹⁾
41	Kanal D Tag an/aus	×	c ¹⁾
42	Zeit ein	×	c ¹⁾
43	Zeit aus	×	c ¹⁾

- 1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.
- 2) In den Datenbytes Data 1 - Data 4 wird ein 16-Bit-Wert übergeben. Dabei ist zu beachten, dass das Low-Byte in Data 1 und das High-Byte in Data 2 geführt wird.

Tabelle 83: Index 00 – Parameter

Bedeutung	Bit	7	6	5	4	3	2	1	0
Erscheint im Parametermenü									
Kanal A									0/1
Kanal B								0/1	
Kanal C							0/1		
Kanal D						0/1			
Nicht benutzte Bits		–	–	–	–				

Beispiel:

Data 1 (Byte 5) = 0x03

Bedeutung:

Die Werte der Wochenschaltuhr WH... von Kanal A und B erscheinen im Parametermenü.

Tabelle 84: Index 01 – Kontrollbyte

Data 1	Bit	7	6	5	4	3	2	1	0
FB-Ausgang		–	–	–	–	–	–	–	Q1 ¹⁾

1) Zustand „1“, wenn die Zählbedingung erfüllt ist.

Kanal A, Index 11/12/13

Index 0x11 Kanal A Wochentag an/aus

Data 1 (Byte 5) – Wochentag an

Data 2 (Byte 6) – Wochentag aus

0x01 = Sonntag...0x07 = Samstag

Wenn der Kanal nicht verwendet wird, ist der 16-Bit-Wert gleich 0x00.

Index 0x12 – Zeit ein (2 Byte)

Index 0x13 – Zeit aus (2 Byte)

Data 1 (Byte 5) – Stunde

Data 2 (Byte 6) – Minute

Beispiel: Zeit ein um 13:43 Uhr

Data 1 = 0x0D

Data 2 = 0x2B

Analyse – Fehlercodes über easyLink

Das Basisgerät easy700 liefert im Fall eines nicht korrekt gewählten Betriebsmodus oder eines ungültigen Telegramms einen definierten Fehlercode zurück. Die Übergabe des Fehlercodes ist wie folgt aufgebaut:

Telegrammaufbau

Byte	Bedeutung	Slave sendet Wert (hex)
0	Toggelbyte	→ Seite 102
1	Antwort	
	Befehl zurückgewiesen	C0
2	Type	00
3	Instanz	00
4	Index	00
5	Fehlercode	→ Tabelle 85

Tabelle 85: Fehlercodes

Fehlercode	Beschreibung
0x01	Es wurde ein unbekanntes Telegramm versendet.
0x02	Es wurde ein unbekanntes Objekt versendet.
0x03	Es wurde ein unbekanntes Kommando versendet.
0x04	Es wurde eine ungültige Instanz versendet.
0x05	Es wurde ein ungültiger Parametersatz verwendet.
0x06	Es wurde versucht, eine Variable zu beschreiben die keine Konstante ist.
0x0C	Das Gerät befindet sich in einem ungültigen Geräte- modus. STOP → RUN oder RUN → STOP
0x0D	Es erfolgte ein ungültiger Displayzugriff. Bitte verlassen Sie die Menü-Ebene, sodass die Status- Anzeige im Display angezeigt wird. Das Schreiben der Uhr ist nicht möglich.
0xF0	Es wurde versucht, einen unbekanntem Parameter anzusteuern.
0xF1	Nicht erlaubter Wert

9 Steuerbefehle easy800/MFD (DPV0)

Datenaustauschverfahren Das Modul „Steuerbefehle 9 Byte“ erlaubt den erweiterten Datenaustausch des easy800 und des MFD-Titan am Kommunikationsbus PROFIBUS-DP. Damit können Sie Dienste aus folgenden Bereichen übertragen:

- „Datum und Uhrzeit lesen/schreiben“ (Seite 156)
- „Abbilddaten lesen/schreiben“ (Seite 160) und
- „Funktionsbaustein-Daten lesen/schreiben“ (Seite 181).

Damit die Daten sicher vom Master zum Slave und umgekehrt über PROFIBUS-DP übertragen werden, ist ein Datenaustauschverfahren notwendig.



Achtung!

Während der Benutzung eines Steuerbefehls behalten die Ein- und Ausgangsdaten den Zustand vor Aufruf des Steuerbefehls. Erst wenn der Datenaustausch des Steuerbefehls beendet ist, sind die Ein-/Ausgangsdaten wieder aktuell.



Warnung!

Es dürfen nur die für den Befehlscode angegebenen Werte verwendet werden.

Überprüfen Sie die Werte, die Sie schreiben, um Fehlfunktionen zu vermeiden.

Voraussetzung:

Das entsprechende Modul „Steuerbefehle 9 Byte“ wurde gewählt.

Der Master löst den Datenaustausch der Steuerbefehle aus und der angesprochene Slave antwortet.

Bei der Kommunikation werden jeweils 9 Datenbytes (Byte 0 = Toggelbyte, Byte 1 bis 8 mit Informationsgehalt) auf dem PROFIBUS übergeben.

Der prinzipielle Aufbau des Telegramms ist in der nachfolgenden Abbildung dargestellt.

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
--------	--------	--------	--------	--------	--------	--------	--------	--------

Byte 0 – Toggelbyte

Das Byte 0 dient dazu, mit der Toggelfunktion das Senden eines Steuerbefehls zu aktivieren.

Bit	7	6	5	4	3	2	1	0
01 _{hex} /81 _{hex}	0/1	0	0	0	0	0	0	1
	Toggelbit	fix						

Vorgehensweise

- ▶ Um einen Befehl abzusetzen, wird das Bit 7 getoggelt; d. h. von 1 auf 0 oder von 0 auf 1 gesetzt.
- ▶ Pollen Sie anschließend so lange auf das Togglebit der Antwort des Koppelmoduls, bis es den gleichen Zustand wie das gesendete Togglebit angenommen hat. Dieser Zustand zeigt der Mastersteuerung an, dass die Antwort auf den abgesetzten Befehl gültig ist.
- ▶ Senden Sie einen neuen Befehl erst ab, wenn Sie eine Antwort erhalten haben (Ändern des Togglebits); sonst wird eventuell die Antwort des vorherigen Befehls überschrieben, bevor diese gelesen werden kann.



Wenn Sie Ein-/Ausgangsdaten und Steuerbefehle gleichzeitig benutzen, gilt:

Wenn der Datenaustausch von Steuerbefehlen beendet ist, werden die Bytes der Ein-Ausgänge wieder aktualisiert.

Alle angegebenen Kommandos und Parameter müssen im hexadezimalen Zahlenformat übergeben werden.

In den nachfolgenden Tabellen sind die verschiedenen Steuerbefehle zusammengefasst. Hierbei werden drei wesentliche Rubriken unterschieden – Echtzeituhr, Abbild und Funktionsbausteine.

Versionsgeschichte

Die nachfolgende Tabelle gibt eine Übersicht über die Änderungen und Neuerungen bei den verschiedenen Geräteversionen des easy800:

Auswirkung auf easyLink	easy800, Geräteversion			
	ab 02	ab 04	ab 05	ab 07
Unterstützung des kompletten PDO-Zugriffs				
R-Daten sind schreibbar	✓	✓	✓	✓
S-Daten sind lesbar	✓	✓	✓	✓
Unterstützung des kompletten SDO-Zugriffs				
Funktionsbausteine	–	A, AR, BC, BT, BV, C, CF, CH, CI, CP, D, DB, DC, FT, GT, HW, HY, LS, MR, NC, OT, PT, PW, SC, ST, T, VC		
	–	–	–	DG, JC, MX, PO, SP, SR, TB
Abbilddaten				
Lesen	–	IW, IA, ID, QW, QA, P, RW, SW, M, MB, MW, MD		
Schreiben	–	QW, QA, M, MB, MW, MD	M, MB, MW, MD	
Uhrenfunktionalität				
Regel-Option bei der Winter-/ Sommerzeitumstellung	–	✓	✓	✓
	–	–	✓	✓

**Datum und Uhrzeit
lesen/schreiben**



Beachten Sie bitte auch die entsprechende Beschreibung zur Echtzeituhr, die im easy800-Handbuch angeführt sind.

Die aktuelle Ausgabe des Handbuchs ist als PDF-Datei im Internet verfügbar: <http://www.eaton.com/moeller> → Support → Suchbegriff: MN04902001Z-DE

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl		
	Lesen	93	–
	Schreiben	B3	–
	Antwort		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Len	05	05
3	Index	00	00
4 - 8	Data 1 - 5		
	Beim Lesen	00	→ Tabelle 86
	Beim Schreiben	→ Tabelle 86	00

Tabelle 86: Byte 4 - 8: Data 1 - 5

Byte	Inhalt	Wert (hex)
4	Data 1 Stunde (0 bis 23)	00 - 17
5	Data 2 Minute (0 bis 59)	00 - 3B
6	Data 3 Tag (1 bis 28; 29, 30, 31; abhängig von Monat und Jahr)	01 - 1F
7	Data 4 Monat (1 bis 12)	01 - 0C
8	Data 5 Jahr (0 - 99, entspricht 2000 - 2099)	00 - 63

Winter-/Sommerzeit, Zeitumstellung

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl		
	Lesen	93	–
	Schreiben	B3	–
	Antwort		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Len	05	05
3	Index	01	01
4 - 8	Data 1 - 5		
	Beim Lesen	00	→ Tabelle 87
	Beim Schreiben	→ Tabelle 87	00

Tabelle 87: Byte 4 - 8: Data 1 - 5

Byte	Inhalt	Wert (hex)	
4	Data 1 Area	keine	00
		manuell	01
		automatische EU	02
		automatische GB	03
		automatische US	04
5	Data 2 ¹⁾	Set Sommerzeit Tag (1 bis 28, 29, 30, 31 abhängig von Monat und Jahr)	00 - 3B
6	Data 3 ¹⁾	Set Sommerzeit Monat (1 bis 12)	01 - 1F
7	Data 4 ¹⁾	Set Winterzeit Tag (1 bis 28, 29, 30, 31 abhängig von Monat und Jahr)	01 - 0C
8	Data 5 ¹⁾	Set Winterzeit Monat (1 bis 12)	00 - 63

1) Die erweiterten Parameter Data 2 bis Data 5 zur automatischen Winter-/Sommerzeitumstellung sind nur relevant, wenn Sie für Data 1 den Parameter „manuell“ ausgewählt haben.

Beispiel

Die Echtzeituhr der easy800 soll auf Freitag 23.05.2003, 14:36 Uhr, gestellt werden.

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	81	80
1	Befehl: Schreiben	B3	–
	Antwort: Schreiben erfolgreich	–	C1
2	Len	05	05
3	Index	00	00
4	Data 1	0E	00
5	Data 2	24	00
6	Data 3	17	00
7	Data 4	05	00
8	Data 5	03	00



Alle Werte müssen hexadezimal übergeben werden.

Abbilddaten lesen/schreiben



Beachten Sie bitte auch die entsprechende Beschreibung zu den möglichen Abbilddaten, die im easy800-Handbuch oder in der easySoft-Hilfe angeführt sind.

Die aktuelle Ausgabe des Handbuches ist als PDF-Datei im Internet verfügbar: <http://www.eaton.com/moeller> → Support → Suchbegriff: MN04902001Z-DE

Desweiteren gilt der Abschnitt „Generelles zum Arbeiten mit Abbilddaten“ auf Seite 64 auch für easy800.

Übersicht

Operanden	Bedeutung	Lesen/ schreiben	Befehl	Seite
IW0	„Lokale Eingänge IW0 lesen“	lesen	01	161
IW1 - IW8	„Eingänge der Netzwerkteilnehmer IW1 bis IW8 lesen“	lesen	01	163
IA1 - IA4	„Lokale Analog-Eingänge IA1 bis IA4 lesen“	lesen	02	164
ID1 - ID16	„Lokale Diagnose ID1 bis ID16 lesen“	lesen	03	166
QW0, QW1 - QW8	„Lokale Ausgänge QW0/Ausgänge der Netzwerkteilnehmer QW1 bis QW8 lesen und schreiben“	lesen/ schreiben	04	168
QA1	„Lokalen Analog-Ausgang QA1 lesen und schreiben“	lesen/ schreiben	05	170
P1 - P4	„Lokale P-Tasten lesen“	lesen	06	171
R1 - R16 S1 - S8	„Eingänge RW.../Ausgänge SW... von easy-Link lesen“	lesen	07/09	173
RN1 - RN32 SN1 - SN32	„Receive-Data Netz RN1...RN32/ Send-Data Netz SN1...SN32 lesen“	lesen	08/0A	175
M...	„Merker lesen und schreiben“	lesen/ schreiben	0B - 0E	177

Lokale Eingänge IWO lesen

Mit diesem Befehlsstring können Sie die lokalen Eingänge des Basisgerätes easy800 bzw. MFD auslesen. Das entsprechende Eingangswort ist hierbei im Intel-Format abgelegt.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl: Lesen	91	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0
2	Len	02	02
3	Type	01	01
4	Index	00	00
5	Data 1 (Low Byte)	00	→ Tabelle 88
6	Data 2 (High Byte)	00	→ Tabelle 88
7 - 8	Data 3 - 4	00	00

Tabelle 88: Byte 5 bis 6: Data 1 bis 2

Data 1	Bit	7	6	5	4	3	2	1	0
I1									0/1
I2								0/1	
...					...				
I8			0/1						
Data 2	Bit	7	6	5	4	3	2	1	0
I9									0/1
I10								0/1	
...					...				
I16			0/1						

Beispiel: Lokale Eingänge IW0 lesen

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	81	80
1	Befehl: Lesen	91	–
	Antwort: Lesen erfolgreich	–	C2
2	Len	02	02
3	Type	01	01
4	Index	00	00
5	Data 1	00	C4
6	Data 2	00	02
7	Data 3	00	00
8	Data 4	00	00



Alle Werte müssen hexadezimal übergeben werden.

Durch die Werte Data 1 = C4 und Data 2 = 02 wird angezeigt, dass die Eingänge I8, I7, I3 und I10 mit Zustand „1“ beschaltet sind.

Eingänge der Netzwerkteilnehmer IW1 bis IW8 lesen

Die easy800 und MFD- Geräte können Sie mit Hilfe des EASYNET sehr einfach dezentral erweitern. Mit dem hier angebotenen Dienst ist der lesende Zugriff auf die Eingänge der einzelnen NET-Teilnehmer realisierbar.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl: Lesen	91	–
	Antwort: Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0
2	Len	02	02
3	Type	01	01
4	Index	01 - 08 ¹⁾	01 - 08 ¹⁾
5	Data 1 (Low Byte)	00	→ Tabelle 88 auf Seite 161.
6	Data 2 (High Byte)	00	
7 - 8	Data 3 - 4	00	00

1) entspricht Adresse des Netzteilnehmers

Lokale Analog-Eingänge IA1 bis IA4 lesen

Die am Basisgerät easy800 bzw. MFD vorhandenen Analog-eingänge können Sie direkt über den PROFIBUS-DP auslesen. Hierbei wird der 16-Bitwert im Intelformat übergeben (LowByte first).

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl: Lesen	91	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0
2	Len	02	02
3	Type	02	02
4	Index	01 - 04 ¹⁾	01 - 04 ¹⁾
5	Data 1 (Low Byte)	00	siehe Beispiel
6	Data 2 (High Byte)	00	siehe Beispiel
7 - 8	Data 3 - 4	00	00

- 1) 01 = Analogeingang I7
 02 = Analogeingang I8
 03 = Analogeingang I11
 04 = Analogeingang I12

Beispiel

Es liegt ein Spannungspegel am Analogeingang 1 an. Die entsprechenden Telegramme zum Lesen des Analogwertes sehen wie folgt aus:

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	81	80
1	Befehl: Lesen	91	–
	Antwort: Lesen erfolgreich	–	C2
2	Len	02	02
3	Type	02	02
4	Index	01 ¹⁾	01 ¹⁾
5	Data 1	00	D9
6	Data 2	00	02
7	Data 3	00	00
8	Data 4	00	00

1) 01 = Analogeingang 1

Byte 5 – Data 1 (Low Byte): D9_{hex}

Byte 6 – Data 2 (High Byte): 02_{hex}

→ entsprechender 16-Bitwert: 02D9_{hex} = 729 (7,29 V)

Lokale Diagnose ID1 bis ID16 lesen

Die lokale Diagnose (ID1 - ID8) gibt den Zustand der einzelnen NET-Teilnehmer an. Die Verbindung zum Remote-Teilnehmer (nur MFD) wird über ID9 angezeigt.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl: Lesen	91	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0
2	Len	02	02
3	Type	03	03
4	Index	00	00
5	Data 1 (Low Byte)	00	→ Tabelle 89
6	Data 2 (High Byte)	00	→ Tabelle 89
7 - 8	Data 3 - 4	00	00

Tabelle 89: Byte 5 bis 6: Data 1 bis 2

Data 1	Bit	7	6	5	4	3	2	1	0
ID1									0/1
ID2									0/1
...					...				
ID8			0/1						
Data 2	Bit	7	6	5	4	3	2	1	0
ID9									0/1
–									1
...					...				
–			1						

0/1= aktiver/inaktiver NET-Teilnehmer, -- nicht belegt

Beispiel

Data 1 = F8, Data 2 = FF → Im easyNet sind die drei Teilnehmer mit der NET-ID 1, 2, 3 vorhanden.

Lokale Ausgänge QW0/Ausgänge der Netzwerkteilnehmer QW1 bis QW8 lesen und schreiben

Die lokalen Ausgänge können Sie über den PROFIBUS-DP direkt lesen und beschreiben. Die Ausgänge werden jedoch nur nach außen durchgeschaltet, wenn sich das Gerät im „Run“-Modus befindet und der angesprochene Ausgang nicht im Schaltplan verwendet wird. → Abschnitt „Abbildaten lesen/schreiben“ auf Seite 160.

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl		
	Lesen	91	–
	Schreiben	B1	–
	Antwort		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Len	02	02
3	Type	04	04
4	Index ¹⁾	00/01 - 08	00/01 - 08
5	Data 1		
	Beim Lesen	00	→ Tabelle 89
	Beim Schreiben	→ Tabelle 90	00
6 - 8	Data 2 - 4	00	00

1) 00 = Lokaler Ausgang

01 - 08 = Ausgänge der Netzwerkteilnehmer 1 - 8

Tabelle 90: Byte5: Data

Data 1	Bit 7	6	5	4	3	2	1	0
Q1								0/1
Q2							0/1	
Q3						0/1		
Q4					0/1			
Q5				0				
Q6			0					
Q7		0						
Q8	0							

Lokalen Analog-Ausgang QA1 lesen und schreiben

Auf den lokalen Analogausgang des Basisgerätes easy800 oder MFD können Sie mit Hilfe des angebotenen Kommandos zugreifen. Beim schreibenden Zugriff auf den Analogausgang wird der Wert allerdings nur nach außen gegeben, wenn das entsprechende Gerät im „Run“-Modus ist und das entsprechende Abbild nicht durch das eigentliche Programm beschrieben wird. → Abschnitt „Abbilddaten lesen/schreiben“ auf Seite 160.

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl		
	Lesen	91	–
	Schreiben	B1	–
	Antwort		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Len	02	02
3	Type	05	05
4	Index	00	00
5 - 6	Data 1 - 2		
	Beim Lesen	00	siehe Beispiel
	Beim Schreiben	siehe Beispiel	00
7 - 8	Data 3 - 4	00	00

Beispiel

Der Analogausgang soll einen Wert von ca. 5 V ausgeben.

500 = 01F4_{hex} Byte 5 – Data 1 (LowByte) : F4_{hex}
 Byte 6 – Data 2 (HighByte): 01_{hex}

Lokale P-Tasten lesen

Die lokalen P-Tasten sind die Display-Courser-Tasten des easy800-/MFD-Basisgerätes. Die Tasten können Sie sowohl im Modus „Run“ als auch im Modus „Stop“ abfragen.



Achten Sie nur darauf, dass die P-Tasten auch unter dem Menüpunkt SYSTEM (im Basisgerät) aktiviert sind.

Bei den P-Tasten müssen Sie nur ein Byte übergeben.

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl: Lesen	91	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0
2	Len	02	02
3	Type	06	06
4	Index	00	00
5	Data 1 (Low Byte)	00	→ Tabelle 91
6 - 8	Data 2 - 4	00	00

Tabelle 91: Byte 5: Data

Data 1	Bit 7	6	5	4	3	2	1	0
P1								0/1
P2								0/1
P3								0/1
P4								0/1
-					0			
-				0				
-			0					
-		0						

Eingänge RW.../Ausgänge SW... von easyLink lesen

Mit diesem Dienst können Sie die lokalen R- und S-Daten und die der NET-Teilnehmer (1 - 8) , die über den easyLink transferiert werden, nochmals aus dem entsprechenden Abbild der easy800/MFD lesen.

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl: Lesen	91	–
	Antwort: Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0
2	Len	02	02
3	Type	bei RW: 07	bei RW: 07
		bei SW: 09	bei SW: 09
4	Index	00/01 - 08 ¹⁾	00/01 - 08 ¹⁾
5	Data 1 (Low Byte)	00	→ Tabelle 92
6	Data 2 (High Byte)	00	→ Tabelle 92
7 - 8	Data 3 - 4	00	00

1) 00 = Lokaler Ein-/Ausgang

01 - 08 = Adresse des Netzwerkteilnehmers (NET-ID 1 - 8)

Tabelle 92: Byte 5 bis 6: Data 1 bis 2

Data 1		Bit	7	6	5	4	3	2	1	0
RW	SW									
R1	S1									0/1
R2	S2									0/1
R3	S3									0/1
R4	S4									0/1
R5	S5									0/1
R6	S6									0/1
R7	S7									0/1
R8	S8									0/1
Data 2		Bit	7	6	5	4	3	2	1	0
R9	–									0/1
R10	–									0/1
R11	–									0/1
R12	–									0/1
R13	–									0/1
R14	–									0/1
R15	–									0/1
R16	–									0/1

**Receive-Data Netz RN1...RN32/
Send-Data Netz SN1...SN32 lesen**

Das EASYNET bietet die Möglichkeit, eine Punkt-zu-Punkt-Verbindung zwischen den einzelnen NET-Teilnehmern zu realisieren. Der Datenaustausch erfolgt hierbei mittels RN-SN-Daten (siehe easy800-Handbuch).



Es können keine RN-SN-Daten des lokalen Gerätes (Index = 0) , an dem sich das EASY204-DP befindet, abgefragt werden. In diesem Fall würde der Befehl mit 0C_{hex} zurückgewiesen.

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Togglebyte	→ Seite 154	
1	Befehl: Lesen	91	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0
2	Len	04	04
3	Type	bei RN1 - RN32: 08	
		bei SN1 - SN32: 0A	
4	Index	01 - 08 ¹⁾	01 - 08 ¹⁾
5 - 8	Data 1 - 4	00	→ Tabelle 93

1) entspricht NET-ID

Tabelle 93: Byte 5 bis 8: Data 1 bis 4

Data 1	Bit	7	6	5	4	3	2	1	0
RN1 SN1					...				0/1
...									0/1
RN8 SN8		0/1							
Data 2	Bit	7	6	5	4	3	2	1	0
RN9 SN9									0/1
...					...				
RN16 SN16		0/1							
Data 3	Bit	7	6	5	4	3	2	1	0
RN17 SN17									0/1
...					...				
RN24 SN24		0/1							
Data 4	Bit	7	6	5	4	3	2	1	0
RN25 SN25									0/1
...					...				
RN32 SN32		0/1							

Merker lesen und schreiben

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl		
	Lesen	91	–
	Schreiben	B1	–
	Antwort		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Len	→ Tabelle 94	→ Tabelle 94
3	Type		
4	Index		
5 - 8	Data 1 - 4		
	Beim Lesen	00	→ „Beispiel 1“ auf Seite 179
	Beim Schreiben	→ „Beispiel 2“ auf Seite 180	00

Tabelle 94: Byte 2 - 4: Len, Type, Index

Operand		Len	Type	Index
Merker-Bit	M1 ... M96	01 _{hex}	0B _{hex}	01 bis 60 _{hex}
Merker-Byte	MB1 ... MB96	01 _{hex}	0C _{hex}	01 bis 60 _{hex}
Merker-Word	MW1 ... MW96	02 _{hex}	0D _{hex}	01 bis 60 _{hex}
Merker-Doppelwort	MD1 ... MD96	04 _{hex}	0E _{hex}	01 bis 60 _{hex}

Sehen Sie sich hierzu eventuell die Merkeraufteilung im easy800-Handbuch etwas detaillierter an. An dieser Stelle soll lediglich ein kleiner Auszug aus dem genannten Handbuch dargestellt werden, um die prinzipielle Aufteilung einmal kurz darzustellen.



Achtung!

Die Funktionsbausteine und DW-Merker (32Bit-Werte) der easy800/MFD arbeiten mit vorzeichenbehafteten Werten.

Gilt für MD, MW, MB, M	Links = Größtwertiges Bit, Byte, Wort			Rechts = Kleinstwertiges Bit, Byte, Wort
32 Bit	MD1			
16 Bit	MW2		MW1	
8 Bit	MB4	MB3	MB2	MB1
1 Bit	M32 bis M25	M24 bis M17	M16 bis M9	M8 bis M1
32 Bit	MD2			
16 Bit	MW4		MW3	
8 Bit	MB8	MB7	MB6	MB5
1 Bit	M64 bis M57	M56 bis M49	M48 bis M41	M40 bis M33



Die entsprechenden Merkerwerte werden im Intelformat übergeben; das erste Byte ist dementsprechend das Low Byte (Byte 5) und das letzte demzufolge das High Byte.

Beispiel 1

Merker-Bit M62 lesen

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	81	80
1	Befehl: Lesen	91	–
	Antwort: Lesen erfolgreich	–	C2
2	Len	01	01
3	Type	0B	0B
4	Index	3E	3E
5	Data 1	00	01
6	Data 2	00	00
7	Data 3	00	00
8	Data 4	00	00

Ergebnis: Data 1 = 01_{hex} → M62 wurde gesetzt

Beispiel 2

Merkerwort MW32 mit 823 beschreiben

$823_{\text{dec}} = 337_{\text{hex}} \rightarrow \text{Data 1} = 37_{\text{hex}}, \text{Data 2} = 03_{\text{hex}}$

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	81	80
1	Befehl: Schreiben	B1	–
	Antwort: Schreiben erfolgreich	–	C1
2	Len	02	02
3	Type	0D	0D
4	Index	20	20
5	Data 1	37	00
6	Data 2	03	00
7	Data 3	00	00
8	Data 4	00	00

**Funktionsbaustein-Daten
lesen/schreiben**

Beachten Sie bitte auch die entsprechende Beschreibung zu den Funktionsbausteinen, die im easy800-Handbuch angeführt sind.

Die aktuelle Ausgabe des Handbuches ist als PDF-Datei im Internet verfügbar: <http://www.eaton.com/moeller> → Support → Suchbegriff: MN04902001Z-DE

Allgemeine Hinweise

Grundsätzlich müssen Sie beim Arbeiten mit den Funktionsbausteinen Folgendes beachten:

- Die entsprechenden Daten werden im Intelformat übergeben: das erste Byte ist das Low Byte (Byte 5) und das letzte das High Byte (Byte 8).
- Die Datenlänge beträgt bis zu 4 Byte. Alle Werte müssen Sie im hexadezimalen Zahlenformat übergeben.
- Alle 32-Bit-Werte werden vorzeichenbehaftet behandelt. Wenn Sie 32-Bit-Werte übergeben, achten Sie darauf, dass der entsprechende Wertebereich dem long integer entspricht, also vorzeichenbehaftet ist.
32-Bit-Wert: $-2\,147\,483\,648 \dots 0 \dots +2\,147\,483\,647$

Übersicht

Operanden	Bedeutung	Lesen/ schreiben	Type (hex)	Seite
A01 - A32	„Analogwertvergleich A01...A32“	lesen/schreiben	11	184
AR01 - AR32	„Arithmetikbaustein AR01...AR32“	lesen/schreiben	12	186
BC01 - BC32	„Block Compare BC01...BC32“	lesen/schreiben	25	188
BT01 - BT32	„Block Transfer BT01...BT32“	lesen/schreiben	26	190
BV01 - BV32	„Boolesche Verknüpfung BV01...BV32“	lesen/schreiben	13	192
C01 - C32	„Zähler C01...C32“	lesen/schreiben	14	194
CF01 - CF04	„Frequenzzähler CF01...CF04“	lesen/schreiben	15	196
CH01 - CH04	„Schneller Zähler CH01...CH04“	lesen/schreiben	16	198
CI01 - CI02	„Inkrementalzähler CI01...CI02“	lesen/schreiben	17	200
CP01 - CP32	„Vergleicher CP01...CP32“	lesen/schreiben	18	202
D01 - D32	„Textausgabe-Baustein D01...D32“	lesen/schreiben	19	204
DB01 - DB32	„Datenbaustein DB01...DB32“	lesen/schreiben	1A	207
DC01 - DC32	„PID-Regler DC01...DC32“	lesen/schreiben	27	209
DG01 - DG16	„Diagnose DG01...DG16“	lesen/schreiben	39	212
FT01 - FT32	„Signalglättungsfilter FT01...FT32“	lesen/schreiben	28	214
GT01 - GT32	„Empfang von Netzdaten GT01...GT32“	lesen	1B	216
HW01 - HW32	„Wochenzeitschaltuhr HW01...HW32“	lesen	1C	218
HY01 - HY32	„Jahreszeitschaltuhr HY01...HY32“	lesen	1D	221
JC01 - JC32	„Bedingter Sprung JC01...JC32“	lesen	2F	224
LS01 - LS32	„Wertskalierung LS01...LS32“	lesen/schreiben	29	226
MR01 - MR32	„Masterreset MR01...MR32“	lesen	0F	228
MX01 - MX32	„Datenmultiplexer MX01...MX32“	lesen/schreiben	31	230
NC01 - NC32	„Zahlenwandler NC01...NC32“	lesen/schreiben	2A	232
OT01 - OT04	„Betriebsstundenzähler OT01...OT04“	lesen/schreiben	1E	234
PO01 - PO04	„Impulsausgabe PO01...PO02“	lesen/schreiben	32	236
PT01 - PT32	„Senden von Netzdaten PT01...PT32“	lesen	1F	239
PW01 - PW02	„Pulsweitenmodulation PW01...PW02“	lesen/schreiben	2B	241

Operanden	Bedeutung	Lesen/ schreiben	Type (hex)	Seite
SC01	„Uhr synchronisieren SC01“	lesen	20	243
SP01 - SP32	„Serielle Ausgabe SP01...SP32“	lesen	35	244
SR01 - SR32	„Schieberegister SR01...SR32“	lesen/schreiben	33	246
ST01	„Sollzykluszeit ST01“	lesen/schreiben	2C	248
T01 - T32	„Zeitrelais T01...T32“	lesen/schreiben	21	250
TB01 - TB32	„Tabellenfunktion TB01...TB32“	lesen/schreiben	32	253
VC01 - VC32	„Wertbegrenzung VC01...VC32“	lesen/schreiben	2D	255

Analogwertvergleichler A01...A32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	11	11
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 95	→ Tabelle 95
5 - 8	Data 1 - 4	00	abhängig vom Index, → Tabelle 96, 97

Tabelle 95: Operandenübersicht

Index (hex)	Operand		lesen	schreiben
00	Bit-IO, → Tabelle 96		×	
01	Mode, → Tabelle 97		×	
02	Vergleichswert 1 I1	I1	×	c ¹⁾
03	Verstärkungsfaktor für I1 (I1 = F1 × Wert) F1	F1	×	c ¹⁾
04	Vergleichswert 2 I2	I2	×	c ¹⁾
05	Verstärkungsfaktor für I2 (I2 = F2 × Wert) F2	F2	×	c ¹⁾
06	Offset für den Wert I1 OS	OS	×	c ¹⁾
07	Schalthysterese für Wert I2 (Wert HY gilt sowohl für die positive als auch negative Hysterese.) HY	HY	×	c ¹⁾

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.



Die Daten für Index 2 bis 7 werden als 32-Bit-Wert im Intelformat (Data 1 – Low Byte bis Data 4 – High Byte) übergeben.

Tabelle 96: Index 0: Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		–	–	–	–	–	–	CY ¹⁾	Q1 ²⁾

- 1) Zustand „1“, wenn der Wertebereich überschritten wird
- 2) Zustand „1“, wenn die Bedingung erfüllt ist (z. B. I1 < I2 bei der Betriebsart LT)

Tabelle 97: Index 1 – Mode

Data 1 (hex)		
00	LT	kleiner (I1 < I2)
01	EQ	gleich (I1 = IGT)
02	GT	größer (I1 > I2)

Arithmetikbaustein AR01...AR32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	12	12
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 98	→ Tabelle 98
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 99, 100
	beim Schreiben	abhängig vom Index, → Tabelle 99, 100	00

Tabelle 98: Operandenübersicht

Index (hex)	Operand		lesen	schreiben
00	Bit-IO, → Tabelle 99		×	
01	Mode, → Tabelle 100		×	
02	erster Operand	I1	×	c ¹⁾
03	zweiter Operand	I2	×	c ¹⁾
04	Ergebnis	QV	×	

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.



Die Daten für Index 2 bis 4 werden als 32-Bit-Wert im Intelformat (Data 1 – Low Byte bis Data 4 – High Byte) übergeben.

Tabelle 99: Index 0: Bit-IO

	Bit	7	6	5	4	3	2	1
FB-Ausgang Data 3		–	–	–	–	–	ZE ¹⁾	CY ¹⁾

- 1) Zustand „1“, wenn der Wert des Bausteinausgangs QV (also das Rechenergebnis) gleich Null ist
- 2) Zustand „1“, wenn der Wertebereich überschritten wird

Tabelle 100: Index 1 – Mode

Data 1 (hex)		
00	ADD	Addieren (I1 + I2 = QV)
01	SUB	Subtrahieren (I1 - I2 = QV)
02	MUL	Multiplizieren (I1 × I2 = QV)
03	DIV	Dividieren (I1 : I2 = QV)

Block Compare BC01...BC32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	→ Seite 154
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	25	25
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 101	→ Tabelle 101
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 102, 103
	beim Schreiben	abhängig vom Index, → Tabelle 102, 103	00

Tabelle 101: Operandenübersicht

Index (hex)	Operand	lesen	schreiben
00	Bit-IO, → Tabelle 102	×	
01	Mode, → Tabelle 103	×	
02	Quellbereich 1 11	×	c ¹⁾
03	Zielbereich 2 12	×	c ¹⁾
04	Anzahl der zu vergleichenden Elemente: 8 (max. 192 Byte) NO	×	c ¹⁾

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.



Die Daten für Index 2 bis 4 werden als 32-Bit-Wert im Intelformat (Data 1 – Low Byte bis Data 4 – High Byte) übergeben.

Tabelle 102: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	EN ¹⁾
FB-Ausgang Data 3		–	–	–	–	EQ ²⁾	E3 ³⁾	E2 ⁴⁾	E1 ⁵⁾

- 1) Aktiviert den Baustein bei Zustand „1“.
- 2) Zustand „1“, wenn die Datenbereiche gleich sind; Zustand „0“, wenn sie ungleich sind

Fehlerausgänge

- 3) Zustand „1“, wenn die Anzahl der Elemente den Quell- oder Zielbereich überschreitet.
- 4) Zustand „1“, wenn sich Quell- und Zielbereich überlappen.
- 5) Zustand „1“, wenn Quell- oder Zielbereich außerhalb des verfügbaren Merkerbereichs liegen (Offsetfehler).

Tabelle 103: Index 1 – Mode

Mode	Data 1 (hex)	Betriebsart
	02	Vergleichen (easy intern Zustandsanzeige Modus Block Compare)

Block Transfer BT01...BT32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	→ Seite 154
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	26	26
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 104	→ Tabelle 104
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 105, 106
	beim Schreiben	abhängig vom Index, → Tabelle 105, 106	00

Tabelle 104: Operandenübersicht

Index (hex)	Operand		lesen	schreiben
00	Bit-IO, → Tabelle 105		×	
01	Mode, → Tabelle 106		×	
02	Quellbereich 1	I1	×	c ¹⁾
03	Zielbereich 2	I2	×	c ¹⁾
04	Anzahl der zu vergleichenden Elemente: max. 192 Byte	NO	×	c ¹⁾

- 1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschriftet ist.



Die Daten für Index 2 und 3 werden als 32-Bit-Wert im Intelformat (Data 1 – LowByte...Data 2 – HighByte) übergeben.

Tabelle 105: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	1 ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	E3 ²⁾	E2 ³⁾	E1 ⁴⁾

1) Transfer von der an I1 angegebenen Quelladresse zu der an I2 angegebenen Zieladresse bei positiver Flanke.

Fehlerausgänge

- 2) Zustand „1“, wenn die Anzahl der Elemente den Quell- oder Zielbereich überschreitet.
- 3) Zustand „1“, wenn sich Quell- und Zielbereich überlappen.
- 4) Zustand „1“, wenn Quell- oder Zielbereich außerhalb des verfügbaren Merkerbereichs liegen (Offsetfehler).

Tabelle 106: Index 1 – Mode

Data 1 (hex)	Betriebsart
00	INI: Initialisiert den Zielbereich mit einem Bytewert, der unter der Quelladresse hinterlegt ist.
01	CPY: Kopiert einen Datenblock von einem Quell- zu einem Zielbereich. Über „NO“ geben Sie die Datenblockgröße vor.

Boolesche Verknüpfung BV01...BV32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	13	13
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 107	→ Tabelle 107
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 108, 109
	beim Schreiben	abhängig vom Index, → Tabelle 108, 109	00

Tabelle 107: Operandenübersicht

Index (hex)	Operand	lesen	schreiben
00	Bit-IO, → Tabelle 108	×	
01	Mode, → Tabelle 109	×	
02	erster Operand I1	×	c ¹⁾
03	zweiter Operand I2	×	c ¹⁾
04	Ergebnis der Verknüpfung QV	×	

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.



Die Daten für Index 2 bis 4 werden als 32-Bit-Wert im Intelformat (Data 1 – Low Byte bis Data 4 – High Byte) übergeben.

Tabelle 108: Index 0: Bit-IO

	Bit	7	6	5	4	3	2	1
FB-Ausgang Data 3		–	–	–	–	–	–	ZE ¹⁾

1) Zustand „1“, wenn der Wert des Bausteinausgangs QV (also das Ergebnis der Verknüpfung) gleich Null ist

Tabelle 109: Index 1 – Mode

Data 1 (hex)		
00	AND	Und-Verknüpfung
01	OR	Oder-Verknüpfung
02	XOR	Exklusiv-Oder-Verknüpfung
03	NET	Invertiert die einzelnen Bit des Wertes von I1. Der invertierte Wert wird vorzeichenbehaftet dezimal dargestellt.

Zähler C01...C32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	14	14
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 110	→ Tabelle 110
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 111
	beim Schreiben	abhängig vom Index, → Tabelle 111	00

Tabelle 110: Operandenübersicht

Index (hex)	Operand	Wert	lesen	schreiben
00	Bit-IO	→ Tabelle 111	×	
01	Mode/Parameter	–	–	–
02	oberer Sollwert SH	im ganzzahligen Bereich von –2 147 483 648 bis +2 147 483 647	×	c ¹⁾
03	unterer Sollwert SL		×	c ¹⁾
04	Vorgabe Istwert SV		×	c ¹⁾
05	Istwert im „Run“-Betrieb QV		×	

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.



Die Daten für Index 2 bis 5 werden als 32-Bit-Wert im Intelformat (Data 1 – Low Byte bis Data 4 – High Byte) übergeben.

Tabelle 111: Index 0: Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	SE ¹⁾	D ²⁾	C ³⁾	RE ⁴⁾
FB-Ausgang Data 3		–	–	–	–	ZE ⁵⁾	CY ⁶⁾	FB ⁷⁾	OF ⁸⁾

- 1) bei positiver Flanke Vorgabe-Istwert übernehmen
- 2) Zählrichtungsangabe: Zustand „0“ = vorwärts zählen, Zustand „1“ = rückwärts zählen
- 3) Zählspule, zählt bei jeder positiven Flanke
- 4) Rücksetzen des Istwertes auf Null
- 5) Zero: Zustand „1“, wenn der Wert des Bausteinganges QV (also der Zählerstand) gleich Null ist
- 6) Carry: Zustand „1“, wenn der Wertebereich überschritten wird.
- 7) Fall below: Zustand „1“, wenn Istwert \leq unterer Sollwert
- 8) Overflow: Zustand „1“, wenn Istwert \geq oberer Sollwert

Frequenzzähler CF01...CF04

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	15	15
3	Instanz	01 - 04	01 - 04
4	Index	→ Tabelle 112	→ Tabelle 112
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 113
	beim Schreiben	abhängig vom Index, → Tabelle 113	00

Tabelle 112: Operandenübersicht

Index (hex)	Operand	lesen	schreiben
00	Bit-IO, → Tabelle 113	×	
01	Mode/Parameter	–	–
02	oberer Sollwert SH	×	c ¹⁾
03	unterer Sollwert SL	×	c ¹⁾
04	Istwert im „Run“-Betrieb QV	×	

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.



Die Daten für Index 2 bis 4 werden als 32-Bit-Wert im Intelformat (Data 1 – Low Byte bis Data 4 – High Byte) übergeben.

Tabelle 113: Index 0: Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	EN ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	ZE ²⁾	FB ³⁾	OF ⁴⁾

- 1) Freigabe des Zählers
- 2) Zero: Zustand „1“, wenn der Wert des Bausteinausganges QV (also der Zählerstand) gleich Null ist
- 3) Fall below: Zustand „1“, wenn Istwert ≤ unterer Sollwert
- 4) Overflow: Zustand „1“, wenn Istwert ≥ oberer Sollwert.

Schneller Zähler CH01...CH04

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	16	16
3	Instanz	01 - 04	01 - 04
4	Index	→ Tabelle 114	→ Tabelle 114
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 114
	beim Schreiben	abhängig vom Index, → Tabelle 114	00

Tabelle 114: Operandenübersicht

Index (hex)	Operand	Wert	lesen	schreiben
00	Bit-IO	→ Tabelle 115	×	
01	Mode/Parameter	–	–	–
02	oberer Sollwert SH	im ganzzahligen Bereich von –2 147 483 648 bis +2 147 483 647	×	c ¹⁾
03	unterer Sollwert SL		×	c ¹⁾
04	Vorgabe Istwert SV		×	c ¹⁾
05	Istwert im „Run“-Betrieb QV		×	

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.



Die Daten für Index 2 bis 5 werden als 32-Bit-Wert im Intelformat (Data 1 – Low Byte bis Data 4 – High Byte) übergeben.

Tabelle 115: Index 0: Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	EN ¹⁾	SE ²⁾	D ³⁾	RE ⁴⁾
FB-Ausgang Data 3		–	–	–	–	ZE ⁵⁾	CY ⁶⁾	FB ⁷⁾	OF ⁸⁾

- 1) Freigabe des Zählers
- 2) bei positiver Flanke Vorgabe-Istwert übernehmen
- 3) Zählrichtungsangabe: Zustand „0“ = vorwärts zählen, Zustand „1“ = rückwärts zählen
- 4) Rücksetzen des Istwertes auf Null
- 5) Zero: Zustand „1“, wenn der Wert des Bausteinausganges QV (also der Zählerstand) gleich Null ist
- 6) Carry: Zustand „1“, wenn der Wertebereich überschritten wird.
- 7) Fall below: Zustand „1“, wenn Istwert ≤ unterer Sollwert
- 8) Overflow: Zustand „1“, wenn Istwert ≥ unterer Sollwert

Inkrementalzähler CI01...CI02

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	17	17
3	Instanz	01 - 02	01 - 02
4	Index	→ Tabelle 116	→ Tabelle 116
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 117
	beim Schreiben	abhängig vom Index, → Tabelle 117	00

Tabelle 116: Operandenübersicht

Index (hex)	Operand	Wert	lesen	schreiben
00	Bit-IO	→ Tabelle 117	×	
01	Mode/Parameter	–	–	–
02	oberer Sollwert SH	im ganzzahligen Bereich von –2 147 483 648 bis +2 147 483 647	×	c ¹⁾
03	unterer Sollwert SL		×	c ¹⁾
04	Vorgabe Istwert SV		×	c ¹⁾
05	Istwert im „Run“-Betrieb QV		×	

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.



Die Daten für Index 2 bis 5 werden als 32-Bit-Wert im Intelformat (Data 1 – Low Byte bis Data 4 – High Byte) übergeben.

Tabelle 117: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	EN ¹⁾	SE ²⁾	RE ³⁾
FB-Ausgang Data 3		–	–	–	–	ZE ⁴⁾	CY ⁵⁾	FB ⁶⁾	OF ⁷⁾

- 1) Freigabe des Zählers
- 2) Bei positiver Flanke Vorgabe-Istwert übernehmen
- 3) Rücksetzen des Istwertes auf Null
- 4) Zero: Zustand „1“, wenn der Wert des Bausteinganges QV (also der Zählerstand) gleich Null ist
- 5) Carry: Zustand „1“, wenn der Wertebereich überschritten wird.
- 6) Fall below: Zustand „1“, wenn Istwert ≤ unterer Sollwert
- 7) Overflow: Zustand „1“, wenn Istwert ≥ unterer Sollwert

Vergleicher CP01...CP32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	18	18
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 118	→ Tabelle 118
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 119
	beim Schreiben	abhängig vom Index, → Tabelle 119	00

Tabelle 118: Operandenübersicht

Index (hex)	Operand	lesen	schreiben
00	Bit-IO, → Tabelle 119	×	
01	Mode/Parameter	–	–
02	Vergleichswert I1	×	c ¹⁾
03	Vergleichswert I2	×	c ¹⁾

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.



Die Daten für Index 2 und 3 werden als 32-Bit-Wert im Intelformat (Data 1 – Low Byte bis Data 4 – High Byte) übergeben.

Tabelle 119: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1
FB-Ausgang Data 3		–	–	–	–	GT ¹⁾	EQ ²⁾	LT ³⁾

- 1) greater than: Zustand „1“, wenn der Wert an I1 größer als der Wert an I2 ist (I1 > I2)
- 2) equal: Zustand „1“, wenn der Wert an I1 gleich dem Wert an I2 ist (I1 = I2)
- 3) less than: Zustand „1“, wenn der Wert an I1 kleiner als der Wert an I2 ist (I1 < I2).

Textausgabe-Baustein D01...D32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	19	19
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 120	→ Tabelle 120
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 121
	beim Schreiben	abhängig vom Index, → Tabelle 121	00

Tabelle 120: Operandenübersicht

Index (hex)	Operand	lesen	schreiben
00	Bit-IO, → Tabelle 121	×	
01	Mode/Parameter	–	–
02	Text Zeile 1, Spalte 1 - 4	×	
03	Text Zeile 1, Spalte 5 - 8	×	
04	Text Zeile 1, Spalte 9 - 12	×	
05	Text Zeile 1, Spalte 13 - 16	×	
06	Text Zeile 2, Spalte 1 - 4	×	
07	Text Zeile 2, Spalte 5 - 8	×	
08	Text Zeile 2, Spalte 9 - 12	×	
09	Text Zeile 2, Spalte 13 - 16	×	
10	Text Zeile 3, Spalte 1 - 4	×	
11	Text Zeile 3, Spalte 5 - 8	×	
12	Text Zeile 3, Spalte 9 - 12	×	
13	Text Zeile 3, Spalte 13 - 16	×	
14	Text Zeile 4, Spalte 1 - 4	×	
15	Text Zeile 4, Spalte 5 - 8	×	
16	Text Zeile 4, Spalte 9 - 12	×	
17	Text Zeile 4, Spalte 13 - 16	×	
18	Variable 1	×	c ¹⁾
19	Variable 2	×	c ¹⁾
20	Variable 3	×	c ¹⁾
21	Variable 4	×	c ¹⁾
22	Skalierung Minimalwert 1	×	
23	Skalierung Minimalwert 2	×	
24	Skalierung Minimalwert 3	×	
25	Skalierung Minimalwert 4	×	
26	Skalierung Maximalwert 1	×	

Index (hex)	Operand	lesen	schreiben
27	Skalierung Maximalwert 2	×	
28	Skalierung Maximalwert 3	×	
29	Skalierung Maximalwert 4	×	
30	Steuerinformationen Zeile 1	×	
31	Steuerinformationen Zeile 2	×	
32	Steuerinformationen Zeile 3	×	
33	Steuerinformationen Zeile 4	×	

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.



Die Variablen 1 bis 4 (Index 18 bis 21) werden als 32-Bit-Wert im Intelformat (Data 1 – Low Byte bis Data 4 – High Byte) übergeben.

Tabelle 121: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	EN ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q1 ²⁾

1) Freigabe des Textbausteins

2) Zustand „1“, Textbaustein ist aktiv

Datenbaustein DB01...DB32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	1A	1A
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 122	→ Tabelle 122
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 123
	beim Schreiben	abhängig vom Index, → Tabelle 123	00

Tabelle 122: Operandenübersicht

Index (hex)	Operand	lesen	schreiben
00	Bit-IO, → Tabelle 123	×	
01	Mode/Parameter	–	–
02	Eingangswert: Wert, I1 der beim Triggern des Bausteins an den Ausgang QV weitergegeben wird.	×	c ¹⁾
03	Ausgangswert QV	×	

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.



Die Daten für Index 2 und 3 werden als 32-Bit-Wert im Intelformat (Data 1 – Low Byte bis Data 4 – High Byte) übergeben.

Tabelle 123: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	T ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q1 ²⁾

1) Übernahme des an I1 liegenden Wertes bei positiver Flanke.
2) Zustand „1“, wenn das Triggersignal den Zustand „1“ besitzt.

PID-Regler DC01...DC32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	→ Seite 154
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	27	27
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 124	→ Tabelle 124
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 125, 126
	beim Schreiben	abhängig vom Index, → Tabelle 125, 126	

Tabelle 124: Operandenübersicht

Index (hex)	Operand		lesen	schreiben
00	Bit-IO, → Tabelle 125		×	
01	Mode, → Tabelle 126		×	
02	Sollwert: -32 768 bis +32 767	I1	×	c ¹⁾
03	Istwert: -32 768 bis +32 767	I2	×	c ¹⁾
04	Proportionalverstärkung [%], Wertebereich: 0 bis 65 535	KP	×	c ¹⁾
05	Nachstellzeit [0,1 s], Wertebereich: 0 bis 65 535	TN	×	c ¹⁾
06	Vorhaltezeit [0,1 s], Wertebereich: 0 bis 65 535	TV	×	c ¹⁾
07	Abtastzeit = Zeit zwischen den Bausteinaufrufen. Wertebereich: 0.1s bis 6553.5s. Wenn Sie den Wert 0 angeben, wird die Abtastzeit von der Programmzykluszeit bestimmt.	TC	×	c ¹⁾
08	Handstellgröße, Wertebereich: -4096 bis +4095	MV	×	c ¹⁾
09	Stellgröße • Betriebsart: UNI, Wertebereich: 0 bis +4095 (12 Bit) • Betriebsart: BIP, Wertebereich: -4096 bis +4095 (13 Bit)	QV	×	

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.



Die Daten für Index 2 bis 9 werden als 32-Bit-Wert im Intelformat (Data 1 – LowByte...Data 2 – HighByte) übergeben.

Tabelle 125: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	SE ¹⁾	ED ²⁾	EI ³⁾	EP ⁴⁾	EN ⁵⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	LI ⁶⁾

- 1) Übernahme der Handstellgröße bei Zustand „1“
- 2) Aktivieren des D-Teils bei Zustand „1“
- 3) Aktivieren des I-Teils bei Zustand „1“
- 4) Aktivieren des P-Teils bei Zustand „1“
- 5) Aktiviert den Baustein bei Zustand „1“.
- 6) Zustand „1“, wenn der Wertebereich der Stellgröße überschritten wurde

Tabelle 126: Index 1 – Mode

Data 1	Betriebsart
UNP unipolar	Die Stellgröße wird als unipolarer 12Bit-Wert ausgegeben. Entsprechender Wertebereich für QV 0 bis 4095.
BIP bipolar	Die Stellgröße wird als bipolarer 13Bit-Wert ausgegeben. Entsprechender Wertebereich für QV –4096 bis 4095

Diagnose DG01...DG16

Telegrammaufbau

Byte	Bedeutung	Wert (hex) sendet	
		Master	Slave
0		→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	39	39
3	Instanz	01 - 10	01 - 10
4	Index	00 - 03	00 - 03
5 - 8	Data 1 - 4 beim Lesen	00	abhängig vom Index, → Tabelle 127, 128

Tabelle 127: Operandenübersicht

Index (hex)	Daten	Data 1, Data 3, Data 4	Data 2	Read/Write
0	Bit-IO	→ Tabelle 128	–	R
2	Diagnose-Register QV	DWORD oder UDINT ¹⁾		R
3	Ausgangs-Zustände ON	DWORD oder UDINT ¹⁾		R

1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte.

Tabelle 128: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	EN ¹⁾
FB-Ausgang Data 3		Q8 ²⁾	Q7 ²⁾	Q6 ²⁾	Q5 ²⁾	Q4 ²⁾	Q3 ²⁾	Q2 ²⁾	Q1 ²⁾
FB-Ausgang Data 4		–	–	–	–	–	–	–	QC ³⁾

- 1) Im Zustand „1“ wird der Funktionsbaustein aktiviert.
- 2) Der Zustand „1“ wird eingenommen, wenn der ausgewählte Sicherheitsbaustein den gewählten Zustand besitzt.
- 3) Der Zustand „1“ wird eingenommen, wenn einer der Ausgänge Q1 bis Q8 den Zustand „1“ eingenommen hat.



Weitere Informationen zu diesem Baustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easySoft-Hilfe.

Signalglättungsfilter FT01...FT32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	→ Seite 154
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	28	28
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 129	→ Tabelle 129
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 130
	beim Schreiben	abhängig vom Index, → Tabelle 130	00

Tabelle 129: Operandenübersicht

Index (hex)	Operand		lesen	schreiben
00	Bit-IO, → Tabelle 130		×	
01	Mode/Parameter		–	–
02	Eingangswert, Wertebereich: –32 768 bis +32 767	I1	×	c ¹⁾
03	Ausgleichszeit [0,1 s] Wertebereich: 0 bis 65 535	TG	×	c ¹⁾
04	Proportionalverstärkung [%], Wertebereich: 0 bis 65 535	KP	×	c ¹⁾
05	Verzögerter Ausgangswert, Wertebereich: –32 768 bis +32 767	QV	×	

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 130: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		–	–	–	–	–	–	–	EN ¹⁾

1) Aktiviert den Baustein bei Zustand „1“

Empfang von Netzdaten GT01...GT32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl: Lesen	92	–
	Antwort: Lesen erfolgreich	–	C2
	Befehl zurück- gewiesen	–	C0
2	Type	1B	1B
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 131	
5 - 8	Data 1 - 4	00	abhängig vom Index, → Tabelle 132, 133

Tabelle 131: Operandenübersicht

Index (hex)	Operand	lesen	schreiben
00	Bit-IO, → Tabelle 132	×	
01	Mode/Parameter, → Tabelle 133	×	–
02	Ausgangswert: Istwert QV aus dem Netzwerk	×	



Die Daten für Index 2 werden als 32-Bit-Wert im Intel-format (Data 1 – Low Byte bis Data 4 – High Byte) übergeben.

Tabelle 132: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q ¹⁾

1) Zustand „1“, wenn ein neuer Wert anliegt, der vom Netzwerk NET übertragen wird.

Tabelle 133: Index 1 – Mode/Parameter (Bezeichnung des FB PUT, dessen Daten zu empfangen sind)

Mode	Data 1	NET-ID ¹⁾	
		0	NET-ID 1
	
		7	NET-ID 8
Parameter	Data 3	Instanz ²⁾	
		0	PT01
	
		31	PT32

1) Nummer des Teilnehmers, der den Wert sendet. Mögliche Teilnehmernummer: 01 bis 08

2) Sendebaustein (z. B. PT 20) des NET-Teilnehmers, der sendet. Mögliche Bausteinnummer: 01 - 32

Wochenzeitschaltuhr HW01...HW32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl: Lesen	92	–
	Antwort: Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0
2	Type	1C	1C
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 134	
5 - 8	Data 1 - 4	00	abhängig vom Index, → Tabelle 135

Tabelle 134: Operandenübersicht

Index (hex)	Operand	lesen	schreiben
00	Bit-IO → Tabelle 135	×	
01	Mode/Parameter	–	–
02	Parameter → Tabelle 136	×	
	Kanal A		
03	Kanal B		
04	Kanal C		
05	Kanal D		

Tabelle 135: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q ¹⁾

1) Zustand „1“, wenn die Einschaltbedingung erfüllt ist.

Die Daten sind in der nachfolgenden Tabelle im Motorola-format dargestellt, obwohl sie tatsächlich im Intelformat übergeben werden.

Tabelle 136: Index 2 - 5, Parameterkanäle A - D

Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
	Date 2								Date 1							
ON	d4	d3	d2	d1	d0	h4	h3	h2	h1	h0	m5	m4	m3	m2	m1	m0
	Wochentag					Stunde					Minute					

Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
	Date 4								Date 3							
OFF	d4	d3	d2	d1	d0	h4	h3	h2	h1	h0	m5	m4	m3	m2	m1	m0
	Wochentag					Stunde					Minute					

m5 bis m0: Minute (0 bis 59)

h4 bis h0: Stunde (0 bis 23)

d5 bis d0: Wochentag (0 = Sonntag bis 6 = Samstag)

Beispiel

Es sollen die Parameter Kanal A der Wochenschaltuhr HW19 gelesen werden.

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	81	80
1	Befehl: Lesen	92	–
	Antwort: Lesen erfolgreich	–	C2
2	Type	1C	1C
3	Instanz	13	13
4	Index	02	02
5	Data 1	00	62
6	Data 2	00	0B
7	Data 3	00	7B
8	Data 4	00	25

Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
	Date 2 = 0B _{hex}								Date 1 = 62 _{hex}							
ON	0	0	0	0	1	0	1	1	0	1	1	0	0	0	1	0
	Wochentag					Stunde				Minute						

Einschaltzeitpunkt:

Wochentag = 01_{hex}...Montag

Stunde = 0D_{hex}...13 Uhr

Minute = 22_{hex}...34 Minuten

Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
	Date 4 = 25 _{hex}								Date 3 = 7B _{hex}							
OFF	0	0	1	0	0	1	0	1	0	1	1	1	1	0	1	1
	Wochentag					Stunde				Minute						

Ausschaltzeitpunkt:

Wochentag = 04_{hex}...Donnerstag

Stunde = 15_{hex}...21 Uhr

Minute = 59_{hex}...34 Minuten

Jahreszeitschaltuhr HY01...HY32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl: Lesen	92	–
	Antwort: Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0
2	Type	1D	1D
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 137	
5 - 8	Data 1 - 4	00	abhängig vom Index, → Tabelle 138

Tabelle 137: Operandenübersicht

Index (hex)	Operand	lesen	schreiben
00	Bit-IO → Tabelle 138	×	
01	Mode/Parameter	–	–
02	Parameter → Tabelle 139	×	
03	Kanal A		
04	Kanal B		
05	Kanal C		
06	Kanal D		

Tabelle 138: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q ¹⁾

1) Zustand „1“, wenn die Einschaltbedingung erfüllt ist.

Die Daten sind in der nachfolgenden Tabelle im Motorola-format dargestellt, obwohl sie tatsächlich im Intelformat übergeben werden.

Tabelle 139: Index 2 - 5, Parameterkanäle A - D

Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
	Date 2								Date 1							
ON	y6	y5	y4	y3	y2	y1	y0	m3	m2	m1	m0	d4	d3	d2	d1	d0
	Jahr							Monat			Tag					

Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
	Date 4								Date 3							
OFF	y6	y5	y4	y3	y2	y1	y0	m3	m2	m1	m0	d4	d3	d2	d1	d0
	Jahr							Monat			Tag					

d4...d0: Tag (1...31), m3...m0: Monat (1...12), y6...y0: Jahr (0: 2000...99: 2099)

Beispiel

Es sollen die Parameter Kanal A der Jahresschaltuhr HY14 geschrieben werden.

Index 2 - 5, Parameterkanäle A - D

Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
	Date 2								Date 1							
ON	0	0	0	0	0	1	1	0	1	1	0	0	1	1	1	0
	Jahr							Monat			Tag					

Einschaltzeitpunkt:

Tag = 14 = 0E_{hex} = 0000 1110b

Monat = 6 (Juni) = 06_{hex} = 0000 0110b

Jahr = 2003 = 03_{hex} = 0000 0011b

Index 2 - 5, Parameterkanäle A - D

Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
	Date 2								Date 1							
Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
	Date 4								Date 3							
OFF	y6	y5	y4	y3	y2	y1	y0	m3	m2	m1	m0	d4	d3	d2	d1	d0
	Jahr								Monat				Tag			

Ausschaltzeitpunkt:

Tag = 3 = 03_{hex} = 0000 0011b

Monat = 10 (Oktober) = 0A_{hex} = 0000 1010b

Jahr = 2012 = 0C_{hex} = 0000 1100b

Resultierendes Telegramm:

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	81	80
1	Befehl: Schreiben	B2	–
	Antwort: Schreiben erfolgreich	–	C1
2	Type	1D	1D
3	Instanz	0E	0E
4	Index	02	02
5	Data 1	8E	00
6	Data 2	06	00
7	Data 3	43	00
8	Data 4	19	00



Weitere Informationen finden Sie im S40-Anwendungshinweis AN27K21d.exe „EASY800/MFD-DP-Hantierungsbausteine für PS416 und PS4-341.

Bedingter Sprung JC01...JC32

Telegrammaufbau

Byte	Bedeutung	Wert (hex) sendet	
		Master	Slave
0		→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	2F	2F
3	Instanz	01 - 20	01 - 20
4	Index	00	00
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 140, 141
	beim Schreiben	abhängig vom Index, → Tabelle 140, 141	00

Tabelle 140: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
0	Bit-IO	→ Tabelle 141	–	→ Tabelle 141	–	R

Tabelle 141: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	EN ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	E1 ²⁾

- 1) Im Zustand „1“ wird im Programm auf die zugehörige Sprungmarke verzweigt.
- 2) Zustand „1“ wird eingenommen, wenn die zugehörige Sprungmarke nicht gefunden wurde.



Weitere Informationen zu diesem Baustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easySoft-Hilfe.

Wertskalierung LS01...LS32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	→ Seite 154
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	29	29
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 142	→ Tabelle 142
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 143
	beim Schreiben	abhängig vom Index, → Tabelle 143	

Tabelle 142: Operandenübersicht

Index (hex)	Operand		lesen	schreiben
00	Bit-IO, → Tabelle 143		×	
01	Mode/Parameter		–	–
02	Eingangswert, Wertebereich: 32 Bit	I1	×	c ¹⁾
03	Stützpunkt 1, X-Koordinate, Wertebereich: 32 Bit	X1	×	c ¹⁾
04	Stützpunkt 1, Y-Koordinate, Wertebereich: 32 Bit	Y1	×	c ¹⁾
05	Stützpunkt 2, X-Koordinate, Wertebereich: 32 Bit	X2	×	c ¹⁾
06	Stützpunkt 2, Y-Koordinate, Wertebereich: 32 Bit	Y2	×	c ¹⁾
07	Ausgangswert: beinhaltet den skalierten Eingangswert	QV	×	

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 143: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		–	–	–	–	–	–	–	EN ¹⁾

1) Aktiviert den Baustein bei Zustand „1“

Masterreset MR01...MR32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl: Lesen	92	–
	Antwort: Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0
2	Type	0F	0F
3	Instanz	01 - 20	01 - 20
4	Index		
	Bit-IO	00	00
	Mode	01	01
5 - 8	Data 1 - 4	00	abhängig vom Index, → Tabelle 144, 145

Tabelle 144: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	T ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q1 ²⁾

- 1) Triggerspule. Wird die Spule getriggert (erhält eine positive Flanke), wird der entsprechende Reset durchgeführt.
- 2) Zustand „1“, wenn die Triggerspule MR...T den Zustand „1“ besitzt.

Tabelle 145: Index 1 – Mode

Data 1 (hex)		
00	Q	die Ausgänge Q..., *Q..., S..., *S..., *SN..., QA01 werden auf den Zustand „0“ zurückgesetzt. * entsprechend der NET-ID
01	M	der Merkerbereich MD01 bis MD48 wird auf Zustand „0“ zurückgesetzt
02	ALL	wirkt auf Q und M.

Datenmultiplexer MX01...MX32

Telegrammaufbau

Byte	Bedeutung	Wert (hex) sendet	
		Master	Slave
0		→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	31	31
3	Instanz	01 - 20	01 - 20
4	Index	00 - 0B	00 - 0B
5 - 8	Data 1 - 4		
	beim Lesen	00	abh. vom Index → Tabelle 146, 147
	beim Schreiben	abhängig vom Index, → Tabelle 146, 147	00

Tabelle 146: Operandenübersicht

Index (hex)	Daten	Data 1 Data 3	Data 2 Data 4	Read/Write
0	Bit-IO	→ Tabelle 147	–	R
2	Kanalauswahl: 0 bis 7	DWORD oder UDINT ¹⁾		R/W ²⁾
3	Eingangswert Kanal 1	DWORD oder UDINT ¹⁾		R/W ²⁾
4	Eingangswert Kanal 2	DWORD oder UDINT ¹⁾		R/W ²⁾
5	Eingangswert Kanal 3	DWORD oder UDINT ¹⁾		R/W ²⁾
6	Eingangswert Kanal 4	DWORD oder UDINT ¹⁾		R/W ²⁾
7	Eingangswert Kanal 5	DWORD oder UDINT ¹⁾		R/W ²⁾
8	Eingangswert Kanal 6	DWORD oder UDINT ¹⁾		R/W ²⁾
9	Eingangswert Kanal 7	DWORD oder UDINT ¹⁾		R/W ²⁾
A	Eingangswert Kanal 8	DWORD oder UDINT ¹⁾		R/W ²⁾
B	Ausgangswert QV	DWORD oder UDINT ¹⁾		R

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte.
- 2) Der Wert kann nur beschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 147: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	EN ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	E1 ²⁾

- 1) Im Zustand „1“ wird der gewählte Eingangswert in den Ausgangswert eingetragen.
- 2) Zustand „1“ wird eingenommen, wenn die Kanalauswahl ungültig ist.



Weitere Informationen zu diesem Baustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easySoft-Hilfe.

Zahlenwandler NC01...NC32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	→ Seite 154
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	2A	2A
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 148	→ Tabelle 148
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 149, 150
	beim Schreiben	abhängig vom Index, → Tabelle 149, 150	00

Tabelle 148: Operandenübersicht

Index (hex)	Operand	lesen	schreiben
00	Bit-IO, → Tabelle 149	×	
01	Mode, → Tabelle 150	×	
02	Eingangswert: I1 Operand der gewandelt werden soll	×	c ¹⁾
03	Ausgangswert: QV beinhaltet das Wandlungsergebnis	×	

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.



Die Daten für Index 2 und 3 werden als 32-Bit-Wert im Intelformat (Data 1 – LowByte...Data 2 – HighByte) übergeben.

Tabelle 149: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		–	–	–	–	–	–	–	EN ¹⁾

1) Aktiviert den Baustein bei Zustand „1“

Tabelle 150: Index 1 – Mode

Data 1 (hex)		
00	BCD	Wandelt einen BCD-codierten Dezimalwert in einen ganzzahligen Wert um.
01	BIN	Wandelt einen ganzzahligen Wert in einen BCD-codierten Dezimalwert um.

Betriebsstundenzähler OT01...OT04

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	→ Seite 154
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	1E	1E
3	Instanz	01 - 04	01 - 04
4	Index	→ Tabelle 151	→ Tabelle 151
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 152
	beim Schreiben	abhängig vom Index, → Tabelle 152	00

Tabelle 151: Operandenübersicht

Index (hex)	Operand	lesen	schreiben
00	Bit-IO, → Tabelle 152	×	
01	Mode/Parameter	–	–
02	oberer Schwellwert I1	×	c ¹⁾
03	Istwert des Betriebsstundenzählers QV	×	

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 152: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	RE ¹⁾	EN ²⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q1 ³⁾

- 1) Resetspule: Zustand „1“ setzt den Zähler-Istwert auf Null zurück.
- 2) Freigabespule
- 3) Zustand „1“, wenn Sollwert erreicht wurde (größer/gleich)



Die Daten für Index 2 und 3 werden als 32-Bit-Wert im Intelformat (Data 1 – Low Byte bis Data 4 – High Byte) übergeben.

Impulsausgabe P001...P002

Telegrammaufbau

Byte	Bedeutung	Wert (hex) sendet	
		Master	Slave
0		→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	32	32
3	Instanz	01 - 02	01 - 02
4	Index	00 - 0A	00 - 0A
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 153, 154
	beim Schreiben	abhängig vom Index, → Tabelle 153, 154	00

Tabelle 153: Operandenübersicht

Index (hex)	Daten	Data 1 Data 3	Data 2 Data 4	Read/ Write
0	Bit-I/O	→ Tabelle 154	–	R
2	Impulsanzahl im Fahrbetrieb I1: 0 bis 2147483647	DWORD oder UDINT ¹⁾		R/W ²⁾
3	Startfrequenz FS: 0 bis 5000 Hz	DWORD oder UDINT ¹⁾		R/W ²⁾
4	Betriebsfrequenz FO: 0 bis 5000 Hz	DWORD oder UDINT ¹⁾		R/W ²⁾
5	Frequenzänderung im Hochlauf RF: 0 bis 65535 mHz	DWORD oder UDINT ¹⁾		R/W ²⁾
6	Frequenzänderung beim Bremsen BF: 0 bis 65535 mHz	DWORD oder UDINT ¹⁾		R/W ²⁾
7	Schrittzahl im Tippbetrieb P1: 0 bis 65535	DWORD oder UDINT ¹⁾		R/W ²⁾
8	Frequenz im Tippbetrieb PF: 0 bis 5000 Hz	DWORD oder UDINT ¹⁾		R/W ²⁾
9	Aktuelle Schrittzahl QV	DWORD oder UDINT ¹⁾		R
A	Aktuelle Frequenz QF	DWORD oder UDINT ¹⁾		R

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte.
- 2) Der Wert kann nur beschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 154: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	TP ¹⁾	BR ²⁾	ST ³⁾	EN ⁴⁾
FB-Ausgang Data 3		–	–	–	–	–	–	E1 ⁵⁾	AC ⁶⁾

- 1) Bei Erhalt einer positiven Flanke wird der Tippbetrieb gestartet.
- 2) Bei Erhalt einer positiven Flanke wird der Fahrauftrag abgebrochen.
- 3) Bei Erhalt einer positiven Flanke wird der Fahrauftrag gestartet.
- 4) Im Zustand „1“ wird der Funktionsbaustein aktiviert.
- 5) Zustand „1“ wird eingenommen, wenn die Parametereingabe ungültig ist.
- 6) Zustand „1“ wird eingenommen, wenn ein Fahrauftrag aktiv ist.



Weitere Informationen zu diesem Baustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easySoft-Hilfe.

Senden von Netzdaten PT01...PT32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	1F	1F
3	Instanz	01 - 20	01 - 20
4	Index	00 - 02	00 - 02
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 155, 156
	beim Schreiben	abhängig vom Index, → Tabelle 155, 156	00

Tabelle 155: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
0	Bit-IO	→ Tabelle 156	–	→ Tabelle 156	–	R
2	Sollwert QV für das Netzwerk	DWORD oder UDINT ¹⁾				R

1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte.

Tabelle 156: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	T ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	E ¹²⁾	AC ³⁾	Q ¹⁴⁾

- 1) Triggerspule. Wird die Spule getriggert (erhält eine positive Flanke), wird der entsprechende Wert dem NET zur Verfügung gestellt.
- 2) Zustand „1“ wird eingenommen, wenn der Sendeauftrag wegen eines Fehlers abgebrochen wurde.
- 3) Zustand „1“ wird eingenommen, wenn die Triggerspule getriggert wird. Zustand „0“ wird eingenommen, wenn der Sendeauftrag erfolgreich durchgeführt wurde oder wegen eines Fehlers abgebrochen wurde.
- 4) Zustand „1“, wenn der Zustand der Triggerspule ebenfalls „1“ ist.



Weitere Informationen zu diesem Baustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easySoft-Hilfe.

Pulsweitenmodulation PW01...PW02

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	→ Seite 154
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	2B	2B
3	Instanz	01 - 02	01 - 02
4	Index	→ Tabelle 157	→ Tabelle 157
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 158
	beim Schreiben	abhängig vom Index, → Tabelle 158	00

Tabelle 157: Operandenübersicht

Index (hex)	Operand		lesen	schreiben
00	Bit-IO, → Tabelle 158		×	
01	Mode/Parameter		–	–
02	Stellgröße, Wertebereich: 0 bis 4095 (12 Bit) SV		×	c ¹⁾
03	Periodendauer [ms], Wertebereich: 0 bis 65535 PD		×	c ¹⁾
04	Mindesteinschaltdauer [ms], Wertebereich: 0 bis 65535 ME		×	c ¹⁾

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschriftet ist.

Tabelle 158: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	EN ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	E1 ²⁾

- 1) Aktiviert den Baustein bei Zustand „1“.
- 2) Zustand „1“, wenn die Mindesteinschaltdauer oder die Mindestausschaltdauer unterschritten wird

Uhr synchronisieren SC01

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	
1	Befehl: Lesen	92	–
	Antwort: Lesen erfolgreich	–	C2
	Befehl zurückgewiesen	–	C0
2	Type	20	20
3	Instanz	01	01
4	Index	→ Tabelle 159	
5 - 8	Data 1 - 4	00	abhängig vom Index, → Tabelle 160

Tabelle 159: Operandenübersicht

Index (hex)	Operand	lesen	schreiben
00	Bit-IO, → Tabelle 160	×	
01	Mode/Parameter	–	–

Tabelle 160: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	T ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q ¹²⁾

- 1) Triggerspule. Wird die Spule getriggert (erhält eine positive Flanke), so wird automatisch das aktuelle Datum, der Wochentag und die Uhrzeit des sendenden Teilnehmers in das Netzwerk NET gestellt.
- 2) Zustand „1“, wenn der Zustand der Triggerspule SC01T_ ebenfalls „1“ ist.

Serielle Ausgabe SP01...SP32

Telegrammaufbau

Byte	Bedeutung	Wert (hex) sendet	
		Master	Slave
0		→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	35	35
3	Instanz	01 - 20	01 - 20
4	Index	00	00
5 - 8	Data 1 - 4 beim Lesen	00	abhängig vom Index, → Tabelle 161, 162

Tabelle 161: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
0	Bit-IO	→ Tabelle 162	–	→ Tabelle 162	–	R

Tabelle 162: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	T ¹⁾	EN ²⁾
FB-Ausgang Data 3		–	–	–	–	–	–	E1 ³⁾	AC ⁴⁾

- 1) Bei Erhalt einer positiven Flanke wird der Sendevorgang ausgelöst.
- 2) Im Zustand „1“ wird der Funktionsbaustein aktiviert.
- 3) Der Zustand „1“ wird eingenommen, wenn beim Sendevorgang ein Fehler aufgetreten ist.
- 4) Der Zustand „1“ wird eingenommen, wenn der Sendevorgang aktiv ist.



Weitere Informationen zu diesem Baustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easySoft-Hilfe.

Schieberegister SR01...SR32

Telegrammaufbau

Byte	Bedeutung	Wert (hex) sendet	
		Master	Slave
0		→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	33	33
3	Instanz	01 - 20	01 - 20
4	Index	00 - 0B	00 - 0B
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 163, 173, 174
	beim Schreiben	abhängig vom Index, → Tabelle 163, 173, 174	00

Tabelle 163: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2 Data 4	Data 3	Read/ Write
0	Bit-IO	→ Tabelle 173	–	→ Tabelle 173	R
1	Mode	→ Tabelle 174	–	–	R
2	Dateneingang vorwärts I1	DWORD oder UDINT ¹⁾			R/W ²⁾
3	Dateneingang rückwärts I2	DWORD oder UDINT ¹⁾			R/W ²⁾
4	Datenausgang 1 (D1)	DWORD oder UDINT ¹⁾			R
5	Datenausgang 2 (D2)	DWORD oder UDINT ¹⁾			R
6	Datenausgang 3 (D3)	DWORD oder UDINT ¹⁾			R
7	Datenausgang 4 (D4)	DWORD oder UDINT ¹⁾			R
8	Datenausgang 5 (D5)	DWORD oder UDINT ¹⁾			R
9	Datenausgang 6 (D6)	DWORD oder UDINT ¹⁾			R
A	Datenausgang 7 (D7)	DWORD oder UDINT ¹⁾			R
B	Datenausgang 8 (D8)	DWORD oder UDINT ¹⁾			R

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte.
- 2) Der Wert kann nur beschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Sollzykluszeit ST01

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	→ Seite 154
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	2C	2C
3	Instanz	01	01
4	Index	→ Tabelle 164	→ Tabelle 164
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 165
	beim Schreiben	abhängig vom Index, → Tabelle 165	00

Tabelle 164: Operandenübersicht

Index (hex)	Operand	lesen	schreiben
00	Bit-IO, → Tabelle 165	×	
01	Mode/Parameter	–	–
02	Zykluszeit in ms, Wertebereich: 0 - 1000	×	c ¹⁾

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 165: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		–	–	–	–	–	–	–	EN ¹⁾

1) Aktiviert den Baustein bei Zustand „1“

Zeitrelais T01...T32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	→ Seite 154
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	21	21
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 166	→ Tabelle 166
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 167, 168
	beim Schreiben	abhängig vom Index, → Tabelle 167, 168	

Tabelle 166: Operandenübersicht

Index (hex)	Operand	lesen	schreiben
00	Bit-IO, → Tabelle 167	×	
01	Mode/Parameter, → Tabelle 168	×	
02	Sollwert 1: I1 Zeitsollwert 1	×	c ¹⁾
03	Sollwert 2: I2 Zeitsollwert 2 (bei Zeitrelais mit 2 Sollwerten)	×	c ¹⁾
04	Istwert: QV Abgelaufene Istzeit im „Run“-Betrieb	×	

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.



Die Daten für Index 2 bis 4 werden als 32-Bit-Wert im Intelformat (Data 1 – Low Byte bis Data 4 – High Byte) übergeben.

Tabelle 167: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	ST ¹⁾	EN ²⁾	RE ³⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q1 ⁴⁾

- 1) Stopp, das Zeitrelais wird gestoppt (Stoppspule)
- 2) Enable, das Zeitrelais wird gestartet (Triggerspule)
- 3) Reset, das Zeitrelais wird zurückgesetzt (Resetspule)
- 4) Schaltkontakt

Tabelle 168: Index 1 – Mode/Parameter

Mode	Data 1	Betriebsart
	0	Ansprechverzögert
	1	Ansprechverzögert mit Zufalls-Sollwert
	2	Rückfallverzögert
	3	Rückfallverzögert mit Zufalls-Sollwert
	4	Ansprechverzögert und Rückfallverzögert (zwei Zeitsollwerte)
	5	Ansprechverzögert und Rückfallverzögert, jeweils mit Zufalls-Sollwert (zwei Zeitsollwerte)
	6	Impulsgeber
	7	Blink-Relais (zwei Zeitsollwerte)
	8	Rückfallverzögert, retriggerbar (easy600 Mode)
	9	Rückfallverzögert mit Zufalls-Sollwert, retriggerbar (easy600 Mode)
Parameter	Data 3	Betriebsart
	0	S (Millisekunden)
	1	M:S (Sekunden)
	2	H:M (Minuten)

Tabellenfunktion TB01...TB32

Telegrammaufbau

Byte	Bedeutung	Wert (hex) sendet	
		Master	Slave
0		→ Seite 154	
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	34	34
3	Instanz	01 - 20	01 - 20
4	Index	00 - 04	00 - 04
5 - 8	Data 1 - 4		
	beim Lesen	–	abhängig vom Index, → Tabelle 169, 170
	beim Schreiben	abhängig vom Index, → Tabelle 169, 170	00

Tabelle 169: Operandenübersicht

Index (hex)	Daten	Data 1 Data 3	Data 2 Data 4	Read/ Write
0	Bit-IO	→ Tabelle 170	–	R
2	Eingangswert I1 für Tabelle von TB...	DWORD oder UDINT ¹⁾		R/W ²⁾
3	Ausgangswert QV aus Tabelle von TB...	DWORD oder UDINT ¹⁾		R
4	Anzahl Einträge QN in Tabelle von TB...	DWORD oder UDINT ¹⁾		R

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte.
- 2) Der Wert kann nur beschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 170: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	RE ¹⁾	RL ²⁾	RF ³⁾	WP ⁴⁾	EN ⁵⁾
FB-Ausgang Data 3		–	–	–	–	–	–	TF ⁶⁾	TE ⁷⁾

- 1) Bei Erhalt einer positiven Flanke werden alle Einträge aus der Tabelle entfernt. Die Anzahl der Tabelleneinträge QN wird auf „0“ gesetzt.
- 2) Bei Erhalt einer positiven Flanke wird der neuste Eintrag in der Tabelle am Ausgang QV ausgegeben und aus der Tabelle entfernt. Die Anzahl der Tabelleneinträge QN wird um eins erniedrigt.
- 3) Bei Erhalt einer positiven Flanke wird der älteste Eintrag in der Tabelle am Ausgang QV ausgegeben und aus der Tabelle entfernt. Die Anzahl der Tabelleneinträge QN wird um eins erniedrigt.
- 4) Bei Erhalt einer positiven Flanke wird der Wert von I1 in die Tabelle übernommen und die Anzahl der Tabelleneinträge QN um eins erhöht, solange die maximale Anzahl von Einträgen nicht überschritten wird. In diesem Fall wird der Wert von I1 am Ausgang QV ausgegeben.
- 5) Im Zustand „1“ wird der Funktionsbaustein aktiviert.
- 6) Der Zustand „1“ wird eingenommen, wenn die Tabelle voll ist.
- 7) Der Zustand „1“ wird eingenommen, wenn die Tabelle leer ist.



Weitere Informationen zu diesem Baustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easySoft-Hilfe.

Wertbegrenzung VC01...VC32

Telegrammaufbau

Byte	Bedeutung	Wert (hex), sendet	
		Master	Slave
0	Toggelbyte	→ Seite 154	→ Seite 154
1	Befehl:		
	Lesen	92	–
	Schreiben	B2	–
	Antwort:		
	Lesen erfolgreich	–	C2
	Schreiben erfolgreich	–	C1
	Befehl zurückgewiesen	–	C0
2	Type	2D	2D
3	Instanz	01 - 20	01 - 20
4	Index	→ Tabelle 171	→ Tabelle 171
5 - 8	Data 1 - 4		
	beim Lesen	00	abhängig vom Index, → Tabelle 172
	beim Schreiben	abhängig vom Index, → Tabelle 172	

Tabelle 171: Operandenübersicht

Index (hex)	Operand	lesen	schreiben
00	Bit-IO, → Tabelle 172	×	
01	Mode/Parameter	–	–
02	Eingangswert I1	×	c ¹⁾
03	Oberer Grenzwert SH	×	c ¹⁾
04	Unterer Grenzwert SL	×	c ¹⁾
05	Ausgangswert: gibt innerhalb der gesetzten Grenzen den Wert aus, der am Eingang I1 anliegt. QV	×	

1) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 172: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		–	–	–	–	–	–	–	EN ¹⁾

1) Aktiviert den Baustein bei Zustand „1“

Tabelle 173: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	BD ¹⁾	FD ²⁾	RE ³⁾	BP ⁴⁾	FP ⁵⁾	EN ⁶⁾
FB-Ausgang Data 3		Q8 ⁷⁾	Q8 ⁷⁾	Q6 ⁷⁾	Q5 ⁷⁾	Q4 ⁷⁾	Q3 ⁷⁾	Q2 ⁷⁾	Q1 ⁷⁾

- 1) Eingangs-Bitwert für die Rückwärts-Schiebeoperation im Mode BIT
- 2) Eingangs-Bitwert für die Vorwärts-Schiebeoperation im Mode BIT
- 3) Im Zustand „1“ wird der Funktionsbaustein zurückgesetzt.
- 4) Bei Erhalt einer positiven Flanke wird im Mode BIT der Wert von BD in das letzte Register-Feld Q8 eingetragen und die ursprünglichen Inhalte der Registerfelder werden um ein Feld in Richtung niedrigerer Feldnummern verschoben. Bei Erhalt einer positiven Flanke wird im Mode DW der Wert von I2 in das letzte Register-Feld D8 eingetragen und die ursprünglichen Inhalte der Registerfelder werden um ein Feld in Richtung niedrigerer Feldnummern verschoben.
- 5) Bei Erhalt einer positiven Flanke wird im Mode BIT der Wert von FD in das erste Register-Feld Q1 eingetragen und die ursprünglichen Inhalte der Registerfelder werden um ein Feld in Richtung höherer Feldnummern verschoben. Bei Erhalt einer positiven Flanke wird im Mode DW der Wert von I1 in das erste Register-Feld D1 eingetragen und die ursprünglichen Inhalte der Registerfelder werden um ein Feld in Richtung höherer Feldnummern verschoben.
- 6) Im Zustand „1“ wird der Funktionsbaustein aktiviert.
- 7) Zustand der acht Felder des Bit-Schieberegisters

Tabelle 174: Index 1 – Mode

Data 1 (hex)		
00	BIT	Betriebsart: Bit schieben
01	DW	Betriebsart: Doppelwort schieben



Weitere Informationen zu diesem Baustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easySoft-Hilfe.

10 Prozessdaten easy600 (DPV1)

Mit Hilfe der PROFIBUS-DPV1-Funktionalität des EASY204-DP (ab Geräteversion 07) können Sie ohne großen Programmieraufwand von einem DPV1-Master der Klasse 1 oder Klasse 2 auf folgende Datenbereiche eines easy600 zugreifen:

- Identifikation
- Betriebsart
- Abbild
- Funktionsbausteine
- Datum und Uhrzeit
- Sommer-/Winterzeit.

Die Daten aus diesen Bereichen befinden sich in Process Data Objekten (Datensätzen) und können mit Hilfe der DPV1- Dienste „Read“ und „Write“ gelesen und/oder geschrieben werden.

Objektübersicht

Die folgende Übersicht zeigt alle im EASY204-DP enthaltenen Objekte, die für ein angeschlossenes easy600 bereitgestellt werden.



Die Attribute API, Slot Number und Index bilden die Adressinformation für den DP-Master für die Ansprache eines Objektes. Weitere Erläuterungen hierzu entnehmen Sie dem folgenden Abschnitt.

Tabelle 175: Process Data Objekte im EASY204-DP für easy600

Objekt-Name	API	Slot Number	Index	Datenlänge (Octets)	lesbar (R) schreibbar (W)	→ Seite
Easy Betriebsart	0	0	100	1	R/W	265
Easy Identifikation	0	1	100	50	R	263
Abbilddaten						
EASY 600 Eingänge (I1 - I16)	0	0	197	4	R	267
EASY 600 Ausgänge und Merker (Q1 - Q8, M1 - M16, D1 - D8)	0	0	198	4	R	269
EASY 600 Funktionsrelais (T1 - T8, C1 - C8, O1 - O4, A1 - A8)	0	0	199	4	R	272
EASY 600 Tasten	0	0	200	1	R	275
Easy Link Eingangsdaten (R1 - R16)	0	0	98	2	R/W	277
Easy Link Ausgangsdaten (S1 - S8)	0	0	99	1	R	279
Funktionsbausteine						
EASY 600 Parameter Zeitrelais T1 bis T8	0	0	211 - 218	6 lesend 5 schreibend	R/W	281
EASY 600 Parameter Zähler C1 bis C8	0	0	221 - 228	2 lesend 3 schreibend	R/W	286

Objekt-Name	API	Slot Number	Index	Datenlänge (Octets)	lesbar (R) schreibbar (W)	→ Seite
EASY 600 Parameter Analogwertvergleichler A1 - A8	0	0	231 - 238	3	R/W	289
EASY 600 Parameter Wochenschaltuhr 01Kanal A	0	0	239	8	R/W	291
...			...			
EASY 600 Parameter Wochenschaltuhr 01 Kanal D	0	0	242	8	R/W	291
EASY 600 Parameter Wochenschaltuhr 02 Kanal A	0	0	243	8	R/W	291
...			...			
EASY 600 Parameter Wochenschaltuhr 02 Kanal D	0	0	246	8	R/W	291
EASY 600 Parameter Wochenschaltuhr 03 Kanal A	0	0	247	8	R/W	291
...			...			
EASY 600 Parameter Wochenschaltuhr 03 Kanal D	0	0	250	8	R/W	291
EASY 600 Parameter Wochenschaltuhr 04 Kanal A	0	0	251	8	R/W	291
...			...			
EASY 600 Parameter Wochenschaltuhr 04 Kanal D	0	0	254	8	R/W	291
Easy Uhr	0	0	97	8	R/W	294
EASY 600 Sommer-/ Winterzeit Einstellung	0	0	210	1	R/W	297

Zugriff auf Objekte

Der Zugriff auf die Process Data Objekte im EASY204-DP mittels der DPV1-Dienste „Read“ und „Write“ erfolgt mit Hilfe der vom DP-Master-System hierfür bereitgestellten Funktionen. Beachten Sie diesbezüglich die Dokumentation des Herstellers. In der Regel werden für den Zugriff Funktionsbausteine bereitgestellt. In IEC 61131-3 basierten Systemen werden oft die von der PROFIBUS-Nutzerorganisation in der Richtlinie 2.182 definierten Funktionsbausteine RDREC (Read) und WRREC (Write) angeboten, die einen optimalen Zugriff auch auf komplexe Datenstrukturen ermöglichen.

Für die Ansprache der Objekte benötigen Sie folgende Angaben:

- Die Adresse der lokalen DPV1-Master-Anschaltung
- Die Teilnehmeradresse des anzusprechenden EASY204-DP
- Den Identifizier der Applikation (API) im EASY204-DP (Angabe nur bei DPV1-Master der Klasse 2 erforderlich)
- Das zu adressierende Modul des EASY204-DP (Slot Number)
- Die Adresse (Index) des gewünschten Process Data Objektes im adressierten Modul des EASY204-DP
- Die Datenlänge des gewünschten Process Data Objektes
- Eine definierte Variable (Speicherbereich) in der lokalen Anwendung, der die ausgelesenen Daten zugewiesen werden sollen oder die die zu schreibenden Daten enthält.

Die Adresse der lokalen DPV1-Master-Anschaltung entnehmen Sie der Topologie Ihres Master-Systems. Die Teilnehmeradresse des anzusprechenden EASY204-DP entnehmen Sie der PROFIBUS-Topologie.

Für das EASY204-DP ist für alle Process Data Objekte der API (Angabe nur bei DPV1-Master der Klasse 2 erforderlich) mit 0 anzugeben.

Die Slot Number, den Index, die Datenlänge und eine beispielhafte Variablendefinition (Deklaration) für ein IEC 61131-3 basierendes System für jedes im EASY204-DP für ein angeschlossenes easy600 vorhandene Process Data Objekt finden Sie in den folgenden Abschnitten.

Identifikation lesen

Objekt „Easy Identifikation“

Dieses Objekt enthält die Geräte-Identifikation des EASY204-DP und des daran angeschlossenen easy-Grundgerätes (ASCII-Strings).

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über die Geräte-Identifikation zu informieren:

- Typ des easy-Grundgerätes
- Version der Firmware des easy-Grundgerätes
- Build Number der Firmware des easy-Grundgerätes
- Typ des Kommunikationsmoduls
- Version der Firmware des Kommunikations-Moduls.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 100.

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 50 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierendem System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_Ident	STRUCT
Type	ARRAY [1...14] OF CHAR oder STRING [14]
SW_Version	ARRAY [1...10] OF CHAR oder STRING [10]
SW_Build	ARRAY [1...10] OF CHAR oder STRING [10]
Modul_Type	ARRAY [1...10] OF CHAR oder STRING [10]
Modul_SW_Version	ARRAY [1...6] OF CHAR oder STRING [6]

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 176: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung
Easy_Ident.Type	Octet 1 bis 14	Typ des easy-Grundgerätes
Easy_Ident.SW_Version	Octet 15 bis 24	Version der Firmware des easy-Grundgerätes ¹⁾
Easy_Ident.SW_Build	Octet 25 bis 34	Build-Nummer der Firmware des easy-Grundgerätes ²⁾
Easy_Ident.Modul_Type	Octet 35 bis 44	Typ des Kommunikations-Moduls
Easy_Ident.Modul_SW_Version	Octet 45 bis 50	Version der Firmware des Kommunikations-Moduls

- 1) Bei easy800/MFD nicht ermittelbar. In den entsprechenden Octets wird daher der Text „n.a.“ (not available) ausgegeben.
- 2) Bei easy600/800/MFD nicht ermittelbar. In den entsprechenden Octets wird daher der Text „n.a.“ (not available) ausgegeben.

Alle Dateninhalte sind als ASCII-Strings kodiert.

**Betriebsart lesen/
schreiben****Objekt „Easy Betriebsart“**

Dieses Objekt enthält den aktuellen Zustand der Betriebsart eines easy-Grundgerätes (Run/Stop).

Dieses Objekt ist lesbar (Read) und schreibbar (Write). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der Betriebsart (Run/Stop) des easy-Grundgerätes zu informieren (Read) oder dem easy-Grundgerät eine neue Betriebsart zu übergeben (Write).

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 0
- Index ist gleich 100.

Länge des Objektes

Die Länge der auszulesenden bzw. zu schreibenden Daten beträgt 1 Octet. Tragen Sie keine andere Länge beim Aufruf des Dienstes „Write“ ein oder eine kleinere Länge beim Aufruf des Dienstes „Read“ ; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_Betriebsart	BYTE oder USINT

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 177: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bezeichnung
Easy_Betriebsart	Octet 1	Betriebsart des easy-Grundgerätes → Tabelle 178

Tabelle 178: Kodierung der Betriebsart

Betriebsart-Kode (hex)	Bedeutung
00	Betriebsart „Stop“
01	Betriebsart „Run“

Abbild lesen/schreiben

Objekt „Easy 600 Eingänge“

Dieses Objekt enthält den aktuellen Zustand der Eingänge des easy600.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der Digital-Eingänge (I1 bis I16) und den aktuellen Analogwert der Eingänge I7 und I8 des easy600 zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 0
- Index ist gleich 197.

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 4 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_600_Eingaenge	STRUCT
Analog_7	BYTE oder USINT
Analog_8	BYTE oder USINT
Digital	ARRAY [1...16] OF BOOL oder ARRAY [1...2] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 179: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung	
Easy_600_Eingaenge.Analog_7	Octet 1	Analogwert von I7	
Easy_600_Eingaenge.Analog_8	Octet 2	Analogwert von I8	
Easy_600_Eingaenge.Digital[1]	Octet 3	Bit 0	Zustand I1
Easy_600_Eingaenge.Digital[2]		Bit 1	Zustand I2
Easy_600_Eingaenge.Digital[3]		Bit 2	Zustand I3
Easy_600_Eingaenge.Digital[4]		Bit 3	Zustand I4
Easy_600_Eingaenge.Digital[5]		Bit 4	Zustand I5
Easy_600_Eingaenge.Digital[6]		Bit 5	Zustand I6
Easy_600_Eingaenge.Digital[7]		Bit 6	Zustand I7
Easy_600_Eingaenge.Digital[8]		Bit 7	Zustand I8
Easy_600_Eingaenge.Digital[9]		Octet 4	Bit 0
Easy_600_Eingaenge.Digital[10]	Bit 1		Zustand I10
Easy_600_Eingaenge.Digital[11]	Bit 2		Zustand I11
Easy_600_Eingaenge.Digital[12]	Bit 3		Zustand I12
	Bit 4		Nicht genutzt
	Bit 5		Nicht genutzt
Easy_600_Eingaenge.Digital[15]	Bit 6		Zustand I15
Easy_600_Eingaenge.Digital[16]	Bit 7		Zustand I16



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekt „Easy 600 Ausgänge und Merker“

Dieses Objekt enthält den aktuellen Zustand der Ausgänge, Merker und Textmerker des easy600.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der digitale Ausgänge (Q1 bis Q8), der Merker (M1 bis M16) und der Textmerker (D1 bis D8) des easy600 zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 0
- Index ist gleich 198.

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 4 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_600_Merker_Ausgaenge	ARRAY [1...32] OF BOOL oder ARRAY [1...4] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 180: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_600_Merker_Ausgaenge[1]	Octet 1	Bit 0	Zustand M1
Easy_600_Merker_Ausgaenge[2]		Bit 1	Zustand M2
Easy_600_Merker_Ausgaenge[3]		Bit 2	Zustand M3
Easy_600_Merker_Ausgaenge[4]		Bit 3	Zustand M4
Easy_600_Merker_Ausgaenge[5]		Bit 4	Zustand M5
Easy_600_Merker_Ausgaenge[6]		Bit 5	Zustand M6
Easy_600_Merker_Ausgaenge[7]		Bit 6	Zustand M7
Easy_600_Merker_Ausgaenge[8]		Bit 7	Zustand M8
Easy_600_Merker_Ausgaenge[9]	Octet 2	Bit 0	Zustand M9
Easy_600_Merker_Ausgaenge[10]		Bit 1	Zustand M10
Easy_600_Merker_Ausgaenge[11]		Bit 2	Zustand M11
Easy_600_Merker_Ausgaenge[12]		Bit 3	Zustand M12
Easy_600_Merker_Ausgaenge[13]		Bit 4	Zustand M13
Easy_600_Merker_Ausgaenge[14]		Bit 5	Zustand M14
Easy_600_Merker_Ausgaenge[15]		Bit 6	Zustand M15
Easy_600_Merker_Ausgaenge[16]		Bit 7	Zustand M16

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_600_Merker_Ausgaenge[17]	Octet 3	Bit 0	Zustand Q1
Easy_600_Merker_Ausgaenge[18]		Bit 1	Zustand Q2
Easy_600_Merker_Ausgaenge[19]		Bit 2	Zustand Q3
Easy_600_Merker_Ausgaenge[20]		Bit 3	Zustand Q4
Easy_600_Merker_Ausgaenge[21]		Bit 4	Zustand Q5
Easy_600_Merker_Ausgaenge[22]		Bit 5	Zustand Q6
Easy_600_Merker_Ausgaenge[23]		Bit 6	Zustand Q7
Easy_600_Merker_Ausgaenge[24]		Bit 7	Zustand Q8
Easy_600_Merker_Ausgaenge[25]	Octet 4	Bit 0	Zustand D1
Easy_600_Merker_Ausgaenge[26]		Bit 1	Zustand D2
Easy_600_Merker_Ausgaenge[27]		Bit 2	Zustand D3
Easy_600_Merker_Ausgaenge[28]		Bit 3	Zustand D4
Easy_600_Merker_Ausgaenge[29]		Bit 4	Zustand D5
Easy_600_Merker_Ausgaenge[30]		Bit 5	Zustand D6
Easy_600_Merker_Ausgaenge[31]		Bit 6	Zustand D7
Easy_600_Merker_Ausgaenge[32]		Bit 7	Zustand D8



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekt „Easy 600 Funktionsrelais“

Dieses Objekt enthält den aktuellen Zustand der Funktionsrelais des easy600.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der Zeitrelais (T1 bis T8), der Zähler (C1 bis C8), der Wochenschaltuhren (W1 bis W4) und der Analogwertvergleicher (A1 bis A8) des easy600 zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 0
- Index ist gleich 199.

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 4 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_600_Funktionsrelais	ARRAY [1...32] OF BOOL oder ARRAY [1...4] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 181: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_600_Funktionsrelais[1]	Octet 1	Bit 0	Zustand T1
Easy_600_Funktionsrelais[2]		Bit 1	Zustand T2
Easy_600_Funktionsrelais[3]		Bit 2	Zustand T3
Easy_600_Funktionsrelais[4]		Bit 3	Zustand T4
Easy_600_Funktionsrelais[5]		Bit 4	Zustand T5
Easy_600_Funktionsrelais[6]		Bit 5	Zustand T6
Easy_600_Funktionsrelais[7]		Bit 6	Zustand T7
Easy_600_Funktionsrelais[8]		Bit 7	Zustand T8
Easy_600_Funktionsrelais[9]	Octet 2	Bit 0	Zustand C1
Easy_600_Funktionsrelais[10]		Bit 1	Zustand C2
Easy_600_Funktionsrelais[11]		Bit 2	Zustand C3
Easy_600_Funktionsrelais[12]		Bit 3	Zustand C4
Easy_600_Funktionsrelais[13]		Bit 4	Zustand C5
Easy_600_Funktionsrelais[14]		Bit 5	Zustand C6
Easy_600_Funktionsrelais[15]		Bit 6	Zustand C7
Easy_600_Funktionsrelais[16]		Bit 7	Zustand C8

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_600_Funktionsrelais[17]	Octet 3	Bit 0	Zustand 71
Easy_600_Funktionsrelais[18]		Bit 1	Zustand 72
Easy_600_Funktionsrelais[19]		Bit 2	Zustand 73
Easy_600_Funktionsrelais[20]		Bit 3	Zustand 74
		Bit 4	nicht genutzt
		Bit 5	nicht genutzt
		Bit 6	nicht genutzt
		Bit 7	nicht genutzt
Easy_600_Funktionsrelais[25]	Octet 4	Bit 0	Zustand A1
Easy_600_Funktionsrelais[26]		Bit 1	Zustand A2
Easy_600_Funktionsrelais[27]		Bit 2	Zustand A3
Easy_600_Funktionsrelais[28]		Bit 3	Zustand A4
Easy_600_Funktionsrelais[29]		Bit 4	Zustand A5
Easy_600_Funktionsrelais[30]		Bit 5	Zustand A6
Easy_600_Funktionsrelais[31]		Bit 6	Zustand A7
Easy_600_Funktionsrelais[32]		Bit 7	Zustand A8



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekt „Easy 600 Tasten“

Dieses Objekt enthält den aktuellen Zustand der Tasten des easy600.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der Tasten (P1 bis P4, ESC, DEL, ALT, OK) des easy600 zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 0
- Index ist gleich 200.

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 1 Octet. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_600_Tasten	ARRAY [1...8] OF BOOL oder BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 182: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_600_Tasten[1]	Octet 1	Bit 0	Zustand P1
Easy_600_Tasten[2]		Bit 1	Zustand P2
Easy_600_Tasten[3]		Bit 2	Zustand P3
Easy_600_Tasten[4]		Bit 3	Zustand P4
Easy_600_Tasten[5]		Bit 4	Zustand ESC
Easy_600_Tasten[6]		Bit 5	Zustand OK
Easy_600_Tasten[7]		Bit 6	Zustand DEL
Easy_600_Tasten[8]		Bit 7	Zustand ALT



Ein gesetztes Bit entspricht einer gedrückten Taste, ein nicht gesetztes Bit entspricht einer nicht gedrückten Taste.

Objekt „Easy Link Eingangsdaten“

Dieses Objekt enthält den aktuellen Zustand der Eingangsdaten auf dem easyLink eines easy-Grundgerätes.

Dieses Objekt ist lesbar (Read) und schreibbar (Write). Es kann nur von einem DPV1-Master der Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um

- sich über den aktuellen Zustand der easyLink-Eingangsdaten (R1 bis R16), die vom DP-Master der Klasse 1 in den zyklischen DP-Daten übertragen werden, zu informieren
- dem angeschlossenen easy-Grundgerät neue Eingangsdaten auf dem easyLink zu übergeben.



Sie können nur dann auf dieses Objekt schreibend zugreifen, wenn **kein** DP-Master der Klasse 1 mit dem angesprochenen EASY204-DP Modul zyklischen Datenaustausch betreibt.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 0
- Index ist gleich 98.

Länge des Objektes

Die Länge der auszulesenden bzw. der zu schreibenden Daten beträgt 2 Octets. Beachten Sie, dass sie keine andere Länge beim Aufruf des Dienstes „Write“ eintragen oder eine kleinere Länge beim Aufruf des Dienstes „Read“, da sie ansonsten eine Fehlermeldung erhalten.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierendem System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_Link_Eingaenge	ARRAY [1...16] OF BOOL oder ARRAY [1...2] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 183: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_Link_Eingaenge[1]	Octet 1	Bit 0	Zustand R1
Easy_Link_Eingaenge[2]		Bit 1	Zustand R2
Easy_Link_Eingaenge[3]		Bit 2	Zustand R3
Easy_Link_Eingaenge[4]		Bit 3	Zustand R4
Easy_Link_Eingaenge[5]		Bit 4	Zustand R5
Easy_Link_Eingaenge[6]		Bit 5	Zustand R6
Easy_Link_Eingaenge[7]		Bit 6	Zustand R7
Easy_Link_Eingaenge[8]		Bit 7	Zustand R8
Easy_Link_Eingaenge[9]	Octet 2	Bit 0	Zustand R9
Easy_Link_Eingaenge[10]		Bit 1	Zustand R10
Easy_Link_Eingaenge[11]		Bit 2	Zustand R11
Easy_Link_Eingaenge[12]		Bit 3	Zustand R12
Easy_Link_Eingaenge[13]		Bit 4	Zustand R13
Easy_Link_Eingaenge[14]		Bit 5	Zustand R14
Easy_Link_Eingaenge[15]		Bit 6	Zustand R15
Easy_Link_Eingaenge[16]		Bit 7	Zustand R16



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekt „Easy Link Ausgangsdaten“

Dieses Objekt enthält den aktuellen Zustand der Ausgangsdaten auf dem easyLink eines easy-Grundgerätes.

Dieses Objekt ist nur lesbar (Read). Es kann nur von einem DPV1-Master der Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der easyLink-Ausgangsdaten (S1 bis S8), die zum DP-Master der Klasse 1 in den zyklischen DP-Daten übertragen werden, zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 0
- Index ist gleich 99.

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 1 Octet. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_Link_Ausgaenge	ARRAY [1...8] OF BOOL oder BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 184: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_Link_Ausgaenge[1]	Octet 1	Bit 0	Zustand S1
Easy_Link_Ausgaenge[2]		Bit 1	Zustand S2
Easy_Link_Ausgaenge[3]		Bit 2	Zustand S3
Easy_Link_Ausgaenge[4]		Bit 3	Zustand S4
Easy_Link_Ausgaenge[5]		Bit 4	Zustand S5
Easy_Link_Ausgaenge[6]		Bit 5	Zustand S6
Easy_Link_Ausgaenge[7]		Bit 6	Zustand S7
Easy_Link_Ausgaenge[8]		Bit 7	Zustand S8



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

**Funktionsbaustein-Daten
lesen/schreiben****Objekte „Easy 600 Parameter Zeitrelais T1 bis
Zeitrelais T8“**

Diese 8 Objekte enthalten jeweils die Parameter der 8 Zeitrelais des easy600.

Diese Objekte sind lesbar (Read) und schreibbar (Write). Sie können von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie diese Objekte, um

- sich über die aktuellen Parameter der Zeitrelais (T1 bis T8) eines easy600 zu informieren
- neue Parameter für die Zeitrelais eines easy600 zu übergeben.

Adressierung der Objekte

Für die Adressierung der Objekte verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 0
- Index ist gleich 211 bis 218 für die Zeitrelais T1 bis T8.

Länge der Objekte

Die Länge der auszulesenden Daten beträgt 6 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Die Länge der zu schreibenden Daten beträgt 5 Octets. Tragen Sie keine andere Länge beim Aufruf des Dienstes „Write“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definitionen (Beispiel) für die Objekte

In einem IEC 61131-3 basierenden System deklarieren Sie für das Lesen der Parameter eines Zeitrelais folgende Variable (Datenbaustein):

Name	Datentyp
Easy_600_Zeitrelais_Istwert	STRUCT
Funktion	BYTE oder USINT
Zeitbereich	BYTE oder USINT
Parameter_Menue	BYTE oder USINT
Steuerzustand	BYTE oder USINT
Aktual_Wert	WORD oder UINT

In einem IEC 61131-3 basierenden System deklarieren Sie für das Schreiben der Parameter eines Zeitrelais folgende Variable (Datenbaustein):

Name	Datentyp
Easy_600_Zeitrelais_Sollwert	ARRAY [1...5] OF BYTE oder ARRAY [1...5] OF USINT

Dateninhalte der Objekte

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte beim Lesen der Parameter eines Zeitrelais. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 185: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung
Easy_600_Zeitrelais_Istwert.Funktion	Octet 1	Funktionsweise des Zeitrelais → Tabelle 187
Easy_600_Zeitrelais_Istwert.Zeitbereich	Octet 2	Zeitbereich des Zeitrelais → Tabelle 188
Easy_600_Zeitrelais_Istwert.Parameter_Menue	Octet 3	Zeitrelais im Parameter-Menü → Tabelle 189
Easy_600_Zeitrelais_Istwert.Steuerzustand	Octet 4	Steuerzustand des Zeitrelais → Tabelle 190
Easy_600_Zeitrelais_Istwert.Zeitwert	Octet 5, 6	Zeitwert des Zeitrelais: [1/100 s] bei Zeitbereich Sekunden [s] bei Zeitbereich Minuten: Sekunden [min] bei Zeitbereich Stunden: Minuten



Beachten Sie beim Zugriff auf den Dateninhalt „Zeitwert des Zeitrelais“ das im PROFIBUS-DP verwendete Motorola-Kodierungsformat (Octet N: High-Byte, Octet N+1: Low-Byte). Sollte das Datenverarbeitungsformat in Ihrem DP-Master-System hiervon abweichen und die DPV1-Zugriffsbefehle keine automatische Konvertierung beinhalten, müssen Sie die notwendige Konvertierung in Ihrem Anwendungsprogramm selber vornehmen. Beachten Sie diesbezüglich die Dokumentation Ihres DP-Master-Systems.

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte beim Schreiben der Parameter eines Zeitrelais. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 186: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung
Easy_600_Zeitrelais_Sollwert[1]	Octet 1	Funktionsweise des Zeitrelais → Tabelle 187
Easy_600_Zeitrelais_Sollwert[2]	Octet 2	Zeitbereich des Zeitrelais → Tabelle 188
Easy_600_Zeitrelais_Sollwert[3]	Octet 3	Zeitrelais im Parameter-Menü → Tabelle 189
Easy_600_Zeitrelais_Sollwert[4]	Octet 4	Zeitsollwert des Zeitrelais
Easy_600_Zeitrelais_Sollwert[5]	Octet 5	→ Tabelle 191

Tabelle 168 zeigt die Kodierung der Dateninhalte für die Funktionsweise des Zeitrelais.

Tabelle 187: Funktionsweise des Zeitrelais

Wert (hex)	Funktion
00	Ansprechverzögert
01	Rückfallverzögert
02	Ansprechverzögert mit Zufallsschalten
03	Rückfallverzögert mit Zufallsschalten
04	Impulsformend
05	Blinkend

Tabelle 188: Zeitbereich des Zeitrelais

Wert (hex)	Zeitbereich
00	Sekunden
01	Minuten:Sekunden
02	Stunden:Sekunden

Tabelle 189: Parameter-Menü des Zeitrelais

Wert (hex)	Parameter-Menü
00	Zeitrelais erscheint im Parameter-Menü
01	Zeitrelais erscheint nicht im Parameter-Menü

Tabelle 190: Steuerzustand des Zeitrelais

Wert (hex)	Steuerzustand
00	Zeitrelais wird vom Betriebssystem nicht verwendet
01	Zeitrelais wird vom Betriebssystem verwendet

Tabelle 191: Zeitsollwert des Zeitrelais

Datenposition		Bedeutung	Erläuterung
Octet 4	Bit 4 bis 7	Wert niederwertigere Einheit, 10er Stelle	1/100 Sekunden (00 bis 99) bei Zeitbereich Sekunden; Sekunden (00 bis 59)
	Bit 0 bis 3	Wert niederwertigere Einheit, 1er Stelle	bei Zeitbereich Minuten:Sekunden; Minuten (00 bis 59) bei Zeitbereich Stunden:Minuten
Octet 5	Bit 4 bis 7	Wert höherwertigere Einheit, 10er Stelle	Sekunden (00 bis 99) bei Zeitbereich Sekunden; Minuten (00 bis 99)
	Bit 0 bis 3	Wert höherwertigere Einheit, 1er Stelle	bei Zeitbereich Minuten:Sekunden; Stunden (00 bis 99) bei Zeitbereich Stunden:Minuten

Objekte „Easy 600 Parameter Zähler C1 bis Zähler C8“

Diese 8 Objekte enthalten jeweils die Parameter der 8 Zähler des easy600.

Diese Objekte sind lesbar (Read) und schreibbar (Write). Sie können von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie diese Objekte, um

- sich über die aktuellen Parameter der Zähler (C1 bis C8) eines easy600 zu informieren
- neue Parameter für die Zähler eines easy600 zu übergeben.

Adressierung der Objekte

Für die Adressierung der Objekte verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 0
- Index ist gleich 221 bis 228 für die Zähler C1 bis C8

Länge der Objekte

Die Länge der auszulesenden Daten beträgt 2 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Die Länge der zu schreibenden Daten beträgt 3 Octets. Tragen Sie keine andere Länge beim Aufruf des Dienstes „Write“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definitionen (Beispiel) für die Objekte

In einem IEC 61131-3 basierendem System deklarieren Sie für das Lesen der Parameter eines Zählers folgende Variable (Datenbaustein):

Name	Datentyp
Easy_600_Zaehler_Istwert	WORD oder UINT

In einem IEC 61131-3 basierenden System deklarieren Sie für das Schreiben der Parameter eines Zählers folgende Variable (Datenbaustein):

Name	Datentyp
Easy_600_Zaehler_Sollwert	STRUCT
Zaehlwert	WORD oder UINT
Parameter_Menue	BYTE oder USINT

Dateninhalte der Objekte

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte beim Lesen der Parameter eines Zählers. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 192: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung
Easy_600_Zaehler_Istwert	Octet 1 und 2	Istwert des Zählers



Beachten Sie beim Zugriff auf den Dateninhalt „Istwert des Zeitrelais“ das im PROFIBUS-DP verwendete Motorola-Kodierungsformat (Octet N: High-Byte, Octet N+1: Low-Byte). Sollte das Datenverarbeitungsformat in Ihrem DP-Master-System hiervon abweichen und die DPV1-Zugriffsbefehle keine automatische Konvertierung beinhalten, müssen Sie die notwendige Konvertierung in Ihrem Anwendungsprogramm selber vornehmen. Beachten Sie diesbezüglich die Dokumentation Ihres DP-Master-Systems.

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte beim Schreiben der Parameter eines Zählers. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 193: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Daten-position	Bedeutung
Easy_600_Zaehler_Sollwert.Zaehlwert	Octet 1 und 2	Sollwert des Zaehlers
Easy_600_Zaehler_Sollwert.Parameter_Menue	Octet 3	Zähler im Parameter-Menü → Tabelle 194



Beachten Sie beim Zugriff auf den Dateninhalt „Sollwert des Zählers“ das im PROFIBUS-DP verwendete Motorola-Kodierungsformat (Octet N: High-Byte, Octet N+1: Low-Byte). Sollte das Datenverarbeitungsformat in Ihrem DP-Master-System hiervon abweichen und die DPV1-Zugriffsbefehle keine automatische Konvertierung beinhalten, müssen Sie die notwendige Konvertierung in Ihrem Anwendungsprogramm selber vornehmen. Beachten Sie diesbezüglich die Dokumentation Ihres DP-Master-Systems.

Tabelle 194: Parameter-Menü des Zählers

Wert (hex)	Parameter-Menü
00	Zähler soll im Parameter-Menü erscheinen
01	Zähler soll nicht im Parameter-Menü erscheinen

Objekte „Easy 600 Parameter Analogwertvergleichers A1 bis A8“

Diese 8 Objekte enthalten jeweils die Parameter der 8 Analogwertvergleichers des easy600.

Diese Objekte sind nur schreibbar (Write). Sie können von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie diese Objekte um neue Parameter für die Analogwertvergleichers eines easy600 zu übergeben

Adressierung der Objekte

Für die Adressierung der Objekte verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 0
- Index ist gleich 231 bis 238 für die Analogwertvergleichers A1 bis A8.

Länge der Objekte

Die Länge der zu schreibenden Daten beträgt 3 Octets. Tragen Sie keine andere Länge beim Aufruf des Dienstes „Write“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definitionen (Beispiel) für die Objekte

In einem IEC 61131-3 basierenden System deklarieren Sie für das Schreiben der Parameter eines Analogwertvergleichers folgende Variable (Datenbaustein):

Name	Datentyp
Easy_600_Analogwert_Vgl	ARRAY [1...3] OF BYTE oder ARRAY [1...3] OF USINT

Dateninhalte der Objekte

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte beim Schreiben der Parameter eines Analogwertvergleichers. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 195: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung
Easy_600_Analogwert_Vgl[1]	Octet 1	Funktion des Analogwertvergleichers → Tabelle 196
Easy_600_Analogwert_Vgl[2]	Octet 2	Vergleichswert
Easy_600_Analogwert_Vgl[3]	Octet 3	Analogwertvergleich im Parameter-Menü → Tabelle 197

Tabelle 196: Funktion des Analogwertvergleichers

Wert (hex)	Funktion
00	Vergleich auf $I7 \cong I8$
01	Vergleich auf $I7 \cong I8$
02	Vergleich auf $I7 \cong$ Vergleichswert
03	Vergleich auf $I7 \cong$ Vergleichswert
04	Vergleich auf $I8 \cong$ Vergleichswert
05	Vergleich auf $I8 \cong$ Vergleichswert

Tabelle 197: Parameter-Menü des Analogwertvergleichers

Wert (hex)	Parameter-Menü
00	Zähler soll im Parameter-Menü erscheinen
01	Zähler soll nicht im Parameter-Menü erscheinen

Objekte „Easy 600 Parameter Wochenschaltuhr 01 Kanal A bis Wochenschaltuhr 04 Kanal D“

Diese 16 Objekte enthalten jeweils die Parameter der 4 Kanäle der 4 Wochenschaltuhren des easy600.

Diese Objekte sind lesbar (Read) und schreibbar (Write). Sie können von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie diese Objekte, um

- sich über die aktuellen Parameter der Kanäle der Wochenschaltuhren (01 bis 04) eines easy600 zu informieren
- neue Parameter für die Kanäle der Wochenschaltuhren eines easy600 zu übergeben.

Adressierung der Objekte

Für die Adressierung der Objekte verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 0
- Index ist gleich 239 bis 242 für Kanal A bis Kanal D der Wochenschaltuhr 01
- Index ist gleich 243 bis 246 für Kanal A bis Kanal D der Wochenschaltuhr 02
- Index ist gleich 247 bis 250 für Kanal A bis Kanal D der Wochenschaltuhr 03
- Index ist gleich 251 bis 254 für Kanal A bis Kanal D der Wochenschaltuhr 04.

Länge der Objekte

Die Länge der auszulesenden bzw. der zu schreibenden Daten beträgt 8 Octets. Tragen Sie keine andere Länge beim Aufruf des Dienstes „Write“ ein oder eine kleinere Länge beim Aufruf des Dienstes „Read“; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definitionen (Beispiel) für die Objekte

In einem IEC 61131-3 basierendem System deklarieren Sie für das Lesen und Schreiben der Parameter eines Kanals einer Wochenschaltuhr folgende Variable (Datenbaustein):

Name	Datentyp
Easy_600_Kanal_Schaltuhr	ARRAY [1...8] OF BYTE oder ARRAY [1...8] OF USINT

Dateninhalte der Objekte

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte beim Lesen oder Schreiben der Parameter eines Kanals einer Wochenschaltuhr. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 198: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_600_Kanal_Schaltuhr[1]	Octet 1		Wochentag beginnend → Tabelle 199
Easy_600_Kanal_Schaltuhr[2]	Octet 2		Wochentag endend → Tabelle 199
Easy_600_Kanal_Schaltuhr[3]	Octet 3	Bit 4 bis 7	Minute Einschalten 10er Stelle
		Bit 0 bis 3	Minute Einschalten 1er Stelle
Easy_600_Kanal_Schaltuhr[4]	Octet 4	Bit 4 bis 7	Stunde Einschalten 10er Stelle
		Bit 0 bis 3	Stunde Einschalten 1er Stelle
Easy_600_Kanal_Schaltuhr[5]	Octet 5	Bit 4 bis 7	Minute Ausschalten 10er Stelle
		Bit 0 bis 3	Minute Ausschalten 1er Stelle
Easy_600_Kanal_Schaltuhr[6]	Octet 6	Bit 4 bis 7	Stunde Ausschalten 10er Stelle
		Bit 0 bis 3	Stunde Ausschalten 1er Stelle
Easy_600_Kanal_Schaltuhr[7]	Octet 7		Schaltzeitpunkt → Tabelle 200
Easy_600_Kanal_Schaltuhr[8]	Octet 8		Kanal der Schaltuhr im Parameter-Menü → Tabelle 201

Tabelle 199: Wochentag des Kanals einer Wochenschaltuhr

Wert (hex)	Bedeutung
00	Keiner eingestellt
01	Montag
02	Dienstag
03	Mittwoch
04	Donnerstag
05	Freitag
06	Samstag
07	Sonntag

Tabelle 200: Schaltzeitpunkt des Kanals einer Wochenschaltuhr

Wert (hex)	Bedeutung
00	Schaltzeitpunkt On < Off
01	Schaltzeitpunkt On > Off

Tabelle 201: Parameter-Menü des Kanals einer Wochenschaltuhr

Wert (hex)	Bedeutung
00	Kanal der Schaltuhr erscheint im Parameter-Menü
01	Kanal der Schaltuhr erscheint nicht im Parameter-Menü

**Datum und Uhrzeit lesen/
schreiben**

Objekt „Easy Uhr“

Dieses Objekt enthält Datum und Uhrzeit der Echtzeituhr eines easy-Grundgerätes.

Dieses Objekt ist lesbar (Read) und schreibbar (Write). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um

- die aktuelle Uhrzeit und das aktuelle Datum eines easy600 auszulesen
- Uhrzeit und Datum eines easy600 einzustellen.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 0
- Index ist gleich 97.

Länge des Objektes

Die Länge der auszulesenden bzw. der zu schreibenden Daten beträgt 8 Octets. Tragen Sie keine andere Länge beim Aufruf des Dienstes „Write“ ein oder eine kleinere Länge beim Aufruf des Dienstes „Read“; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definitionen (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie für das Lesen und Schreiben der Echtzeituhr folgende Variable (Datenbaustein):

Name	Datentyp
Easy_Uhr	DATE_AND_TIME

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes beim Lesen oder Schreiben. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 202: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung
Easy_Uhr	Octet 1 bis 8	Uhrzeit und Datum des Easy → Tabelle 203

Tabelle 203: BCD-Kodierung des Dateninhalts Uhrzeit und Datum

Datenposition		Bedeutung	Erläuterung
Octet 1	Bit 4 bis 7	Jahreszahl, 10er Stelle	Wert 90 entspricht 1990 Wert 99 entspricht 1999 Wert 00 entspricht 2000 Wert 89 entspricht 2089
	Bit 0 bis 3	Jahreszahl, 1er Stelle	
Octet 2	Bit 4 bis 7	Monatszahl, 10er Stelle	
	Bit 0 bis 3	Monatszahl, 1er Stelle	
Octet 3	Bit 4 bis 7	Tageszahl, 10er Stelle	
	Bit 0 bis 3	Tageszahl, 1er Stelle	
Octet 4	Bit 4 bis 7	Stundenzahl, 10er Stelle	
	Bit 0 bis 3	Stundenzahl, 1er Stelle	
Octet 5	Bit 4 bis 7	Minutenzahl, 10er Stelle	
	Bit 0 bis 3	Minutenzahl, 1er Stelle	
Octet 6	Bit 4 bis 7	Sekundenzahl, 10er Stelle	
	Bit 0 bis 3	Sekundenzahl, 1er Stelle	

Daten-position		Bedeutung	Erläuterung
Octet 7	Bit 4 bis 7	Millisekundenzahl, 100er Stelle	
	Bit 0 bis 3	Millisekundenzahl, 10er Stelle	
Octet 8	Bit 4 bis 7	Millisekundenzahl, 1er Stelle	
	Bit 0 bis 3	Wochentag	



Die Uhr im easy600 verwendet intern kein Jahr, Monat, Tag und keine Millisekunden. Daher sind beim Lesen des Objektes die entsprechenden Datenfelder mit „0“ belegt. Beim Schreiben des Objektes werden Werte in diesen Datenfeldern ignoriert.

Sommer-/Winterzeit lesen/ schreiben Objekte Easy 600 Sommer-/Winterzeit Einstellung

Dieses Objekt enthält die Sommer-/Winterzeit Einstellung des easy600.

Dieses Objekt ist lesbar (Read) und schreibbar (Write). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um

- die aktuelle Sommer-/Winterzeit Einstellung der Uhr eines easy600 auszulesen
- die Uhr eines easy600 auf Sommer-/Winterzeit einzustellen.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 0
- Index ist gleich 210.

Länge des Objektes

Die Länge der auszulesenden bzw. der zu schreibenden Daten beträgt 1 Octet. Tragen Sie keine andere Länge beim Aufruf des Dienstes „Write“ ein oder eine kleinere Länge beim Aufruf des Dienstes „Read“; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definitionen (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie für das Lesen und Schreiben der Sommer-/Winterzeit Einstellung folgende Variable (Datenbaustein):

Name	Datentyp
Easy_600_Uhr_Einstellung	BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes beim Lesen oder Schreiben. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 204: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bezeichnung
Easy_600_Uhr_Einstellung	Octet 1	Einstellung der easy-Uhr auf Sommer-/Winterzeit → Tabelle 205

Tabelle 205: Kodierung Einstellung Sommer-/Winterzeit easy600

Wert (hex)	Wirkung
00	Easy Uhr ist/wird auf Winterzeit eingestellt
01	Easy Uhr ist/wird auf Sommerzeit eingestellt

11 Prozessdaten easy700 (DPV1)

Mit Hilfe der PROFIBUS-DPV1-Funktionalität des EASY204-DP (ab Geräteversion 07) können Sie ohne großen Programmieraufwand von einem DPV1-Master der Klasse 1 oder Klasse 2 auf folgende Datenbereiche eines easy700 zugreifen:

- Identifikation
- Betriebsart
- Abbild
- Funktionsbausteine
- Datum und Uhrzeit
- Sommer-/Winterzeit

Die Daten aus diesen Bereichen befinden sich in Process Data Objekten (Datensätzen) und können mit Hilfe der DPV1- Dienste „Read“ und „Write“ gelesen und/oder geschrieben werden.

Objektübersicht

Die folgende Übersicht zeigt alle im EASY204-DP enthaltenen Objekte, die für ein angeschlossenes easy700 bereitgestellt werden.



Die Attribute API, Slot Number und Index bilden die Adressinformation für den DP-Master für die Ansprache eines Objektes. Weitere Erläuterungen hierzu entnehmen Sie dem folgenden Abschnitt.

Tabelle 206: Process Data Objekte im EASY204-DP für easy700

Objekt-Name	API	Slot Number	Index	Datenlänge (Octets)	lesbar (R) schreibbar (W)	→ Seite
Easy Betriebsart	0	0	100	1	R/W	265
Easy Identifikation	0	1	100	50	R	263
Abbilddaten						
Easy 700/800 Eingänge (I1 – I16)	0	1	250	2	R	304
Easy 700/800 Analogeingang IA1 bis IA4	0	1	197 - 200	2	R	306
Easy 700/800 Ausgänge (Q1 – Q8)	0	1	251	1	R	308
Easy 700/800 Tasten	0	1	249	1	R	310
Easy 700 Analogwertvergleichler	0	2	1	2	R	312
Easy 700 Wochenschaltuhren	0	2	2	1	R	314
Easy 700 Jahresschaltuhren	0	2	3	1	R	316
Easy 700 Masterreset	0	2	4	1	R	318
Easy 700 Textbausteine	0	2	5	2	R	320
Easy 700 Zeitglieder	0	2	6	2	R	322
Easy 700 Zähler	0	2	7	2	R	324
Easy 700 Betriebsstundenzähler	0	2	8	1	R	326

Objekt-Name	API	Slot Number	Index	Datenlänge (Octets)	lesbar (R) schreibbar (W)	→ Seite
Easy 700 Aktualwert Merker M	0	2	11	2	R	328
Easy 700 Aktualwert Merker N	0	2	12	2	R	330
Easy 700 Neuer Wert Merker M1 bis M16	0	2	101 - 116	1	W	334
Easy 700 Neuer Wert Merker N1 bis N16	0	0	117 - 132	1	W	334
Easy Link Eingangsdaten (R1 – R16)	0	0	98	2	R/W	277
Easy Link Ausgangsdaten (S1 – S8)	0	0	99	1	R	279
Funktionsbausteine						
Easy 700/800 Schreiben Daten Funktionsbaustein	0	1	97	8	W	341
Easy 700/800 Auswahl Daten Funktionsbaustein	0	1	99	4	W	337
Easy 700/800 Lesen Daten Funktionsbaustein	0	1	98	4	R	339
Easy Uhr	0	0	97	8	R/W	294
Easy 700 Sommer-/ Winterzeit Einstellung	0	2	9	16	R/W	356

Zugriff auf Objekte

Der Zugriff auf die Process Data Objekte im EASY204-DP mittels der DPV1-Dienste „Read“ und „Write“ erfolgt mit Hilfe der vom DP-Master-System hierfür bereitgestellten Funktionen. Beachten Sie diesbezüglich die Dokumentation des Herstellers. In der Regel werden für den Zugriff Funktionsbausteine bereitgestellt. In IEC 61131-3 basierten Systemen werden oft die von der PROFIBUS-Nutzerorganisation in der Richtlinie 2.182 definierten Funktionsbausteine RDREC (Read) und WRREC (Write) angeboten, die einen optimalen Zugriff auch auf komplexe Datenstrukturen ermöglichen.

Für die Ansprache der Objekte benötigen Sie folgende Angaben:

- Die Adresse der lokalen DPV1-Master-Anschaltung
- Die Teilnehmeradresse des anzusprechenden EASY204-DP
- Den Identifizier der Applikation (API) im EASY204-DP (Angabe nur bei DPV1-Master der Klasse 2 erforderlich)
- Das zu adressierende Modul des EASY204-DP (Slot Number)
- Die Adresse (Index) des gewünschten Process Data Objektes im adressierten Modul des EASY204-DP
- Die Datenlänge des gewünschten Process Data Objektes
- Eine definierte Variable (Speicherbereich) in der lokalen Anwendung, der die ausgelesenen Daten zugewiesen werden sollen oder die die zu schreibenden Daten enthält.

Die Adresse der lokalen DPV1-Master-Anschaltung entnehmen Sie der Topologie Ihres Master-Systems. Die Teilnehmeradresse des anzusprechenden EASY204-DP entnehmen Sie der PROFIBUS-Topologie.

Für das EASY204-DP ist für alle Process Data Objekte der API (Angabe nur bei DPV1-Master der Klasse 2 erforderlich) mit 0 anzugeben.

- Die Slot Number, den Index, die Datenlänge und eine beispielhafte Variablendefinition (Deklaration) für ein IEC 61131-3 basierendes System für jedes im EASY204-DP für ein angeschlossenes easy700 vorhandene Process Data Objekt finden Sie in den folgenden Abschnitten.

Identifikation lesen**Objekt „Easy Identifikation“**

Die Beschreibung dieses Objektes ist identisch zum easy600 (DPV1), → Seite 263.

**Betriebsart lesen/
schreiben****Objekt „Easy Betriebsart“**

Die Beschreibung dieses Objektes ist identisch zum easy600 (DPV1), → Seite 265.

Abbild lesen/schreiben

Objekt „Easy 700/800 Eingänge“

Dieses Objekt enthält den aktuellen Zustand der Eingänge eines easy700 oder eines easy800/MFD.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der Digital-Eingänge (I1 bis I16) eines easy700 oder easy800/MFD zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 250

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 2 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_800_Eingaenge	ARRAY [1...16] OF BOOL oder ARRAY [1...2] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 207: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_700_800_Eingaenge[1]	Octet 1	Bit 0	Zustand I1
Easy_700_800_Eingaenge[2]		Bit 1	Zustand I2
Easy_700_800_Eingaenge[3]		Bit 2	Zustand I3
Easy_700_800_Eingaenge[4]		Bit 3	Zustand I4
Easy_700_800_Eingaenge[5]		Bit 4	Zustand I5
Easy_700_800_Eingaenge[6]		Bit 5	Zustand I6
Easy_700_800_Eingaenge[7]		Bit 6	Zustand I7
Easy_700_800_Eingaenge[8]		Bit 7	Zustand I8
Easy_700_800_Eingaenge[9]	Octet 2	Bit 0	Zustand I9
Easy_700_800_Eingaenge[10]		Bit 1	Zustand I10
Easy_700_800_Eingaenge[11]		Bit 2	Zustand I11
Easy_700_800_Eingaenge[12]		Bit 3	Zustand I12
Easy_700_800_Eingaenge[13]		Bit 4	Zustand I13
Easy_700_800_Eingaenge[14]		Bit 5	Zustand I14
Easy_700_800_Eingaenge[15]		Bit 6	Zustand I15
Easy_700_800_Eingaenge[16]		Bit 7	Zustand I16



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekte „Easy 700/800 Analogeingänge IA1 bis IA4“

Diese Objekte enthalten den aktuellen Zustand der Analog-Eingänge eines easy700 oder eines easy800/MFD.

Diese Objekte sind nur lesbar (Read). Sie können von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie diese Objekte, um sich über den aktuellen Zustand der Analog-Eingänge (IA1 bis IA4) eines easy700 oder easy800/MFD zu informieren.

Adressierung der Objekte

Für die Adressierung der Objekte verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 197 bis 200 für die Analogeingänge IA1 bis IA4

Länge der Objekte

Die Länge der auszulesenden Daten beträgt 2 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für die Objekte

In einem IEC 61131-3 basierendem System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_800_Analog_I	Word oder UINT

Dateninhalte der Objekte

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 208: Adresslage und Bedeutung der Dateninhalte der Objekte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung
Easy_700_800_Analog_I	Octet 1 und 2	Wert des Analogeingangs IA



Beachten Sie beim Zugriff auf den Dateninhalt „Wert des Analogeingangs IA“ das im PROFIBUS-DP verwendete Motorola-Kodierungsformat (Octet N: High-Byte, Octet N+1: Low-Byte). Sollte das Datenverarbeitungsformat in Ihrem DP-Master-System hiervon abweichen und die DPV1-Zugriffsbefehle keine automatische Konvertierung beinhalten, müssen Sie die notwendige Konvertierung in Ihrem Anwendungsprogramm selber vornehmen. Beachten Sie diesbezüglich die Dokumentation Ihres DP-Master-Systems.

Objekt „Easy 700/800 Ausgänge“

Dieses Objekt enthält den aktuellen Zustand der Ausgänge eines easy700 oder eines easy800/MFD.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der Digital-Ausgänge (Q1 bis Q8) eines easy700 oder easy800/MFD zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 251

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 1 Octet. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_800_Ausgaenge	ARRAY [1...8] OF BOOL oder BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 209: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
	Octet	Bit	
Easy_700_800_Ausgaenge[1]	Octet 1	Bit 0	Zustand Q1
Easy_700_800_Ausgaenge[2]		Bit 1	Zustand Q2
Easy_700_800_Ausgaenge[3]		Bit 2	Zustand Q3
Easy_700_800_Ausgaenge[4]		Bit 3	Zustand Q4
Easy_700_800_Ausgaenge[5]		Bit 4	Zustand Q5
Easy_700_800_Ausgaenge[6]		Bit 5	Zustand Q6
Easy_700_800_Ausgaenge[7]		Bit 6	Zustand Q7
Easy_700_800_Ausgaenge[8]		Bit 7	Zustand Q8



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekt „Easy 700/800 Tasten“

Dieses Objekt enthält den aktuellen Zustand der Tasten eines easy700 oder eines easy800/MFD.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der Tasten (P1 bis P4) eines easy700 oder easy800/MFD zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 249

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 1 Octet. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_800_Tasten	ARRAY [1...8] OF BOOL oder BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 210: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_700_800_Tasten[1]	Octet 1	Bit 0	Zustand P1
Easy_700_800_Tasten[2]		Bit 1	Zustand P2
Easy_700_800_Tasten[3]		Bit 2	Zustand P3
Easy_700_800_Tasten[4]		Bit 3	Zustand P4
		Bit 4	nicht genutzt
		Bit 5	nicht genutzt
		Bit 6	nicht genutzt
		Bit 7	nicht genutzt



Ein gesetztes Bit entspricht einer gedrückten Taste, ein nicht gesetztes Bit entspricht einer nicht gedrückten Taste.

Objekt „Easy 700 Analogwertvergleich“

Dieses Objekt enthält den aktuellen Zustand der Analogwertvergleich eines easy700.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der Analogwertvergleich (A1 bis A16) eines easy700 zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 2
- Index ist gleich 1

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 2 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_Analogwert_Vgl	ARRAY [1...16] OF BOOL oder ARRAY [1...2] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 211: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_700_Analogwert_Vgl[1]	Octet 1	Bit 0	Zustand A1
Easy_700_Analogwert_Vgl[2]		Bit 1	Zustand A2
Easy_700_Analogwert_Vgl[3]		Bit 2	Zustand A3
Easy_700_Analogwert_Vgl[4]		Bit 3	Zustand A4
Easy_700_Analogwert_Vgl[5]		Bit 4	Zustand A5
Easy_700_Analogwert_Vgl[6]		Bit 5	Zustand A6
Easy_700_Analogwert_Vgl[7]		Bit 6	Zustand A7
Easy_700_Analogwert_Vgl[8]		Bit 7	Zustand A8
Easy_700_Analogwert_Vgl[9]	Octet 2	Bit 0	Zustand A9
Easy_700_Analogwert_Vgl[10]		Bit 1	Zustand A10
Easy_700_Analogwert_Vgl[11]		Bit 2	Zustand A11
Easy_700_Analogwert_Vgl[12]		Bit 3	Zustand A12
Easy_700_Analogwert_Vgl[13]		Bit 4	Zustand A13
Easy_700_Analogwert_Vgl[14]		Bit 5	Zustand A14
Easy_700_Analogwert_Vgl[15]		Bit 6	Zustand A15
Easy_700_Analogwert_Vgl[16]		Bit 7	Zustand A16



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekt „Easy 700 Wochenschaltuhren“

Dieses Objekt enthält den aktuellen Zustand der Wochenschaltuhren eines easy700.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der Wochenschaltuhren (☐1 bis ☐8) eines easy700 zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 2
- Index ist gleich 2

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 1 Octet. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_Wochenschaltuhr	ARRAY [1...8] OF BOOL oder BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 212: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_700_Wochenschaltuhr[1]	Octet 1	Bit 0	Zustand ⓪ 1
Easy_700_Wochenschaltuhr[2]		Bit 1	Zustand ⓪ 2
Easy_700_Wochenschaltuhr[3]		Bit 2	Zustand ⓪ 3
Easy_700_Wochenschaltuhr[4]		Bit 3	Zustand ⓪ 4
Easy_700_Wochenschaltuhr[5]		Bit 4	Zustand ⓪ 5
Easy_700_Wochenschaltuhr[6]		Bit 5	Zustand ⓪ 6
Easy_700_Wochenschaltuhr[7]		Bit 6	Zustand ⓪ 7
Easy_700_Wochenschaltuhr[8]		Bit 7	Zustand ⓪ 8



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekt „Easy 700 Jahresschaltuhren“

Dieses Objekt enthält den aktuellen Zustand der Jahresschaltuhren eines easy700.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der Jahresschaltuhren (Y1 bis Y8) eines easy700 zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 2
- Index ist gleich 3

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 1 Octet. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_Jahresschaltuhr	ARRAY [1...8] OF BOOL oder BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 213: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung	
Easy_700_Jahresschaltuhr[1]	Octet 1	Bit 0	Zustand Y1
Easy_700_Jahresschaltuhr[2]		Bit 1	Zustand Y2
Easy_700_Jahresschaltuhr[3]		Bit 2	Zustand Y3
Easy_700_Jahresschaltuhr[4]		Bit 3	Zustand Y4
Easy_700_Jahresschaltuhr[5]		Bit 4	Zustand Y5
Easy_700_Jahresschaltuhr[6]		Bit 5	Zustand Y6
Easy_700_Jahresschaltuhr[7]		Bit 6	Zustand Y7
Easy_700_Jahresschaltuhr[8]		Bit 7	Zustand Y8



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekt „Easy 700 Masterreset“

Dieses Objekt enthält den aktuellen Zustand der Masterreset eines easy700.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der Masterreset (M1 bis M4) eines easy700 zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 2
- Index ist gleich 4

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 1 Octet. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_Masterreset	ARRAY [1...8] OF BOOL oder BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 214: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_700_Masterreset[1]	Octet 1	Bit 0	Zustand Z1
Easy_700_Masterreset[2]		Bit 1	Zustand Z2
Easy_700_Masterreset[3]		Bit 2	Zustand Z3
Easy_700_Masterreset[4]		Bit 3	Zustand Z4
		Bit 4	nicht genutzt
		Bit 5	nicht genutzt
		Bit 6	nicht genutzt
		Bit 7	nicht genutzt



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekt „Easy 700 Textbausteine“

Dieses Objekt enthält den aktuellen Zustand der Textbausteine eines easy700.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der Textbausteine (D1 bis D16) eines easy700 zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 2
- Index ist gleich 5

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 2 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_Textbaustein	ARRAY [1...16] OF BOOL oder ARRAY [1...2] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 215: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_700_Textbaustein[1]	Octet 1	Bit 0	Zustand D1
Easy_700_Textbaustein[2]		Bit 1	Zustand D2
Easy_700_Textbaustein[3]		Bit 2	Zustand D3
Easy_700_Textbaustein[4]		Bit 3	Zustand D4
Easy_700_Textbaustein[5]		Bit 4	Zustand D5
Easy_700_Textbaustein[6]		Bit 5	Zustand D6
Easy_700_Textbaustein[7]		Bit 6	Zustand D7
Easy_700_Textbaustein[8]		Bit 7	Zustand D8
Easy_700_Textbaustein[9]	Octet 2	Bit 0	Zustand D9
Easy_700_Textbaustein[10]		Bit 1	Zustand D10
Easy_700_Textbaustein[11]		Bit 2	Zustand D11
Easy_700_Textbaustein[12]		Bit 3	Zustand D12
Easy_700_Textbaustein[13]		Bit 4	Zustand D13
Easy_700_Textbaustein[14]		Bit 5	Zustand D14
Easy_700_Textbaustein[15]		Bit 6	Zustand D15
Easy_700_Textbaustein[16]		Bit 7	Zustand D16



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekt „Easy 700 Zeitglieder“

Dieses Objekt enthält den aktuellen Zustand der Zeitglieder eines easy700.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der Zeitglieder (T1 bis T16) eines easy700 zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 2
- Index ist gleich 6

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 2 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_Zeitglied	ARRAY [1...16] OF BOOL oder ARRAY [1...2] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 216: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_700_Zeitglied[1]	Octet 1	Bit 0	Zustand T1
Easy_700_Zeitglied[2]		Bit 1	Zustand T2
Easy_700_Zeitglied[3]		Bit 2	Zustand T3
Easy_700_Zeitglied[4]		Bit 3	Zustand T4
Easy_700_Zeitglied[5]		Bit 4	Zustand T5
Easy_700_Zeitglied[6]		Bit 5	Zustand T6
Easy_700_Zeitglied[7]		Bit 6	Zustand T7
Easy_700_Zeitglied[8]		Bit 7	Zustand T8
Easy_700_Zeitglied[9]	Octet 2	Bit 0	Zustand T9
Easy_700_Zeitglied[10]		Bit 1	Zustand T10
Easy_700_Zeitglied[11]		Bit 2	Zustand T11
Easy_700_Zeitglied[12]		Bit 3	Zustand T12
Easy_700_Zeitglied[13]		Bit 4	Zustand T13
Easy_700_Zeitglied[14]		Bit 5	Zustand T14
Easy_700_Zeitglied[15]		Bit 6	Zustand T15
Easy_700_Zeitglied[16]		Bit 7	Zustand T16



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekt „Easy 700 Zähler“

Dieses Objekt enthält den aktuellen Zustand der Zähler eines easy700.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der Zähler (C1 bis C16) eines easy700 zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 2
- Index ist gleich 7

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 2 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierendem System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_Zaehler	ARRAY [1...16] OF BOOL oder ARRAY [1...2] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 217: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_700_Zaehler[1]	Octet 1	Bit 0	Zustand C1
Easy_700_Zaehler[2]		Bit 1	Zustand C2
Easy_700_Zaehler[3]		Bit 2	Zustand C3
Easy_700_Zaehler[4]		Bit 3	Zustand C4
Easy_700_Zaehler[5]		Bit 4	Zustand C5
Easy_700_Zaehler[6]		Bit 5	Zustand C6
Easy_700_Zaehler[7]		Bit 6	Zustand C7
Easy_700_Zaehler[8]		Bit 7	Zustand C8
Easy_700_Zaehler[9]	Octet 2	Bit 0	Zustand C9
Easy_700_Zaehler[10]		Bit 1	Zustand C10
Easy_700_Zaehler[11]		Bit 2	Zustand C11
Easy_700_Zaehler[12]		Bit 3	Zustand C12
Easy_700_Zaehler[13]		Bit 4	Zustand C13
Easy_700_Zaehler[14]		Bit 5	Zustand C14
Easy_700_Zaehler[15]		Bit 6	Zustand C15
Easy_700_Zaehler[16]		Bit 7	Zustand C16



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekt „Easy 700 Betriebsstundenzähler“

Dieses Objekt enthält den aktuellen Zustand der Betriebsstundenzähler eines easy700.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der Betriebsstundenzähler (O1 bis O4) eines easy700 zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 2
- Index ist gleich 8

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 1 Octet. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_Betriebsstunden	ARRAY [1...8] OF BOOL oder BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 218: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_700_Betriebsstunden[1]	Octet 1	Bit 0	Zustand O1
Easy_700_Betriebsstunden[2]		Bit 1	Zustand O2
Easy_700_Betriebsstunden[3]		Bit 2	Zustand O3
Easy_700_Betriebsstunden[4]		Bit 3	Zustand O4
		Bit 4	nicht genutzt
		Bit 5	nicht genutzt
		Bit 6	nicht genutzt
		Bit 7	nicht genutzt



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekt „Easy 700 Aktualwert Merker M“

Dieses Objekt enthält den aktuellen Zustand der Merker M eines easy700.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der Merker M (M1 bis M16) eines easy700 zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 2
- Index ist gleich 11

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 2 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_M_Merker	ARRAY [1...16] OF BOOL oder ARRAY [1...2] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 219: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_700_M_Merker[1]	Octet 1	Bit 0	Zustand M1
Easy_700_M_Merker[2]		Bit 1	Zustand M2
Easy_700_M_Merker[3]		Bit 2	Zustand M3
Easy_700_M_Merker[4]		Bit 3	Zustand M4
Easy_700_M_Merker[5]		Bit 4	Zustand M5
Easy_700_M_Merker[6]		Bit 5	Zustand M6
Easy_700_M_Merker[7]		Bit 6	Zustand M7
Easy_700_M_Merker[8]		Bit 7	Zustand M8
Easy_700_M_Merker[9]	Octet 2	Bit 0	Zustand M9
Easy_700_M_Merker[10]		Bit 1	Zustand M10
Easy_700_M_Merker[11]		Bit 2	Zustand M11
Easy_700_M_Merker[12]		Bit 3	Zustand M12
Easy_700_M_Merker[13]		Bit 4	Zustand M13
Easy_700_M_Merker[14]		Bit 5	Zustand M14
Easy_700_M_Merker[15]		Bit 6	Zustand M15
Easy_700_M_Merker[16]		Bit 7	Zustand M16



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekt „Easy 700 Aktualwert Merker N“

Dieses Objekt enthält den aktuellen Zustand der Merker N eines easy700.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der Merker N (N1 bis N16) eines easy700 zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 2
- Index ist gleich 12

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 2 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_N_Merker	ARRAY [1...16] OF BOOL oder ARRAY [1...2] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 220: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_700_N_Merker[1]	Octet 1	Bit 0	Zustand N1
Easy_700_N_Merker[2]		Bit 1	Zustand N2
Easy_700_N_Merker[3]		Bit 2	Zustand N3
Easy_700_N_Merker[4]		Bit 3	Zustand N4
Easy_700_N_Merker[5]		Bit 4	Zustand N5
Easy_700_N_Merker[6]		Bit 5	Zustand N6
Easy_700_N_Merker[7]		Bit 6	Zustand N7
Easy_700_N_Merker[8]		Bit 7	Zustand N8
Easy_700_N_Merker[9]	Octet 2	Bit 0	Zustand N9
Easy_700_N_Merker[10]		Bit 1	Zustand N10
Easy_700_N_Merker[11]		Bit 2	Zustand N11
Easy_700_N_Merker[12]		Bit 3	Zustand N12
Easy_700_N_Merker[13]		Bit 4	Zustand N13
Easy_700_N_Merker[14]		Bit 5	Zustand N14
Easy_700_N_Merker[15]		Bit 6	Zustand N15
Easy_700_N_Merker[16]		Bit 7	Zustand N16



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekte „Easy 700 Neuer Wert Merker M1 bis M16“

Diese Objekte ermöglichen es, den Zustand der Merker M eines easy700 zu steuern.

Diese Objekte sind nur schreibbar (Write). Sie können von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie diese Objekte, um den aktuellen Zustand der Merker M (M1 bis M16) eines easy700 zu steuern.

Adressierung der Objekte

Für die Adressierung der Objekte verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 2
- Index ist gleich 101 bis 116 für die Merker M1 bis M16

Länge der Objekte

Die Länge der zu schreibenden Daten beträgt 1 Octet. Tragen Sie beim Aufruf des Dienstes „Write“ keine andere Länge ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für die Objekte

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_Soll_M_Merker	BYTE oder USINT

Dateninhalte der Objekte

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 221: Adresslage und Bedeutung der Dateninhalte der Objekte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung
Easy_700_Soll_M_Merker	Octet 1	Sollwert für M-Merker: 00 = FALSE FF = TRUE

Objekte „Easy 700 Neuer Wert Merker N1 bis N16“

Diese Objekte ermöglichen die Steuerung des Zustands der Merker N eines easy700.

Diese Objekte sind nur schreibbar (Write). Sie können von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie diese Objekte um den aktuellen Zustand der Merker N (N1 bis N16) eines easy700 zu steuern.

Adressierung der Objekte

Für die Adressierung der Objekte verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 2
- Index ist gleich 117 bis 132 für die Merker N1 bis N16

Länge der Objekte

Die Länge der zu schreibenden Daten beträgt 1 Octet. Tragen Sie beim Aufruf des Dienstes „Write“ keine andere Länge ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für die Objekte

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_Soll_N_Merker	BYTE oder USINT

Dateninhalte der Objekte

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 222: Adresslage und Bedeutung der Dateninhalte der Objekte

Variablen-Zugriff (Beispiel)	Daten- position	Bedeutung
Easy_700_Soll_N_Merker	Octet 1	Sollwert für N-Merker: 00 = FALSE FF = TRUE

Objekt „Easy Link Eingangsdaten“

Die Beschreibung dieses Objektes ist identisch zu easy600 (DPV1), → Seite 277.

Objekt „Easy Link Ausgangsdaten“

Die Beschreibung dieses Objektes ist identisch zu easy600 (DPV1), → Seite 279.

**Funktionsbaustein-Daten
lesen/schreiben****Verfahrensweise**

Das Lesen und Schreiben der Daten von Funktionsbausteinen ist mit Hilfe der folgenden generischen Objekte einfach möglich:

- „Easy 700/800 Auswahl Daten Funktionsbaustein“
- „Easy 700/800 Lesen Daten Funktionsbaustein“
- „Easy 700/800 Schreiben Daten Funktionsbaustein“ .

Um Daten eines Funktionsbausteins zu lesen, sind zwei Objektzugriffe notwendig:

- Auswahl der gewünschten Daten eines Funktionsbausteins für den nachfolgenden Auslesevorgang:
→ Objekt „Easy 700/800 Auswahl Daten Funktionsbaustein“, Seite 337.
- Lesen der ausgewählten Daten des Funktionsbausteins:
→ Objekt „Easy 700/800 Lesen Daten Funktionsbaustein“, Seite 339.

Um Daten eines Funktionsbausteins zu schreiben, ist nur ein Objektzugriff notwendig. Dabei werden gleichzeitig die gewünschten Daten eines Funktionsbausteines ausgewählt und neue Werte für diese übergeben.

→ Objekt „Easy 700/800 Schreiben Daten Funktionsbaustein“, Seite 341.

Objekt „Easy 700/800 Auswahl Daten Funktionsbaustein“

Mit Hilfe dieses Objekts wählen Sie die Daten eines Funktionsbausteins eines easy700 aus, die Sie später lesen möchten.

Dieses Objekt ist nur schreibbar (Write). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um die gewünschten Daten eines Funktionsbausteins eines easy700 für einen oder mehrere nachfolgende Lesezugriffe mit Hilfe des Objekts „Easy 700/800 Lesen Daten Funktionsbaustein“ auszuwählen. Die Auswahl bleibt für alle nachfolgenden Lesezugriffe so lange gültig, bis Sie neue Daten durch Schreiben auf das Objekt „Easy 700/800 Auswahl Daten Funktionsbaustein“ ausgewählt haben.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 99

Länge des Objektes

Die Länge der zu schreibenden Daten beträgt 4 Octets. Tragen Sie beim Aufruf des Dienstes „Write“ keine andere Länge ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierendem System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_800_Adr_FB_Read	ARRAY [1...4] OF BYTE oder ARRAY [1...4] OF USINT

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 223: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung	
Easy_700_800_Adr_FB_Read[1]	Octet 1	Befehl	8D _{hex} für easy700
			B2 _{hex} für easy800/MFD
Easy_700_800_Adr_FB_Read[2]	Octet 2	Typ	→ Beschreibung der Funktionsbausteine
Easy_700_800_Adr_FB_Read[3]	Octet 3	Instanz	
Easy_700_800_Adr_FB_Read[4]	Octet 4	Index	

Octet 2 bis Octet 4 (Typ, Instanz, Index) identifizieren, welche Funktionsbaustein-Daten zukünftig mit Hilfe des Objektes „Easy 700/800 Lesen Daten Funktionsbaustein“ gelesen werden sollen. Diese Angaben sind spezifisch für die jeweiligen Funktionsbaustein-Daten und stehen bei der entsprechenden Funktionsbaustein-Beschreibung.

Objekt „Easy 700/800 Lesen Daten Funktionsbaustein“

Mit Hilfe dieses Objekts lesen Sie die gewünschten Daten eines Funktionsbausteins eines easy700 aus.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um die gewünschten Daten eines Funktionsbausteins eines easy700 auslesen.



Die gewünschten Daten müssen Sie vorab durch Schreiben des Objekts „Easy 700/800 Auswahl Daten Funktionsbaustein“ ausgewählt haben.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 98

Länge des Objektes

Die Länge der zu lesenden Daten beträgt 8 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_800_FB_Read	ARRAY [1..8] OF BYTE oder ARRAY [1..8] OF USINT

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 224: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Daten-position	Bedeutung	
Easy_700_800_FB_Read[1]	Octet 1	Befehl	89 _{hex} für easy700 92 _{hex} für easy800/MFD
Easy_700_800_FB_Read[2]	Octet 2	Typ	→ Beschreibung der Funktionsbausteine
Easy_700_800_FB_Read[3]	Octet 3	Instanz	
Easy_700_800_FB_Read[4]	Octet 4	Index	
Easy_700_800_FB_Read[5]	Octet 5	Data 1	
Easy_700_800_FB_Read[6]	Octet 6	Data 2	
Easy_700_800_FB_Read[7]	Octet 7	Data 3	
Easy_700_800_FB_Read[8]	Octet 8	Data 4	

Octet 2 bis Octet 4 (Typ, Instanz, Index) identifizieren, welche Funktionsbaustein-Daten ausgelesen wurden.



Diese Angabe ist identisch mit der Auswahl, die Sie durch Schreiben des Objekts „Easy 700/800 Auswahl Daten Funktionsbaustein“ zuletzt getroffen haben.

Octet 5 bis Octet 8 (Data 1 bis Data 4) enthalten den aktuellen Wert der gelesenen Funktionsbaustein-Daten. Aufbau und Bedeutung dieser Werte sind spezifisch für die jeweils gelesenen Funktionsbaustein-Daten. Lesen Sie diese in der Beschreibung des jeweiligen Funktionsbausteins nach (ab Seite 343).

Objekt „Easy 700/800 Schreiben Daten Funktionsbaustein“

Mit Hilfe dieses Objekts wählen Sie die gewünschten Daten eines Funktionsbausteins eines easy700 aus und übergeben diesen neue Werte.

Dieses Objekt ist nur schreibbar (Write). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um den gewünschten Daten eines Funktionsbausteins eines easy700 neue Werte zuzuweisen.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 97

Länge des Objektes

Die Länge der zu schreibenden Daten beträgt 8 Octets. Tragen Sie beim Aufruf des Dienstes „Write“ keine andere Länge ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_800_FB_Write	ARRAY [1..8] OF BYTE oder ARRAY [1..8] OF USINT

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 225: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung	
Easy_700_800_FB_Write[1]	Octet 1	Befehl	8D _{hex} für easy700
			B2 _{hex} für easy800/MFD
Easy_700_800_FB_Write[2]	Octet 2	Typ	→ Beschreibung der Funktionsbausteine
Easy_700_800_FB_Write[3]	Octet 3	Instanz	
Easy_700_800_FB_Write[4]	Octet 4	Index	
Easy_700_800_FB_Write[5]	Octet 5	Data 1	
Easy_700_800_FB_Write[6]	Octet 6	Data 2	
Easy_700_800_FB_Write[7]	Octet 7	Data 3	
Easy_700_800_FB_Write[8]	Octet 8	Data 4	

Octet 2 bis Octet 4 (Typ, Instanz, Index) identifizieren, welche Funktionsbaustein-Daten geschrieben werden sollen. Die entsprechenden Angaben sind spezifisch für die jeweiligen Funktionsbaustein-Daten und können den folgenden Abschnitten zu dem jeweiligen Funktionsbaustein entnommen werden.

Octet 5 bis Octet 8 (Data 1 bis Data 4) enthalten den Wert für die zu schreibenden Funktionsbaustein-Daten. Der Aufbau und die Bedeutung dieser Werte ist spezifisch für die jeweils zu schreibenden Funktionsbaustein-Daten und kann den folgenden Abschnitten zu dem jeweiligen Funktionsbaustein entnommen werden. Beachten Sie die Angaben in diesen Abschnitten, welche Funktionsbaustein-Daten unter welchen Bedingungen schreibbar sind.

**Analogwertvergleich/Schwellwertschalter:
A1 - A16**

Operanden-Auswahl	Wert (hex)
Type	8D
Instanz ¹⁾	00 - 0F
Index	00 - 07, → Tabelle 226

1) Der Wert 00_{hex} wählt den Analogwertvergleich A1 aus, der Wert 0F_{hex} den Analogwertvergleich A16.

Tabelle 226: Operandenübersicht

Index	Operand	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Parameter	→ Tabelle 227		–	–	R
01	Kontrollbyte	–		→ Tabelle 228	–	R
02	Vergleichswert I1	WORD oder UINT ¹⁾		–	–	R/W ²⁾
03	Vergleichswert I2	WORD oder UINT ¹⁾		–	–	R/W ²⁾
04	Verstärkungsfaktor F1 (I1 = F1*I1)	WORD oder UINT ¹⁾		–	–	R/W ²⁾
05	Verstärkungsfaktor F2 (I2 = F2*I2)	WORD oder UINT ¹⁾		–	–	R/W ²⁾
06	Offset OS (I1 = OS + Istwert an I1)	WORD oder UINT ¹⁾		–	–	R/W ²⁾
07	Schalthytere HY für den Wert I2	WORD oder UINT ¹⁾		–	–	R/W ²⁾

1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 2 enthält High-Byte

2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 227: Index 00 – Parameter

Bedeutung	Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Erscheint im Parametermenü																	
ja/nein																	0/1
Vergleich auf																	
FB nicht benutzt														0	0	0	
EQ (=)														0	0	1	
GE (\geq)														0	1	0	
LE (\leq)														0	1	1	
GT (>)														1	0	0	
LT (<)														1	0	1	
Verwendung als Konstante und somit beschreibbar																	
I1= Konstante														0/1			
F1= Konstante													0/1				
I2= Konstante											0/1						
F2 = Konstante										0/1							
OS = Konstante									0/1								
HY = Konstante								0/1									
Nicht verwendet		0	0	0	0	0	0										

Tabelle 228: Index 01 – Kontrollbyte (Data 3)

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		-	-	-	-	-	-	-	Q1 ¹⁾

1) Zustand „1“, wenn die Vergleichsbedingung erfüllt ist.



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy700-Handbuch (MN05013003Z-DE; frühere Bezeichnung AWB2528-1508D) oder in der easy-Soft-Hilfe.

Zähler C1 bis C16

Operanden-Auswahl	Wert (hex)
Type	8F
Instanz ¹⁾	00 - 0F
Index	00 - 0, → Tabelle 2293

- 1) Der Wert 00_{hex} wählt den Zähler C1 aus, der Wert 0F_{hex} den Zähler C16.

Tabelle 229: Operandenübersicht

Index	Operand	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Parameter	→ Tabelle 230		–	–	R
01	Kontrollbyte	→ Tabelle 231		–	–	R
02	Istwert	WORD oder UINT ¹⁾		–	–	R/W ²⁾
03	Zählersollwert 2	WORD oder UINT ¹⁾		–	–	R/W ²⁾

- Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 2 enthält High-Byte
- Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschriftet ist.

Tabelle 230: Index 00 – Parameter

Bedeutung	Bit	7	6	5	4	3	2	1	0
Erscheint im Parametermenü									
ja/nein									0/1
Zählermodus									
FB nicht benutzt							0	0	
Vor-/Rückwärtszähler (N)							0	1	
Schneller Vor-/Rückwärtszähler (H)							1	0	
Frequenzmesser (F)							1	1	
Verwendung als Konstante und somit beschreibbar									
Zählersollwert S1						0/1			
Nicht benutzte Bits		–	–	–	–				

Tabelle 231: Index 01 – Kontrollbyte

Data 1	Bit	7	6	5	4	3	2	1	0
FB-Ausgang		–	–	–	–	C ⁴⁾	RE ³⁾	D ²⁾	Q ¹⁾

- 1) Schaltkontakt
- 2) Zählrichtungsangabe: Zustand „0“ = vorwärts zählen
Zustand „1“ = rückwärts zählen
- 3) Reset, das Zeitrelais wird zurückgesetzt (Resetspule)
- 4) Zählerspule, zählt bei jeder positiven Flanke



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy700-Handbuch (MN05013003Z-DE; frühere Bezeichnung AWB2528-1508D) oder in der easy-Soft-Hilfe.

Betriebsstundenzähler O1 bis O4

Operanden-Auswahl	Wert (hex)
Type	92
Instanz ¹⁾	00 - 03
Index	00 - 03, → Tabelle 232

- 1) Der Wert 00_{hex} wählt den Betriebsstundenzähler O1 aus, der Wert 03_{hex} den Betriebsstundenzähler O4.

Tabelle 232: Operandenübersicht

Index	Operand	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Parameter	→ Tabelle 233		–	–	R
01	Kontrollbyte	→ Tabelle 234		–	–	R
02	Istwert	WORD oder UINT ¹⁾		–	–	R/W ²⁾
03	Zählersollwert 2	WORD oder UINT ¹⁾		–	–	R/W ²⁾

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 2 enthält High-Byte
- 2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschriftet ist.

Tabelle 233: Index 00 – Parameter

Bedeutung	Bit	7	6	5	4	3	2	1	0
Erscheint im Parametermenü									
ja/nein									0/1
Verwendung im Programm									
Sollwert S1								0/1	
Nicht benutzte Bits		–	–	–	–	–	–		

Tabelle 234: Index 01 – Kontrollbyte

Data 1	Bit	7	6	5	4	3	2	1	0
FB-Ausgang		–	–	–	–	–	RE ³⁾	EN ²⁾	Q1 ¹⁾

- 1) Schaltkontakt
- 2) Enable, das Zeitrelais wird gestartet (Triggerspule)
- 3) Reset, das Zeitrelais wird zurückgesetzt (Resetspule)



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy700-Handbuch (MN05013003Z-DE; frühere Bezeichnung AWB2528-1508D) oder in der easy-Soft-Hilfe.

Zeitrelais T1 bis T16

Operanden-Auswahl	Wert (hex)
Type	92
Instanz ¹⁾	00 - 0F
Index	00 - 04, → Tabelle 235

1) Der Wert 00_{hex} wählt das Zeitrelais T1 aus, der Wert 0F_{hex} das Zeitrelais T16.

Tabelle 235: Operandenübersicht

Index	Operand	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Parameter	→ Tabelle 236		–	–	R
01	Kontrollbyte	→ Tabelle 237		–	–	R
02	Istwert 1	WORD oder UINT ¹⁾		–	–	R/W ²⁾
03	Zeitsollwert 1	WORD oder UINT ¹⁾		–	–	R/W ²⁾
	Zeitsollwert 2	WORD oder UINT ¹⁾		–	–	R/W ²⁾

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 2 enthält High-Byte
- 2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 236: Index 00 – Parameter

Bedeutung	Bit	7	6	5	4	3	2	1	0
Erscheint im Parametermenü									
ja/nein									0/1
Timer-Modus									
Ansprechverzögert						0	0	0	
Rückfallverzögert						0	0	1	
Ansprechverzögert mit Zufalls-Sollwert						0	1	0	
Rückfallverzögert mit Zufalls-Sollwert						0	1	1	
Ansprechverzögert und Rückfallverzögert (zwei Zeitsollwerte)						1	0	0	
Ansprechverzögert und Rückfallverzögert, jeweils mit Zufalls-Sollwert (zwei Zeitsollwerte)						1	0	1	
Impulsgeber						1	1	0	
Blink-Relais (zwei Zeitsollwerte)						1	1	1	
Zeitbasis									
FB nicht benutzt				0	0				
Millisekunde: S				0	1				
Sekunde: M:S				1	0				
Minute: H:M				1	1				
Verwendung als Konstante und somit beschreibbar									
Zeitsollwert S1			0/1						
Zeitsollwert S2		0/1							

Tabelle 237: Index 01 – Kontrollbyte

	Bit	7	6	5	4	3	2	1	0
FB-Ein-/Ausgang Data 3		–	–	–	–	ST ⁴⁾	RE ³⁾	EN ²⁾	Q1 ¹⁾

- 1) Schaltkontakt
- 2) Enable, das Zeitrelais wird gestartet (Triggerspule)
- 3) Reset, das Zeitrelais wird zurückgesetzt (Resetspule)
- 4) Stopp, das Zeitrelais wird gestoppt (Stoppspule)



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy700-Handbuch (MN05013003Z-DE; frühere Bezeichnung AWB2528-1508D) oder in der easy-Soft-Hilfe.

Jahresschaltuhren Y1 bis Y8

Operanden-Auswahl	Wert (hex)
Type	A2
Instanz ¹⁾	00 - 07
Index	→ Tabelle 238

1) Der Wert 00_{hex} wählt die Jahresschaltuhr Y1 aus, der Wert 07_{hex} die Jahresschaltuhr Y8.

Tabelle 238: Operandenübersicht

Index (hex)	Operand	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Parameter	→ Tabelle 239	–	–	–	R
01	Kontrollbyte → Tabelle 240	→ Tabelle 240	–	–	–	R
	Kanal A					
11	Zeitpunkt ein	Tag ¹⁾	Monat ¹⁾	Jahr ²⁾	–	R/W ³⁾
12	Zeitpunkt aus	Tag ¹⁾	Monat ¹⁾	Jahr ²⁾	–	R/W ³⁾
	Kanal B					
21	Zeitpunkt ein	Tag ¹⁾	Monat ¹⁾	Jahr ²⁾	–	R/W ³⁾
22	Zeitpunkt aus	Tag ¹⁾	Monat ¹⁾	Jahr ²⁾	–	R/W ³⁾
	Kanal C					
31	Zeitpunkt ein	Tag ¹⁾	Monat ¹⁾	Jahr ²⁾	–	R/W ³⁾
32	Zeitpunkt aus	Tag ¹⁾	Monat ¹⁾	Jahr ²⁾	–	R/W ³⁾
	Kanal D					
41	Zeitpunkt ein	Tag ¹⁾	Monat ¹⁾	Jahr ²⁾	–	R/W ³⁾
42	Zeitpunkt aus	Tag ¹⁾	Monat ¹⁾	Jahr ²⁾	–	R/W ³⁾

- 1) Wert wird in hexadezimaler Kodierung übergeben
- 2) Wert wird in hexadezimaler Kodierung übergeben; 00 entspricht Jahr 2000
- 3) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 239: Index 00 – Parameter

Bedeutung	Bit	7	6	5	4	3	2	1	0
Erscheint im Parametermenü									
Kanal A									0/1
Kanal B								0/1	
Kanal C							0/1		
Kanal D						0/1			
Nicht benutzte Bits		–	–	–	–				

Tabelle 240: Index 01 – Kontrollbyte

Data 1	Bit	7	6	5	4	3	2	1	0
FB-Ausgang		–	–	–	–	–	–	–	Q1 ¹⁾

1) Zustand „1“, wenn die Zählbedingung erfüllt ist.



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy700-Handbuch (MN05013003Z-DE; frühere Bezeichnung AWB2528-1508D) oder in der easy-Soft-Hilfe.

Wochenschaltuhren 01 bis 08

Operanden-Auswahl	Wert (hex)
Type	A1
Instanz ¹⁾	00 - 07
Index	→ Tabelle 241

1) Der Wert 00_{hex} wählt die Wochenschaltuhr 01 aus, der Wert 07_{hex} die Wochenschaltuhr 08.

Tabelle 241: Operandenübersicht

Index (hex)	Operand	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Parameter	→ Tabelle 242	–	–	–	R
01	Kontrollbyte	→ Tabelle 243	–	–	–	R
	Kanal A					
11	Tag an/aus	Tag ein ¹⁾	Tag aus ¹⁾	–	–	R/W ³⁾
12	Zeit ein	Stunde ²⁾	Minute ²⁾	–	–	R/W ³⁾
13	Zeit aus	Stunde ²⁾	Minute ²⁾	–	–	R/W ³⁾
	Kanal B					
21	Tag an/aus	Tag ein ¹⁾	Tag aus ¹⁾	–	–	R/W ³⁾
22	Zeitpunkt ein	Stunde ²⁾	Minute ²⁾	–	–	R/W ³⁾
23	Zeitpunkt aus	Stunde ²⁾	Minute ²⁾	–	–	R/W ³⁾
	Kanal C					
31	Tag an/aus	Tag ein ¹⁾	Tag aus ¹⁾	–	–	R/W ³⁾
32	Zeitpunkt ein	Stunde ²⁾	Minute ²⁾	–	–	R/W ³⁾
33	Zeitpunkt aus	Stunde ²⁾	Minute ²⁾	–	–	R/W ³⁾

Index (hex)	Operand	Data 1	Data 2	Data 3	Data 4	Read/Write
	Kanal D					
41	Tag an/aus	Tag aus ¹⁾	Monat ¹⁾	–	–	R/W ³⁾
42	Zeitpunkt ein	Minute ²⁾	Monat ¹⁾	–	–	R/W ³⁾
43	Zeitpunkt aus	Minute ²⁾	Monat ¹⁾	–	–	R/W ³⁾

- 1) Wert wird in hexadezimaler Kodierung übergeben; 01 entspricht Sonntag...07 entspricht Samstag
- 2) Wert wird in hexadezimaler Kodierung übergeben
- 3) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 242: Index 00 – Parameter

Bedeutung	Bit 7	6	5	4	3	2	1	0
Erscheint im Parametermenü								
Kanal A								0/1
Kanal B							0/1	
Kanal C						0/1		
Kanal D					0/1			
Nicht benutzte Bits	–	–	–	–				

Tabelle 243: Index 01 – Kontrollbyte

Data 1	Bit 7	6	5	4	3	2	1	0
FB-Ausgang	–	–	–	–	–	–	–	Q1 ¹⁾

- 1) Zustand „1“, wenn die Zählbedingung erfüllt ist.



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy700-Handbuch (MN05013003Z-DE; frühere Bezeichnung AWB2528-1508D) oder in der easy-Soft-Hilfe.

**Datum und Uhrzeit lesen/
schreiben****Objekt „Easy Uhr“**

Die Beschreibung dieses Objektes ist identisch zum easy600 (DPV1), → Seite 294.

**Sommer-/Winterzeit lesen/
schreiben****Objekte „Easy 700 Sommer-/Winterzeit Einstellung“**

Dieses Objekt enthält die Sommer-/Winterzeit Einstellung des easy700.

Dieses Objekt ist lesbar (Read) und schreibbar (Write). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um

- die aktuelle Sommer-/Winterzeit Einstellung der Uhr eines easy700 auszulesen
- eine neue Sommer-/Winterzeit Einstellung der Uhr eines easy700 zu übergeben

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 2
- Index ist gleich 9

Länge des Objektes

Die Länge der auszulesenden bzw. der zu schreibenden Daten beträgt 16 Octets. Tragen Sie beim Aufruf des Dienstes „Write“ keine andere Länge ein und beim Aufruf des Dienstes „Read“ keine kleinere Länge; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definitionen (Beispiel) für das Objekt

In einem IEC 61131-3 basierendem System deklarieren Sie für das Lesen und Schreiben der Sommer-/Winterzeit Einstellung folgende Variable (Datenbaustein):

Name	Datentyp
Easy_700_Uhr_Umstellung	ARRAY [1...16] OF BYTE oder ARRAY [1...16] OF USINT

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes beim Lesen oder Schreiben. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 244: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Daten- position	Bedeutung	Wertebereich (dezimal)
Easy_700_Uhr_Umstellung[1]	Octet 1	Umschaltbedingung → Tabelle 245	0 bis 4
Easy_700_Uhr_Umstellung[2]	Octet 2	Regel 1 für Wechsel auf Sommerzeit → Tabelle 246	0 bis 5
Easy_700_Uhr_Umstellung[3]	Octet 3	Wochentag für Wechsel auf Sommerzeit → Tabelle 247	0 bis 6
Easy_700_Uhr_Umstellung[4]	Octet 4	Regel 2 für Wechsel auf Sommerzeit → Tabelle 248	0 bis 2
Easy_700_Uhr_Umstellung[5]	Octet 5	Tag der Umstellung für Wechsel auf Sommerzeit	1 bis 31 BCD -Kodierung
Easy_700_Uhr_Umstellung[6]	Octet 6	Monat der Umstellung für Wechsel auf Sommerzeit	1 bis 12 BCD-Kodierung
Easy_700_Uhr_Umstellung[7]	Octet 7	Stunde der Umstellung für Wechsel auf Sommerzeit	0 bis 23 BCD-Kodierung
Easy_700_Uhr_Umstellung[8]	Octet 8	Minute der Umstellung für Wechsel auf Sommerzeit	0 bis 59 BCD-Kodierung

Variablen-Zugriff (Beispiel)	Daten- position	Bedeutung	Wertebereich (dezimal)
Easy_700_Uhr_ Umstellung[9]	Octet 9	Zeitdifferenz → Tabelle 249	0 bis 5
Easy_700_Uhr_ Umstellung[10]	Octet 10	Regel 1 für Wechsel auf Winterzeit → Tabelle 246	0 bis 4
Easy_700_Uhr_ Umstellung[11]	Octet 11	Wochentag für Wechsel auf Winterzeit → Tabelle 247	0 bis 5
Easy_700_Uhr_ Umstellung[12]	Octet 12	Regel 2 für Wechsel auf Winterzeit → Tabelle 248	0 bis 6
Easy_700_Uhr_ Umstellung[13]	Octet 13	Tag der Umstellung für Wechsel auf Winterzeit	1 bis 31 BCD -Kodierung
Easy_700_Uhr_ Umstellung[14]	Octet 14	Monat der Umstellung für Wechsel auf Winterzeit	1 bis 12 BCD-Kodierung
Easy_700_Uhr_ Umstellung[15]	Octet 15	Stunde der Umstellung für Wechsel auf Winterzeit	0 bis 23 BCD-Kodierung
Easy_700_Uhr_ Umstellung[16]	Octet 16	Minute der Umstellung für Wechsel auf Winterzeit	0 bis 59 BCD-Kodierung

Octet 1 enthält die Umschaltbedingung für den Wechsel auf Sommer-/Winterzeit, Tabelle 245 zeigt die möglichen Umschaltbedingungen.

Tabelle 245: Umschaltbedingungen für Sommer-/Winterzeit
Wechsel

Wert (hex)	Bedeutung	Hinweis
00	keine Umschaltung	Octet 2 bis 16 sind ohne Bedeutung
01	Umschaltung erfolgt gemäß: <ul style="list-style-type: none"> • Octet 2 bis 9 für Wechsel auf Sommerzeit • Octet 9 bis 16 für Wechsel auf Winterzeit 	
02	Umschaltung erfolgt gemäß Regeln für EU	Octet 2 bis 16 sind ohne Bedeutung
03	Umschaltung erfolgt gemäß Regeln für GB	
04	Umschaltung erfolgt gemäß Regeln für USA	

Ist in Octet 1 der Wert 0 eingetragen, erfolgt keine Umstellung der Uhr auf Sommer-/Winterzeit. Bei Auswahl der Werte 2 bis 4 erfolgt die Umstellung der Uhr automatisch gemäß den gesetzlichen Bestimmungen für das ausgewählte Land.

Wird in Octet 1 der Wert 1 angegeben, erfolgt die Umstellung der Uhr auf Sommer-/Winterzeit gemäß den selbst definierten Regeln in Octet 2 bis 16.

Octet 2 enthält die erste Regel für die Umstellung auf Sommerzeit, Octet 10 für die Umstellung auf Winterzeit. Tabelle 246 zeigt die möglichen Regeln an.

Tabelle 246: Regel 1 für Sommer-/Winterzeit Wechsel

Wert (hex)	Bedeutung	
00	Umschaltung erfolgt an dem Zeitpunkt, der definiert ist in...	... <ul style="list-style-type: none"> • Octet 5 bis 8 für Wechsel auf Sommerzeit • Octet 13 bis 16 für Wechsel auf Winterzeit In diesem Fall sind ohne Bedeutung: <ul style="list-style-type: none"> • Octet 3 und 4 bei Wechsel auf Sommerzeit • Octet 11 und 12 bei Wechsel auf Winterzeit
01	Umschaltung erfolgt am ersten Wochentag, der definiert ist in...	... <ul style="list-style-type: none"> • Octet 3 für Wechsel auf Sommerzeit • Octet 11 für Wechsel auf Winterzeit in Zusammenhang mit der definierten Regel 2 in: <ul style="list-style-type: none"> • Octet 4 bei Wechsel auf Sommerzeit • Octet 12 bei Wechsel auf Winterzeit
02 ¹⁾	Umschaltung erfolgt am zweiten Wochentag, der definiert ist in...	
03 ¹⁾	Umschaltung erfolgt am dritten Wochentag, der definiert ist in...	
04 ¹⁾	Umschaltung erfolgt am vierten Wochentag, der definiert ist in...	
05 ¹⁾	Umschaltung erfolgt am letzten Wochentag, der definiert ist in...	

1) In der Regel 2 muss die Bedeutung „im“ eingetragen sein. Die Regel 2 wird definiert in:

- Octet 4 bei Wechsel auf Sommerzeit
- Octet 12 bei Wechsel auf Winterzeit

Enthält Octet 2 oder Octet 10 einer der Werte 1 bis 5, muss auch die zweite Regel für die Umstellung auf Sommerzeit (Octet 4) oder Winterzeit (Octet 12) definiert werden. Tabelle 247 zeigt die möglichen Regeln an.

Tabelle 247: Regel 2 für Sommer-/Winterzeit Wechsel

Wert (hex)	Bedeutung	Hinweis
00 ¹⁾	Umschaltung erfolgt an dem definierten Wochentag und Monat	Der Wochentag wird definiert im: <ul style="list-style-type: none"> • Octet 3 bei Wechsel auf Sommerzeit • Octet 11 beim Wechsel auf Winterzeit Monat bzw. Datum werden definiert im: <ul style="list-style-type: none"> • Octet 5 bzw. 6 bei Wechsel auf Sommerzeit • Octet 13 bzw. 14 bei Wechsel auf Winterzeit
01	Umschaltung erfolgt an dem definierten Wochentag nach dem definiertem Datum	
02	Umschaltung erfolgt an dem definierten Wochentag vor dem definiertem Datum	

1) In diesem Fall sind ohne Bedeutung:

- Octet 5 bei Wechsel auf Sommerzeit
- Octet 13 bei Wechsel auf Winterzeit

Tabelle 248: Kodierung des Wochentags

Wert (hexadezimal)	Bedeutung
00	Sonntag
01	Montag
02	Dienstag
03	Mittwoch
04	Donnerstag
05	Freitag
06	Samstag

Tabelle 249: Kodierung der Zeitdifferenz (Octet 9)

Wert (hexadezimal)	Zeitdifferenz der Umstellung beträgt
00	30 Minuten
01	1 Stunde
02	90 Minuten
03	2 Stunden
04	150 Minuten
05	3 Stunden

12 Prozessdaten easy800/MFD (DPV1)

Mit Hilfe der PROFIBUS-DPV1-Funktionalität des EASY204-DP (ab Geräteversion 07) können Sie ohne großen Programmieraufwand von einem DPV1-Master der Klasse 1 oder Klasse 2 auf folgende Datenbereiche eines easy800/MFD zugreifen:

- Identifikation
- Betriebsart
- Abbild
- Funktionsbausteine
- Datum und Uhrzeit
- Sommer-/Winterzeit.

Die Daten aus diesen Bereichen befinden sich in Process Data Objekten (Datensätzen) und können mit Hilfe der DPV1- Dienste „Read“ und „Write“ gelesen und/oder geschrieben werden.

Objektübersicht

Die folgende Übersicht enthält alle im EASY204-DP enthaltene Objekte, die für ein angeschlossenes easy800/MFD bereitgestellt werden.



Die Attribute API, Slot Number und Index bilden die Adressinformation für den DP-Master für die Ansprache eines Objektes. Weitere Erläuterungen hierzu entnehmen Sie dem folgenden Abschnitt.

Tabelle 250: Process Data Objekte im EASY204-DP für easy800/MFD

Objekt-Name	API	Slot Number	Index	Datenlänge (Octets)	Read/Write	→ Seite
Easy Betriebsart	0	0	100	1	R/W	265
Easy Identifikation	0	1	100	50	R	263
Abbilddaten						
Easy 700/800 Eingänge (I1 - I16)	0	1	250	2	R	304
Easy 700/800 Analogeingänge IA1 bis IA4	0	1	197 - 200	2	R	306
Easy 700/800 Ausgänge (Q1 - Q8)	0	1	251	1	R	308
Easy 700/800 Tasten	0	1	249	1	R	310
Easy 800 Analogausgang (QA1)	0	1	252	2	R	367
Easy 800 Lokale Diagnose (ID1 - ID16)	0	1	253	2	R	370
Easy 800 Eingänge Netzwerkteilnehmer IW1 bis IW8	0	1	211 - 218	2	R	372
Easy 800 Link-Eingänge Netzwerkteilnehmer RW1 bis RW8	0	1	221 - 228	2	R	374
Easy 800 Ausgänge Netzwerkteilnehmer QW1 bis QW8	0	1	231 - 238	1	R	376
Easy 800 Link-Ausgänge Netzwerkteilnehmer SW1 bis SW8	0	1	241 - 248	1	R	378

Objekt-Name	API	Slot Number	Index	Datenlänge (Octets)	Read/Write	→ Seite
Easy 800 Receive Data Netz RNW1 bis RNW8	0	0	201 - 208	4	R	380
Easy 800 Send Data Netz SNW1 bis SNW8	0	1	201 - 208	4	R	382
Easy 800 Bit Merker M1 bis M96	0	0	1 - 96	1	R/W	384
Easy 800 Byte Merker MB1 bis MB96	0	0	101 - 196	1	R/W	386
Easy 800 Wort Merker MW1 bis MW96	0	1	1 - 96	2	R/W	388
Easy 800 Doppelwort Merker MD1 bis MD96	0	1	101 - 196	4	R/W	390
Easy 800 8 Byte Daten	0	1	229	8	R/W	392
Easy 800 16 Byte Daten	0	1	230	16	R/W	394
Easy 800 32 Byte Daten	0	1	239	32	R/W	396
Easy 800 64 Byte Daten	0	1	240	64	R/W	399
Easy Link Eingangsdaten (R1 - R16)	0	0	98	2	R/W	277
Easy Link Ausgangsdaten (S1 - S8)	0	0	99	1	R	279
Funktionsbausteine						
Easy 700/800 Schreiben Daten Funktionsbaustein	0	1	97	8	W	341
Easy 700/800 Auswahl Daten Funktionsbaustein	0	1	99	4	W	337
Easy 700/800 Lesen Daten Funktionsbaustein	0	1	98	4	R	339
Easy Uhr	0	0	97	8	R/W	294
Easy 800 Sommer-/Winterzeit Einstellung	0	0	209	20	R/W	458

Zugriff auf Objekte

Der Zugriff auf die Process Data Objekte im EASY204-DP mittels der DPV1-Dienste „Read“ und „Write“ erfolgt mit Hilfe der vom DP-Master-System hierfür bereitgestellten Funktionen. Beachten Sie diesbezüglich die Dokumentation des Herstellers. In der Regel werden für den Zugriff Funktionsbausteine bereitgestellt. In IEC 61131-3 basierten Systemen werden oft die von der PROFIBUS-Nutzerorganisation in der Richtlinie 2.182 definierten Funktionsbausteine RDREC (Read) und WRREC (Write) angeboten, die einen optimalen Zugriff auch auf komplexe Datenstrukturen ermöglichen.

Für die Ansprache der Objekte benötigen Sie folgende Angaben:

- Die Adresse der lokalen DPV1-Master-Anschaltung
- Die Teilnehmeradresse des anzusprechenden EASY204-DP
- Den Identifier der Applikation (API) im EASY204-DP (Angabe nur bei DPV1-Master Klasse 2 erforderlich)
- Das zu adressierende Modul des EASY204-DP (Slot Number)
- Die Adresse (Index) des gewünschten Process Data Objektes im adressierten Modul des EASY204-DP
- Die Datenlänge des gewünschten Process Data Objektes
- Eine definierte Variable (Speicherbereich) in der lokalen Anwendung, der die ausgelesenen Daten zugewiesen werden sollen oder die die zu schreibenden Daten enthält.

Die Adresse der lokalen DPV1-Master-Anschaltung entnehmen Sie der Topologie Ihres Master-Systems. Die Teilnehmeradresse des anzusprechenden EASY204-DP entnehmen Sie der PROFIBUS-Topologie.

Für das EASY204-DP ist für alle Process Data Objekte der API (Angabe nur bei DPV1-Master Klasse 2 erforderlich) mit 0 anzugeben.

- Die Slot Number, den Index, die Datenlänge und eine beispielhafte Variablendefinition (Deklaration) für ein IEC 61131-3 basierendes System für jedes im EASY204-DP für ein angeschlossenes easy800/MFD vorhandene Process Data Objekt finden Sie in den folgenden Abschnitten.

Identifikation lesen**Objekt „Easy Identifikation“**

Die Beschreibung dieses Objektes ist identisch zum easy600 (DPV1), → Seite 263.

**Betriebsart lesen/
schreiben****Objekt „Easy Betriebsart“**

Die Beschreibung dieses Objektes ist identisch zum easy600 (DPV1), → Seite 265.

Abbild lesen/schreiben**Objekt „Easy 700/800 Eingänge“**

Die Beschreibung dieses Objektes ist identisch zum easy700 (DPV1), → Seite 304.

Objekte „Easy 700/800 Analogeingänge IA1 bis IA4“

Die Beschreibung dieses Objektes ist identisch zum easy700 (DPV1), → Seite 306.

Objekt „Easy 700/800 Ausgänge“

Die Beschreibung dieses Objektes ist identisch zum easy700 (DPV1), → Seite 308.

Objekt „Easy 700/800 Tasten“

Die Beschreibung dieses Objektes ist identisch zum easy700 (DPV1), → Seite 310.

Objekt „Easy 800 Analogausgang“

Dieses Objekt enthält den aktuellen Zustand des Analog-Ausgangs eines easy800/MFD.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand des Analog-Ausgangs (QA1) eines easy800/MFD zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 252

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 2 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_800_Analogausgang	Word oder UINT

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 251: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung
Easy_800_Analogausgang	Octet 1 und 2	Analogwert von QA1



Beachten Sie beim Zugriff auf den Dateninhalt „Analogwert von QA1“ das im PROFIBUS-DP verwendete Motorola-Kodierungsformat (Octet N: High-Byte, Octet N+1: Low-Byte). Sollte das Datenverarbeitungsformat in Ihrem DP-Master-System hiervon abweichen und die DPV1-Zugriffsbefehle keine automatische Konvertierung beinhalten, müssen Sie die notwendige Konvertierung in Ihrem Anwendungsprogramm selber vornehmen. Beachten Sie diesbezüglich die Dokumentation Ihres DP-Master-Systems.

Objekt „Easy 800 Lokale Diagnose“

Dieses Objekt enthält die aktuelle Diagnose der NET-Verbindungen eines easy800/MFD.

Dieses Objekt ist nur lesbar (Read). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um sich über den aktuellen Zustand der NET-Teilnehmer eines easy800/MFD und den Zustand der Verbindung zum Remote-Teilnehmer (nur MFD) zu informieren.

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 253

Länge des Objektes

Die Länge der auszulesenden Daten beträgt 2 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_800_NET_Diag	ARRAY [1...16] OF BOOL oder ARRAY [1...2] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 252: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_800_NET_Diag[1]	Octet 1	Bit 0	Zustand ID1 (Zustand Netteilnehmer 1)
Easy_800_NET_Diag[2]		Bit 1	Zustand ID2 (Zustand Netteilnehmer 2)
Easy_800_NET_Diag[3]		Bit 2	Zustand ID3 (Zustand Netteilnehmer 3)
Easy_800_NET_Diag[4]		Bit 3	Zustand ID4 (Zustand Netteilnehmer 4)
Easy_800_NET_Diag[5]		Bit 4	Zustand ID5 (Zustand Netteilnehmer 5)
Easy_800_NET_Diag[6]		Bit 5	Zustand ID6 (Zustand Netteilnehmer 6)
Easy_800_NET_Diag[7]		Bit 6	Zustand ID7 (Zustand Netteilnehmer 7)
Easy_800_NET_Diag[8]		Bit 7	Zustand ID8 (Zustand Netteilnehmer 8)
Easy_800_NET_Diag[9]	Octet 2	Bit 0	Zustand ID9 (Remote-Verbindung MFD)
Easy_800_NET_Diag[10]		Bit 1	Nicht verwendet
Easy_800_NET_Diag[11]		Bit 2	Nicht verwendet
Easy_800_NET_Diag[12]		Bit 3	Nicht verwendet
Easy_800_NET_Diag[13]		Bit 4	Nicht verwendet
Easy_800_NET_Diag[14]		Bit 5	Nicht verwendet
Easy_800_NET_Diag[15]		Bit 6	Nicht verwendet
Easy_800_NET_Diag[16]		Bit 7	Nicht verwendet



Ein gesetztes Bit entspricht dem Zustand „Teilnehmer vorhanden“, ein nicht gesetztes Bit entspricht dem Zustand „Teilnehmer nicht vorhanden“.

Objekte „Easy 800 Eingänge Netzwerkteilnehmer IW1 bis IW8“

Diese Objekte enthalten den aktuellen Zustand der Eingänge eines EASYNET-Teilnehmers.

Diese Objekte sind nur lesbar (Read). Sie können von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie diese Objekte, um sich über den aktuellen Zustand der Eingänge (I1 bis I16) eines EASYNET-Teilnehmers zu informieren.

Adressierung der Objekte

Für die Adressierung der Objekte verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 211 bis 218 für die EASYNET-Teilnehmer 1 bis 8

Länge der Objekte

Die Länge der auszulesenden Daten beträgt 2 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für die Objekte

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_800_NET_Eingaenge	ARRAY [1...16] OF BOOL oder ARRAY [1...2] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 253: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
	Octet 1	Bit	
Easy_800_NET_Eingaenge[1]	Octet 1	Bit 0	Zustand I1
Easy_800_NET_Eingaenge[2]		Bit 1	Zustand I2
Easy_800_NET_Eingaenge[3]		Bit 2	Zustand I3
Easy_800_NET_Eingaenge[4]		Bit 3	Zustand I4
Easy_800_NET_Eingaenge[5]		Bit 4	Zustand I5
Easy_800_NET_Eingaenge[6]		Bit 5	Zustand I6
Easy_800_NET_Eingaenge[7]		Bit 6	Zustand I7
Easy_800_NET_Eingaenge[8]		Bit 7	Zustand I8
Easy_800_NET_Eingaenge[9]	Octet 2	Bit 0	Zustand I9
Easy_800_NET_Eingaenge[10]		Bit 1	Zustand I10
Easy_800_NET_Eingaenge[11]		Bit 2	Zustand I11
Easy_800_NET_Eingaenge[12]		Bit 3	Zustand I12
Easy_800_NET_Eingaenge[13]		Bit 4	Zustand I13
Easy_800_NET_Eingaenge[14]		Bit 5	Zustand I14
Easy_800_NET_Eingaenge[15]		Bit 6	Zustand I15
Easy_800_NET_Eingaenge[16]		Bit 7	Zustand I16



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekte „Easy 800 Link-Eingänge Netzwerkteilnehmer RW1 bis RW8“

Diese Objekte enthalten den aktuellen Zustand der Link-Eingänge eines EASYNET-Teilnehmers.

Diese Objekte sind nur lesbar (Read). Sie können von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie diese Objekte, um sich über den aktuellen Zustand der Link-Eingänge (R1 bis R16) eines EASYNET-Teilnehmers zu informieren.

Adressierung der Objekte

Für die Adressierung der Objekte verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 221 bis 228 für die EASYNET-Teilnehmer 1 bis 8

Länge der Objekte

Die Länge der auszulesenden Daten beträgt 2 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für die Objekte

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_800_NET_Link_Eingaenge	ARRAY [1...16] OF BOOL oder ARRAY [1...2] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 254: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_800_NET_Link_Eingaenge[1]	Octet 1	Bit 0	Zustand R1
Easy_800_NET_Link_Eingaenge[2]		Bit 1	Zustand R2
Easy_800_NET_Link_Eingaenge[3]		Bit 2	Zustand R3
Easy_800_NET_Link_Eingaenge[4]		Bit 3	Zustand R4
Easy_800_NET_Link_Eingaenge[5]		Bit 4	Zustand R5
Easy_800_NET_Link_Eingaenge[6]		Bit 5	Zustand R6
Easy_800_NET_Link_Eingaenge[7]		Bit 6	Zustand R7
Easy_800_NET_Link_Eingaenge[8]		Bit 7	Zustand R8
Easy_800_NET_Link_Eingaenge[9]	Octet 2	Bit 0	Zustand R9
Easy_800_NET_Link_Eingaenge[10]		Bit 1	Zustand R10
Easy_800_NET_Link_Eingaenge[11]		Bit 2	Zustand R11
Easy_800_NET_Link_Eingaenge[12]		Bit 3	Zustand R12
Easy_800_NET_Link_Eingaenge[13]		Bit 4	Zustand R13
Easy_800_NET_Link_Eingaenge[14]		Bit 5	Zustand R14
Easy_800_NET_Link_Eingaenge[15]		Bit 6	Zustand R15
Easy_800_NET_Link_Eingaenge[16]		Bit 7	Zustand R16



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekte „Easy 800 Ausgänge Netzwerkteilnehmer SW1 bis SW8“

Diese Objekte enthalten den aktuellen Zustand der Ausgänge eines EASYNET-Teilnehmers.

Diese Objekte sind nur lesbar (Read). Sie können von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie diese Objekte, um sich über den aktuellen Zustand der Ausgänge (Q1 bis Q8) eines EASYNET-Teilnehmers zu informieren.

Adressierung der Objekte

Für die Adressierung der Objekte verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 231 bis 238 für die EASYNET-Teilnehmer 1 bis 8

Länge der Objekte

Die Länge der auszulesenden Daten beträgt 1 Octet. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für die Objekte

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_800_NET_Ausgaenge	ARRAY [1...8] OF BOOL oder BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 255: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_800_NET_Ausgaenge[1]	Octet 1	Bit 0	Zustand Q1
Easy_800_NET_Ausgaenge[2]		Bit 1	Zustand Q2
Easy_800_NET_Ausgaenge[3]		Bit 2	Zustand Q3
Easy_800_NET_Ausgaenge[4]		Bit 3	Zustand Q4
Easy_800_NET_Ausgaenge[5]		Bit 4	Zustand Q5
Easy_800_NET_Ausgaenge[6]		Bit 5	Zustand Q6
Easy_800_NET_Ausgaenge[7]		Bit 6	Zustand Q7
Easy_800_NET_Ausgaenge[8]		Bit 7	Zustand Q8



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekte „Easy 800 Link-Ausgänge Netzwerkteilnehmer SW1 bis SW8“

Diese Objekte enthalten den aktuellen Zustand der Link-Ausgänge eines EASYNET-Teilnehmers.

Diese Objekte sind nur lesbar (Read). Sie können von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie diese Objekte, um sich über den aktuellen Zustand der Link-Ausgänge (S1 bis S8) eines EASYNET-Teilnehmers zu informieren.

Adressierung der Objekte

Für die Adressierung der Objekte verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 241 bis 248 für die EASYNET-Teilnehmer 1 bis 8

Länge der Objekte

Die Länge der auszulesenden Daten beträgt 1 Octet. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für die Objekte

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_800_NET_Link_Ausgaenge	ARRAY [1..8] OF BOOL oder BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 256: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_800_NET_Link_Ausgaenge[1]	Octet 1	Bit 0	Zustand S1
Easy_800_NET_Link_Ausgaenge[2]		Bit 1	Zustand S2
Easy_800_NET_Link_Ausgaenge[3]		Bit 2	Zustand S3
Easy_800_NET_Link_Ausgaenge[4]		Bit 3	Zustand S4
Easy_800_NET_Link_Ausgaenge[5]		Bit 4	Zustand S5
Easy_800_NET_Link_Ausgaenge[6]		Bit 5	Zustand S6
Easy_800_NET_Link_Ausgaenge[7]		Bit 6	Zustand S7
Easy_800_NET_Link_Ausgaenge[8]		Bit 7	Zustand S8



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekte „Easy 800 Receive-Data Netzwerkteilnehmer RNW1 bis RNW8“

Diese Objekte enthalten den aktuellen Zustand der Receive-Data eines EASYNET-Teilnehmers.

Diese Objekte sind nur lesbar (Read). Sie können von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie diese Objekte, um sich über den aktuellen Zustand der Receive-Data (RN1 bis RN32) eines EASYNET-Teilnehmers zu informieren.

Adressierung der Objekte

Für die Adressierung der Objekte verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 0
- Index ist gleich 201 bis 208 für die EASYNET-Teilnehmer 1 bis 8

Länge der Objekte

Die Länge der auszulesenden Daten beträgt 4 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für die Objekte

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_800_NET_Empfangsdaten	ARRAY [1...32] OF BOOL oder ARRAY [1...4] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 257: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung
Easy_800_NET_Empfangsdaten[1]	Octet 1	Bit 0	Zustand RN1
Easy_800_NET_Empfangsdaten[2]		Bit 1	Zustand RN2
...	
Easy_800_NET_Empfangsdaten[7]		Bit 6	Zustand RN7
Easy_800_NET_Empfangsdaten[8]		Bit 7	Zustand RN8
Easy_800_NET_Empfangsdaten[9]	Octet 2	Bit 0	Zustand RN9
Easy_800_NET_Empfangsdaten[10]		Bit 1	Zustand RN10
...	
Easy_800_NET_Empfangsdaten[15]		Bit 6	Zustand RN15
Easy_800_NET_Empfangsdaten[16]		Bit 7	Zustand RN16
Easy_800_NET_Empfangsdaten[17]	Octet 3	Bit 0	Zustand RN17
Easy_800_NET_Empfangsdaten[18]		Bit 1	Zustand RN18
...	
Easy_800_NET_Empfangsdaten[23]		Bit 6	Zustand RN23
Easy_800_NET_Empfangsdaten[24]		Bit 7	Zustand RN24
Easy_800_NET_Empfangsdaten[25]	Octet 4	Bit 0	Zustand RN25
Easy_800_NET_Empfangsdaten[26]		Bit 1	Zustand RN26
...	
Easy_800_NET_Empfangsdaten[31]		Bit 6	Zustand RN31
Easy_800_NET_Empfangsdaten[32]		Bit 7	Zustand RN32



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekte „Easy 800 Send-Data Netzwerkteilnehmer SNW1 bis SNW8“

Diese Objekte enthalten den aktuellen Zustand der Send-Data eines EASYNET-Teilnehmers.

Diese Objekte sind nur lesbar (Read). Sie können von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie diese Objekte, um sich über den aktuellen Zustand der Send-Data (SN1 bis SN32) eines EASYNET-Teilnehmers zu informieren.

Adressierung der Objekte

Für die Adressierung der Objekte verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 201 bis 208 für die EASYNET-Teilnehmer 1 bis 8

Länge der Objekte

Die Länge der auszulesenden Daten beträgt 4 Octets. Tragen Sie keine kleinere Länge beim Aufruf des Dienstes „Read“ ein; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für die Objekte

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_800_NET_Sendedaten	ARRAY [1...32] OF BOOL oder ARRAY [1...4] OF BYTE

Dateninhalte der Objekte

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 258: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition		Bedeutung	
Easy_800_NET_Sendedaten[1]	Octet 1	Bit 0	Zustand SN1	
Easy_800_NET_Sendedaten[2]		Bit 1	Zustand SN2	
...		
Easy_800_NET_Sendedaten[7]		Bit 6	Zustand SN7	
Easy_800_NET_Sendedaten[8]		Bit 7	Zustand SN8	
Easy_800_NET_Sendedaten[9]		Octet 2	Bit 0	Zustand SN9
Easy_800_NET_Sendedaten[10]			Bit 1	Zustand SN10
...		
Easy_800_NET_Sendedaten[15]	Bit 6		Zustand SN15	
Easy_800_NET_Sendedaten[16]	Bit 7		Zustand SN16	
Easy_800_NET_Sendedaten[17]	Octet 3		Bit 0	Zustand SN17
Easy_800_NET_Sendedaten[18]		Bit 1	Zustand SN18	
...		
Easy_800_NET_Sendedaten[23]		Bit 6	Zustand SN23	
Easy_800_NET_Sendedaten[24]		Bit 7	Zustand SN24	
Easy_800_NET_Sendedaten[25]		Octet 4	Bit 0	Zustand SN25
Easy_800_NET_Sendedaten[26]	Bit 1		Zustand SN26	
...	
Easy_800_NET_Sendedaten[31]	Bit 6		Zustand SN31	
Easy_800_NET_Sendedaten[32]	Bit 7		Zustand SN32	



Ein gesetztes Bit entspricht dem Zustand „Ein“, ein nicht gesetztes Bit entspricht dem Zustand „Aus“.

Objekte „Easy 800 Bit Merker M1 bis M96“

Diese Objekte enthalten den aktuellen Zustand der Bit-Merker eines easy800/MFD.

Diese Objekte sind lesbar (Read) und schreibbar (Write). Sie können von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie diese Objekte, um sich

- über den aktuellen Zustand der Bit-Merker M1 bis M96 des easy800 zu informieren (Read)
- oder dem easy800 einen neuen Sollwert für die Bit-Merker M1 bis M96 zu übergeben (Write).

Adressierung der Objekte

Für die Adressierung der Objekte verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 0
- Index ist gleich 1 bis 96 für die Bit-Merker M1 bis M96

Länge der Objekte

Die Länge der auszulesenden bzw. zu schreibenden Daten beträgt 1 Octet. Tragen Sie beim Aufruf des Dienstes „Write“ keine andere Länge ein und beim Aufruf des Dienstes „Read“ keine kleinere Länge; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für die Objekte

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_800_Bit_Merker	BYTE oder USINT

Dateninhalte der Objekte

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 259: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung
Easy_800_Bit__Merker	Octet 1	Wert Bit-Merker: 00 _{hex} = FALSE FF _{hex} = TRUE

Objekte „Easy 800 Byte Merker MB1 bis MB96“

Diese Objekte enthalten den aktuellen Zustand der Byte-Merker eines easy800/MFD.

Diese Objekte sind lesbar (Read) und schreibbar (Write). Sie können von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie diese Objekte, um sich

- über den aktuellen Zustand der Byte-Merker MB1 bis MB96 des easy800 zu informieren (Read)
- oder dem easy800 einen neuen Sollwert für die Byte-Merker MB1 bis MB96 zu übergeben (Write).

Adressierung der Objekte

Für die Adressierung der Objekte verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 0
- Index ist gleich 101 bis 196 für die Byte-Merker MB1 bis MB96

Länge der Objekte

Die Länge der auszulesenden bzw. zu schreibenden Daten beträgt 1 Octet. Tragen Sie beim Aufruf des Dienstes „Write“ keine andere Länge ein und beim Aufruf des Dienstes „Read“ keine kleinere Länge; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für die Objekte

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_800_Byte_Merker	BYTE oder USINT

Dateninhalte der Objekte

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 260: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung
Easy_800_Byte__Merker	Octet 1	Wert Byte-Merker

Objekte „Easy 800 Wort Merker MW1 bis MW96“

Diese Objekte enthalten den aktuellen Zustand der Wort-Merker eines easy800/MFD.

Diese Objekte sind lesbar (Read) und schreibbar (Write). Sie können von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie diese Objekte, um sich

- über den aktuellen Zustand der Wort-Merker MW1 bis MW96 des easy800 zu informieren (Read)
- oder dem easy800 einen neuen Sollwert für die Wort-Merker MW1 bis MW96 zu übergeben (Write).

Adressierung der Objekte

Für die Adressierung der Objekte verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 1 bis 96 für die Wort-Merker MW1 bis MW96

Länge der Objekte

Die Länge der auszulesenden bzw. zu schreibenden Daten beträgt 2 Octets. Tragen Sie beim Aufruf des Dienstes „Write“ keine andere Länge ein und beim Aufruf des Dienstes „Read“ keine kleinere Länge; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für die Objekte

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_800_Wort_Merker	WORD oder UINT

Dateninhalte der Objekte

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 261: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Daten-position	Bedeutung
Easy_800_Wort__Merker	Octet 1 und 2	Wert Wort-Merker



Beachten Sie beim Zugriff auf den Dateninhalt „Wert Wort-Merker“ das im PROFIBUS-DP verwendete Motorola-Kodierungsformat (Octet N: High-Byte, Octet N+1: Low-Byte). Sollte das Datenverarbeitungsformat in Ihrem DP-Master-System hiervon abweichen und die DPV1-Zugriffsbefehle keine automatische Konvertierung beinhalten, müssen Sie die notwendige Konvertierung in Ihrem Anwendungsprogramm selber vornehmen. Beachten Sie diesbezüglich die Dokumentation Ihres DP-Master-Systems.

Objekte „Easy 800 Doppelwort Merker MD1 bis MD96“

Diese Objekte enthalten den aktuellen Zustand der Doppelwort-Merker eines easy800/MFD.

Diese Objekte sind lesbar (Read) und schreibbar (Write). Sie können von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie diese Objekte, um sich

- über den aktuellen Zustand der Doppelwort-Merker MD1 bis MD96 des easy800 zu informieren (Read)
- oder dem easy800 einen neuen Sollwert für die Doppelwort-Merker MD1 bis MD96 zu übergeben (Write).

Adressierung der Objekte

Für die Adressierung der Objekte verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 101 bis 196 für die Wort Merker MD1 bis MD96

Länge der Objekte

Die Länge der auszulesenden bzw. zu schreibenden Daten beträgt 4 Octets. Tragen Sie beim Aufruf des Dienstes „Write“ keine andere Länge ein und beim Aufruf des Dienstes „Read“ keine kleinere Länge; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definition (Beispiel) für die Objekte

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_800_Doppelwort_Merker	DWORD oder UDINT

Dateninhalte der Objekte

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte der Objekte. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 262: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung
Easy_800_Doppelwort__Merker	Octet 1 bis 4	Wert Doppelwort-Merker



Beachten Sie beim Zugriff auf den Dateninhalt „Wert Doppelwort-Merker“ das im PROFIBUS-DP verwendete Motorola-Kodierungsformat (Octet N: High-Byte, Octet N+1: Low-Byte). Sollte das Datenverarbeitungsformat in Ihrem DP-Master-System hiervon abweichen und die DPV1-Zugriffsbefehle keine automatische Konvertierung beinhalten, müssen Sie die notwendige Konvertierung in Ihrem Anwendungsprogramm selber vornehmen. Beachten Sie diesbezüglich die Dokumentation Ihres DP-Master-Systems.

Objekt „Easy 800 8 Byte Daten“

Dieses Objekt dient dazu, bis zu 8 Byte applikationsspezifische Daten zwischen einem PROFIBUS-DPV1-Master und einem easy800/MFD auszutauschen.

Dieses Objekt ist lesbar (Read) und schreibbar (Write). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um

- maximal 8 Byte applikationsspezifische Daten (Inhalt der Doppelwortmerker MD67 und MD68) aus dem easy800 auszulesen (Read)
- oder maximal 8 Byte applikationsspezifische Daten (Inhalt der Doppelwortmerker MD67 und MD68) dem easy800 zu übergeben (Write).

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 229

Länge des Objektes

Die Länge der auszulesenden bzw. zu schreibenden Daten beträgt maximal 8 Octets. Sie können auch eine kleinere Länge (1 bis 7 Octets) beim Aufruf der Dienste „Read“ oder „Write“ eintragen, wenn Sie nicht den gesamten Datenbereich nutzen wollen .

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierendem System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_800_8_Byte_Daten	Applikationsspezifisch, z. B. ARRAY [1...8] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 263: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung
Easy_800_8_Byte_Daten[1]	Octet 1	Wert MD 67 Byte 1
Easy_800_8_Byte_Daten[2]	Octet 2	Wert MD 67 Byte 2
Easy_800_8_Byte_Daten[3]	Octet 3	Wert MD 67 Byte 3
Easy_800_8_Byte_Daten[4]	Octet 4	Wert MD 67 Byte 4
Easy_800_8_Byte_Daten[5]	Octet 5	Wert MD 68 Byte 1
Easy_800_8_Byte_Daten[6]	Octet 6	Wert MD 68 Byte 2
Easy_800_8_Byte_Daten[7]	Octet 7	Wert MD 68 Byte 3
Easy_800_8_Byte_Daten[8]	Octet 8	Wert MD 68 Byte 4

Objekt „Easy 800 16 Byte Daten“

Dieses Objekt dient dazu bis zu 16 Byte applikationsspezifische Daten zwischen einem PROFIBUS-DPV1-Master und einem easy800/MFD auszutauschen.

Dieses Objekt ist lesbar (Read) und schreibbar (Write). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um

- maximal 16 Byte applikationsspezifische Daten (Inhalt der Doppelwortmerker MD69 bis MD72) aus dem easy800 auszulesen (Read)
- oder maximal 16 Byte applikationsspezifische Daten (Inhalt der Doppelwortmerker MD69 bis MD72) dem easy800 zu übergeben (Write).

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 230

Länge des Objektes

Die Länge der auszulesenden bzw. zu schreibenden Daten beträgt maximal 16 Octets. Sie können auch eine kleinere Länge (1 bis 15 Octets) beim Aufruf der Dienste „Read“ oder „Write“ eintragen, wenn Sie nicht den gesamten Datenbereich nutzen wollen .

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_800_16_Byte_Daten	Applikationsspezifisch, z. B. ARRAY [1...16] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 264: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung
Easy_800_16_Byte_Daten[1]	Octet 1	Wert MD 69 Byte 1
Easy_800_16_Byte_Daten[2]	Octet 2	Wert MD 69 Byte 2
Easy_800_16_Byte_Daten[3]	Octet 3	Wert MD 69 Byte 3
Easy_800_16_Byte_Daten[4]	Octet 4	Wert MD 69 Byte 4
Easy_800_16_Byte_Daten[5]	Octet 5	Wert MD 70 Byte 1
Easy_800_16_Byte_Daten[6]	Octet 6	Wert MD 70 Byte 2
Easy_800_16_Byte_Daten[7]	Octet 7	Wert MD 70 Byte 3
Easy_800_16_Byte_Daten[8]	Octet 8	Wert MD 70 Byte 4
Easy_800_16_Byte_Daten[9]	Octet 9	Wert MD 71 Byte 1
Easy_800_16_Byte_Daten[10]	Octet 10	Wert MD 71 Byte 2
Easy_800_16_Byte_Daten[11]	Octet 11	Wert MD 71 Byte 3
Easy_800_16_Byte_Daten[12]	Octet 12	Wert MD 71 Byte 4
Easy_800_16_Byte_Daten[13]	Octet 13	Wert MD 72 Byte 1
Easy_800_16_Byte_Daten[14]	Octet 14	Wert MD 72 Byte 2
Easy_800_16_Byte_Daten[15]	Octet 15	Wert MD 72 Byte 3
Easy_800_16_Byte_Daten[16]	Octet 16	Wert MD 72 Byte 4

Objekt „Easy 800 32 Byte Daten“

Dieses Objekt dient dazu bis zu 32 Byte applikationsspezifische Daten zwischen einem PROFIBUS-DPV1-Master und einem easy800/MFD auszutauschen.

Dieses Objekt ist lesbar (Read) und schreibbar (Write). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um

- maximal 32 Byte applikationsspezifische Daten (Inhalt der Doppelwortmerker MD73 bis MD80) aus dem easy800 auszulesen (Read)
- oder maximal 32 Byte applikationsspezifische Daten (Inhalt der Doppelwortmerker MD73 bis MD80) dem easy800 zu übergeben (Write).

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 239

Länge des Objektes

Die Länge der auszulesenden bzw. zu schreibenden Daten beträgt maximal 32 Octets. Sie können auch eine kleinere Länge (1 bis 31 Octets) beim Aufruf der Dienste „Read“ oder „Write“ eintragen, wenn Sie nicht den gesamten Datenbereich nutzen wollen .

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_800_32_Byte_Daten	Applikationsspezifisch z. B. ARRAY [1...32] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 265: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Daten-position	Bedeutung
Easy_800_32_Byte_Daten[1]	Octet 1	Wert MD 73 Byte 1
Easy_800_32_Byte_Daten[2]	Octet 2	Wert MD 73 Byte 2
Easy_800_32_Byte_Daten[3]	Octet 3	Wert MD 73 Byte 3
Easy_800_32_Byte_Daten[4]	Octet 4	Wert MD 73 Byte 4
Easy_800_32_Byte_Daten[5]	Octet 5	Wert MD 74 Byte 1
Easy_800_32_Byte_Daten[6]	Octet 6	Wert MD 74 Byte 2
Easy_800_32_Byte_Daten[7]	Octet 7	Wert MD 74 Byte 3
Easy_800_32_Byte_Daten[8]	Octet 8	Wert MD 74 Byte 4
Easy_800_32_Byte_Daten[9]	Octet 9	Wert MD 75 Byte 1
Easy_800_32_Byte_Daten[10]	Octet 10	Wert MD 75 Byte 2
Easy_800_32_Byte_Daten[11]	Octet 11	Wert MD 75 Byte 3
Easy_800_32_Byte_Daten[12]	Octet 12	Wert MD 75 Byte 4
Easy_800_32_Byte_Daten[13]	Octet 13	Wert MD 76 Byte 1
Easy_800_32_Byte_Daten[14]	Octet 14	Wert MD 76 Byte 2
Easy_800_32_Byte_Daten[15]	Octet 15	Wert MD 76 Byte 3
Easy_800_32_Byte_Daten[16]	Octet 16	Wert MD 76 Byte 4
Easy_800_32_Byte_Daten[17]	Octet 17	Wert MD 77 Byte 1
Easy_800_32_Byte_Daten[18]	Octet 18	Wert MD 77 Byte 2
Easy_800_32_Byte_Daten[19]	Octet 19	Wert MD 77 Byte 3
Easy_800_32_Byte_Daten[20]	Octet 20	Wert MD 77 Byte 4
Easy_800_32_Byte_Daten[21]	Octet 21	Wert MD 78 Byte 1
Easy_800_32_Byte_Daten[22]	Octet 22	Wert MD 78 Byte 2
Easy_800_32_Byte_Daten[23]	Octet 23	Wert MD 78 Byte 3

Variablen-Zugriff (Beispiel)	Daten- position	Bedeutung
Easy_800_32_Byte_Daten[24]	Octet 24	Wert MD 78 Byte 4
Easy_800_32_Byte_Daten[25]	Octet 25	Wert MD 79 Byte 1
Easy_800_32_Byte_Daten[26]	Octet 26	Wert MD 79 Byte 2
Easy_800_32_Byte_Daten[27]	Octet 27	Wert MD 79 Byte 3
Easy_800_32_Byte_Daten[28]	Octet 28	Wert MD 79 Byte 4
Easy_800_32_Byte_Daten[29]	Octet 29	Wert MD 80 Byte 1
Easy_800_32_Byte_Daten[30]	Octet 30	Wert MD 80 Byte 2
Easy_800_32_Byte_Daten[31]	Octet 31	Wert MD 80 Byte 3
Easy_800_32_Byte_Daten[32]	Octet 32	Wert MD 80 Byte 4

Objekt „Easy 800 64 Byte Daten“

Dieses Objekt dient dazu bis zu 64 Byte applikationsspezifische Daten zwischen einem PROFIBUS-DPV1-Master und einem easy800/MFD auszutauschen.

Dieses Objekt ist lesbar (Read) und schreibbar (Write). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um

- maximal 64 Byte applikationsspezifische Daten (Inhalt der Doppelwortmerker MD81 bis MD97) aus dem easy800 auszulesen (Read)
- oder maximal 64 Byte applikationsspezifische Daten (Inhalt der Doppelwortmerker MD81 bis MD96) dem easy800 zu übergeben (Write).

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 1
- Index ist gleich 240

Länge des Objektes

Die Länge der auszulesenden bzw. zu schreibenden Daten beträgt maximal 64 Octets. Sie können auch eine kleinere Länge (1 bis 63 Octets) beim Aufruf der Dienste „Read“ oder „Write“ eintragen, wenn Sie nicht den gesamten Datenbereich nutzen wollen .

Variablen-Definition (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie folgende Variable (Datenbaustein):

Name	Datentyp
Easy_800_64_Byte_Daten	Applikationsspezifisch z. B. ARRAY [1...64] OF BYTE

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 266: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Datenposition	Bedeutung
Easy_800_64_Byte_Daten[1]	Octet 1	Wert MD 81 Byte 1
Easy_800_64_Byte_Daten[2]	Octet 2	Wert MD 81 Byte 2
Easy_800_64_Byte_Daten[3]	Octet 3	Wert MD 81 Byte 3
Easy_800_64_Byte_Daten[4]	Octet 4	Wert MD 81 Byte 4
Easy_800_64_Byte_Daten[5]	Octet 5	Wert MD 82 Byte 1
Easy_800_64_Byte_Daten[6]	Octet 6	Wert MD 82 Byte 2
Easy_800_64_Byte_Daten[7]	Octet 7	Wert MD 82 Byte 3
Easy_800_64_Byte_Daten[8]	Octet 8	Wert MD 82 Byte 4
Easy_800_64_Byte_Daten[9]	Octet 9	Wert MD 83 Byte 1
Easy_800_64_Byte_Daten[10]	Octet 10	Wert MD 83 Byte 2
Easy_800_64_Byte_Daten[11]	Octet 11	Wert MD 83 Byte 3
Easy_800_64_Byte_Daten[12]	Octet 12	Wert MD 83 Byte 4
Easy_800_64_Byte_Daten[13]	Octet 13	Wert MD 84 Byte 1
Easy_800_64_Byte_Daten[14]	Octet 14	Wert MD 84 Byte 2
Easy_800_64_Byte_Daten[15]	Octet 15	Wert MD 84 Byte 3
Easy_800_64_Byte_Daten[16]	Octet 16	Wert MD 84 Byte 4
Easy_800_64_Byte_Daten[17]	Octet 17	Wert MD 85 Byte 1
Easy_800_64_Byte_Daten[18]	Octet 18	Wert MD 85 Byte 2
Easy_800_64_Byte_Daten[19]	Octet 19	Wert MD 85 Byte 3
Easy_800_64_Byte_Daten[20]	Octet 20	Wert MD 85 Byte 4
Easy_800_64_Byte_Daten[21]	Octet 21	Wert MD 86 Byte 1
Easy_800_64_Byte_Daten[22]	Octet 22	Wert MD 86 Byte 2
Easy_800_64_Byte_Daten[23]	Octet 23	Wert MD 86 Byte 3

Variablen-Zugriff (Beispiel)	Daten-position	Bedeutung
Easy_800_64_Byte_Daten[24]	Octet 24	Wert MD 86 Byte 4
Easy_800_64_Byte_Daten[25]	Octet 25	Wert MD 87 Byte 1
Easy_800_64_Byte_Daten[26]	Octet 26	Wert MD 87 Byte 2
Easy_800_64_Byte_Daten[27]	Octet 27	Wert MD 87 Byte 3
Easy_800_64_Byte_Daten[28]	Octet 28	Wert MD 87 Byte 4
Easy_800_64_Byte_Daten[29]	Octet 29	Wert MD 88 Byte 1
Easy_800_64_Byte_Daten[30]	Octet 30	Wert MD 88 Byte 2
Easy_800_64_Byte_Daten[31]	Octet 31	Wert MD 88 Byte 3
Easy_800_64_Byte_Daten[32]	Octet 32	Wert MD 88 Byte 4
Easy_800_64_Byte_Daten[33]	Octet 33	Wert MD 89 Byte 1
Easy_800_64_Byte_Daten[34]	Octet 34	Wert MD 89 Byte 2
Easy_800_64_Byte_Daten[35]	Octet 35	Wert MD 89 Byte 3
Easy_800_64_Byte_Daten[36]	Octet 36	Wert MD 89 Byte 4
Easy_800_64_Byte_Daten[37]	Octet 37	Wert MD 90 Byte 1
Easy_800_64_Byte_Daten[38]	Octet 38	Wert MD 90 Byte 2
Easy_800_64_Byte_Daten[39]	Octet 39	Wert MD 90 Byte 3
Easy_800_64_Byte_Daten[40]	Octet 40	Wert MD 90 Byte 4
Easy_800_64_Byte_Daten[41]	Octet 41	Wert MD 91 Byte 1
Easy_800_64_Byte_Daten[42]	Octet 42	Wert MD 91 Byte 2
Easy_800_64_Byte_Daten[43]	Octet 43	Wert MD 91 Byte 3
Easy_800_64_Byte_Daten[44]	Octet 44	Wert MD 91 Byte 4
Easy_800_64_Byte_Daten[45]	Octet 45	Wert MD 92 Byte 1
Easy_800_64_Byte_Daten[46]	Octet 46	Wert MD 92 Byte 2
Easy_800_64_Byte_Daten[47]	Octet 47	Wert MD 92 Byte 3
Easy_800_64_Byte_Daten[48]	Octet 48	Wert MD 92 Byte 4
Easy_800_64_Byte_Daten[49]	Octet 49	Wert MD 93 Byte 1
Easy_800_64_Byte_Daten[50]	Octet 50	Wert MD 93 Byte 2
Easy_800_64_Byte_Daten[51]	Octet 51	Wert MD 93 Byte 3

Variablen-Zugriff (Beispiel)	Daten-position	Bedeutung
Easy_800_64_Byte_Daten[52]	Octet 52	Wert MD 93 Byte 4
Easy_800_64_Byte_Daten[53]	Octet 53	Wert MD 94 Byte 1
Easy_800_64_Byte_Daten[54]	Octet 54	Wert MD 94 Byte 2
Easy_800_64_Byte_Daten[55]	Octet 55	Wert MD 94 Byte 3
Easy_800_64_Byte_Daten[56]	Octet 56	Wert MD 94 Byte 4
Easy_800_64_Byte_Daten[57]	Octet 57	Wert MD 95 Byte 1
Easy_800_64_Byte_Daten[58]	Octet 58	Wert MD 95 Byte 2
Easy_800_64_Byte_Daten[59]	Octet 59	Wert MD 95 Byte 3
Easy_800_64_Byte_Daten[60]	Octet 60	Wert MD 95 Byte 4
Easy_800_64_Byte_Daten[61]	Octet 61	Wert MD 96 Byte 1
Easy_800_64_Byte_Daten[62]	Octet 62	Wert MD 96 Byte 2
Easy_800_64_Byte_Daten[63]	Octet 63	Wert MD 96 Byte 3
Easy_800_64_Byte_Daten[64]	Octet 64	Wert MD 96 Byte 4

Objekt „Easy Link Eingangsdaten“

Die Beschreibung dieses Objektes ist identisch zum easy600 (DPV1), → Seite 277.

Objekt „Easy Link Ausgangsdaten“

Die Beschreibung dieses Objektes ist identisch zum easy600 (DPV1), → Seite 279.

**Funktionsbaustein-Daten
lesen/schreiben****Verfahrensweise**

Das Lesen und Schreiben der Daten von Funktionsbausteinen ist mit Hilfe der folgenden generischen Objekte einfach möglich:

- „Easy 700/800 Auswahl Daten Funktionsbaustein“
- „Easy 700/800 Lesen Daten Funktionsbaustein“
- „Easy 700/800 Schreiben Daten Funktionsbaustein“ .

Um Daten eines Funktionsbausteins zu lesen, sind zwei Objektzugriffe notwendig:

- Auswahl der gewünschten Daten eines Funktionsbausteins für den nachfolgenden Auslesevorgang:
→ Objekt „Easy 700/800 Auswahl Daten Funktionsbaustein“, Seite 337.
- Lesen der ausgewählten Daten des Funktionsbausteins:
→ Objekt „Easy 700/800 Lesen Daten Funktionsbaustein“, Seite 339.

Um Daten eines Funktionsbausteins zu schreiben, ist nur ein Objektzugriff notwendig. Dabei werden gleichzeitig die gewünschten Daten eines Funktionsbausteines ausgewählt und neue Werte für diese übergeben.

→ Objekt „Easy 700/800 Schreiben Daten Funktionsbaustein“, Seite 341.

Analogwertvergleichler A1 bis A32

Operanden-Auswahl	Wert (hex)
Type	11
Instanz ¹⁾	01 - 20
Index	00 - 07, → Tabelle 267

1) Der Wert 01_{hex} wählt den Analogwertvergleichler A1 aus, der Wert 20_{hex} den Analogwertvergleichler A32.

Tabelle 267: Operandenübersicht

Index	Operand	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	–	–	→ Tab. 268	–	R
01	Mode	→ Tab. 269	–	–	–	R
02	Vergleichswert I1	DWORD oder UDINT ¹⁾				R/W ²⁾
03	Verstärkungsfaktor F1 (I1 = F1*Wert)	DWORD oder UDINT ¹⁾				R/W ²⁾
04	Vergleichswert I2	DWORD oder UDINT ¹⁾				R/W ²⁾
05	Verstärkungsfaktor F2 (I2 = F2*Wert)	DWORD oder UDINT ¹⁾				R/W ²⁾
06	Offset OS für I1	DWORD oder UDINT ¹⁾				R/W ²⁾
07	Schalthyre HY für den Wert I2	DWORD oder UDINT ¹⁾				R/W ²⁾

1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte

2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 268: Index 0: Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		–	–	–	–	–	–	CY ¹⁾	Q1 ²⁾

- 1) Zustand „1“, wenn der Wertebereich überschritten wird
- 2) Zustand „1“, wenn die Bedingung erfüllt ist (z. B. I1 < I2 bei der Betriebsart LT)

Tabelle 269: Index 1 – Mode

Data 1 (hex)		
00	LT	kleiner (I1 < I2)
01	EQ	gleich (I1 = IGT)
02	GT	größer (I1 > I2)



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Arithmetikbausteine AR1 bis AR32

Operanden-Auswahl	Wert (hex)
Type	12
Instanz ¹⁾	01 - 20
Index	00 - 04, → Tabelle 270

1) Der Wert 01_{hex} für die Instanz wählt den Arithmetikbaustein AR1 aus, der Wert 20_{hex} den Arithmetikbaustein AR32.

Tabelle 270: Operandenübersicht

Index	Operand	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	–	–	→ Tab. 271	–	R
01	Mode	→ Tab. 272	–	–	–	R
02	erster Operand I1	DWORD oder UDINT ¹⁾				R/W ²⁾
03	zweiter Operand I2	DWORD oder UDINT ¹⁾				R/W ²⁾
04	Ergebnis QV	DWORD oder UDINT ¹⁾				R

1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte

2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 271: Index 0: Bit-IO

	Bit	7	6	5	4	3	2	1
FB-Ausgang Data 3		–	–	–	–	–	ZE ¹⁾	CY ¹⁾

- 1) Zustand „1“, wenn der Wert des Bausteinausgangs QV (also das Rechenergebnis) gleich Null ist
- 2) Zustand „1“, wenn der Wertebereich überschritten wird

Tabelle 272: Index 1 – Mode

Data 1 (hex)		
00	ADD	Addieren ($I1 + I2 = QV$)
01	SUB	Subtrahieren ($I1 - I2 = QV$)
02	MUL	Multiplizieren ($I1 \times I2 = QV$)
03	DIV	Dividieren ($I1 : I2 = QV$)



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Block Compare Bausteine BC1 bis BC32

Operanden-Auswahl	Wert (hex)
Type	25
Instanz ¹⁾	01 - 20
Index	00 - 04, → Tabelle 273

1) Der Wert 01_{hex} für die Instanz wählt den Block Compare Baustein BC1 aus, der Wert 20_{hex} den Block Compare Baustein BC32.

Tabelle 273: Operandenübersicht

Index	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	→ Tab. 274	–	→ Tab. 274	–	R
01	Mode	→ Tab. 275	–	–	–	R
02	Quellbereich I1	DWORD oder UDINT ¹⁾				R/W ²⁾
03	Zielbereich I2	DWORD oder UDINT ¹⁾				R/W ²⁾
04	Anzahl der zu vergleichenden Elemente NO (max. 192 Byte)	DWORD oder UDINT ¹⁾				R/W ²⁾

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte
- 2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 274: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	EN ¹⁾
FB-Ausgang Data 3		–	–	–	–	EQ ²⁾	E3 ³⁾	E2 ⁴⁾	E1 ⁵⁾

- 1) Aktiviert den Baustein bei Zustand „1“.
- 2) Zustand „1“, wenn die Datenbereiche gleich sind; Zustand „0“, wenn sie ungleich sind

Fehlerausgänge

- 3) Zustand „1“, wenn die Anzahl der Elemente den Quell- oder Zielbereich überschreitet.
- 4) Zustand „1“, wenn sich Quell- und Zielbereich überlappen.
- 5) Zustand „1“, wenn Quell- oder Zielbereich außerhalb des verfügbaren Merkerbereichs liegen (Offsetfehler).

Tabelle 275: Index 1 – Mode

Mode	Data 1 (hex)	Betriebsart
	02	Vergleichen (easy intern Zustandsanzeige Modus Block Compare)



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Block Transfer Bausteine BT1 bis BT32

Operanden-Auswahl	Wert (hex)
Type	26
Instanz ¹⁾	01 - 20
Index	00 - 04, → Tabelle 276

1) Der Wert 01_{hex} für die Instanz wählt den Block Transfer Baustein BT1 aus, der Wert 20_{hex} den Block Transfer Baustein BC32.

Tabelle 276: Operandenübersicht

Index	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	→ Tab. 277	–	→ Tab. 277	–	R
01	Mode	→ Tab. 278	–	–	–	R
02	Quellbereich I1	DWORD oder UDINT ¹⁾				R/W ²⁾
03	Zielbereich I2	DWORD oder UDINT ¹⁾				R/W ²⁾
04	Anzahl der zu transferierenden Elemente NO (max. 192 Byte)	DWORD oder UDINT ¹⁾				R/W ²⁾

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte
- 2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 277: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	T ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	E3 ²⁾	E2 ³⁾	E1 ⁴⁾

1) Transfer von der an I1 angegebenen Quelladresse zu der an I2 angegebenen Zieladresse bei positiver Flanke.

Fehlerausgänge

- 2) Zustand „1“, wenn die Anzahl der Elemente den Quell- oder Zielbereich überschreitet.
- 3) Zustand „1“, wenn sich Quell- und Zielbereich überlappen.
- 4) Zustand „1“, wenn Quell- oder Zielbereich außerhalb des verfügbaren Merkerbereichs liegen (Offsetfehler).

Tabelle 278: Index 1 – Mode

Data 1 (hex)	Betriebsart
00	INI: Initialisiert den Zielbereich mit einem Bytewert, der unter der Quelladresse hinterlegt ist.
01	CPY: Kopiert einen Datenblock von einem Quell- zu einem Zielbereich. Über „NO“ geben Sie die Datenblockgröße vor.



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Boolesche Verknüpfung Bausteine BV1 bis BV32

Operanden-Auswahl	Wert (hex)
Type	13
Instanz ¹⁾	01 - 20
Index	00 - 04, → Tabelle 279

1) Der Wert 01_{hex} für die Instanz wählt den Boolesche Verknüpfung Baustein BV1 aus, der Wert 20_{hex} den Boolesche Verknüpfung Baustein BV32.

Tabelle 279: Operandenübersicht

Index	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	–	–	→ Tab. 280	–	R
01	Mode	→ Tab. 281	–	–	–	R
02	erster Operand I1	DWORD oder UDINT ¹⁾				R/W ²⁾
03	zweiter Operand I2	DWORD oder UDINT ¹⁾				R/W ²⁾
04	Ergebnis QV	DWORD oder UDINT ¹⁾				R/W ²⁾

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte
- 2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 280: Index 0: Bit-IO

	Bit	7	6	5	4	3	2	1
FB-Ausgang Data 3		–	–	–	–	–	–	ZE ¹⁾

1) Zustand „1“, wenn der Wert des Bausteinausgangs QV (also das Ergebnis der Verknüpfung) gleich Null ist

Tabelle 281: Index 1 – Mode

Data 1 (hex)		
00	AND	Und-Verknüpfung
01	OR	Oder-Verknüpfung
02	XOR	Exklusiv-Oder-Verknüpfung
03	NET	Invertiert die einzelnen Bit des Wertes von I1. Der invertierte Wert wird vorzeichenbehaftet dezimal dargestellt.



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Zähler C1 bis C32

Operanden-Auswahl	Wert (hex)
Type	14
Instanz ¹⁾	01 - 20
Index	00 - 05, → Tabelle 282

1) Der Wert 01_{hex} für die Instanz wählt den Zähler Baustein C1 aus, der Wert 20_{hex} den Zähler Baustein C32.

Tabelle 282: Operandenbedeutung

Index	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	→ Tab. 283	–	→ Tab. 283	–	R
02	oberer Sollwert SH	DWORD oder UDINT ¹⁾				R/W ²⁾
03	unterer Sollwert SL	DWORD oder UDINT ¹⁾				R/W ²⁾
04	Vorgabe Istwert SV	DWORD oder UDINT ¹⁾				R/W ²⁾
05	Aktueller Istwert QV	DWORD oder UDINT ¹⁾				R

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte
- 2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 283: Index 0: Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	SE ¹⁾	D ²⁾	C ³⁾	RE ⁴⁾
FB-Ausgang Data 3		–	–	–	–	ZE ⁵⁾	CY ⁶⁾	FB ⁷⁾	OF ⁸⁾

- 1) bei positiver Flanke Vorgabe-Istwert übernehmen
- 2) Zählrichtungsangabe: Zustand „0“ = vorwärts zählen, Zustand „1“ = rückwärts zählen
- 3) Zählspule, zählt bei jeder positiven Flanke
- 4) Rücksetzen des Istwertes auf Null
- 5) Zero: Zustand „1“, wenn der Wert des Bausteinausganges QV (also der Zählerstand) gleich Null ist
- 6) Carry: Zustand „1“, wenn der Wertebereich überschritten wird.
- 7) Fall below: Zustand „1“, wenn Istwert \leq unterer Sollwert
- 8) Overflow: Zustand „1“, wenn Istwert \geq oberer Sollwert



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Frequenzzähler CF1 bis CF4

Operanden-Auswahl	Wert (hex)
Type	15
Instanz ¹⁾	01 - 04
Index	00 - 04, → Tabelle 284

1) Der Wert 01_{hex} für die Instanz wählt den Frequenzzähler CF1 aus, der Wert 04_{hex} den Frequenzzähler CF4.

Tabelle 284: Operandenübersicht

Index	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	→ Tab. 285	–	→ Tab. 285	–	R
02	oberer Sollwert SH	DWORD oder UDINT ¹⁾				R/W ²⁾
03	unterer Sollwert SL	DWORD oder UDINT ¹⁾				R/W ²⁾
04	Aktueller Istwert QV	DWORD oder UDINT ¹⁾				R

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte
- 2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 285: Index 0: Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	EN ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	ZE ²⁾	FB ³⁾	OF ⁴⁾

- 1) Freigabe des Zählers
- 2) Zero: Zustand „1“, wenn der Wert des Bausteinausganges QV (also der Zählerstand) gleich Null ist
- 3) Fall below: Zustand „1“, wenn Istwert \leq unterer Sollwert
- 4) Overflow: Zustand „1“, wenn Istwert \geq oberer Sollwert.



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; vormals AWB2528-1423D) oder in der easySoft-Hilfe.

Schneller Zähler CH1 bis CH4

Operanden-Auswahl	Wert (hex)
Type	16
Instanz ¹⁾	01 - 04
Index	00 - 05, → Tabelle 286

1) Der Wert 01_{hex} für die Instanz wählt den Schnellen Zähler CH1 aus, der Wert 04_{hex} den Schnellen Zähler CH4.

Tabelle 286: Operandenübersicht

Index	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	→ Tab. 287	–	→ Tab. 287	–	R
02	oberer Sollwert SH	DWORD oder UDINT ¹⁾				R/W ²⁾
03	unterer Sollwert SL	DWORD oder UDINT ¹⁾				R/W ²⁾
04	Vorgabe Istwert SV	DWORD oder UDINT ¹⁾				R/W ²⁾
05	Aktueller Istwert QV	DWORD oder UDINT ¹⁾				R

1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte

2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 287: Index 0: Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	EN ¹⁾	SE ²⁾	D ³⁾	RE ⁴⁾
FB-Ausgang Data 3		–	–	–	–	ZE ⁵⁾	CY ⁶⁾	FB ⁷⁾	OF ⁸⁾

- 1) Freigabe des Zählers
- 2) bei positiver Flanke Vorgabe-Istwert übernehmen
- 3) Zählrichtungsangabe: Zustand „0“ = vorwärts zählen, Zustand „1“ = rückwärts zählen
- 4) Rücksetzen des Istwertes auf Null
- 5) Zero: Zustand „1“, wenn der Wert des Bausteinausganges QV (also der Zählerstand) gleich Null ist
- 6) Carry: Zustand „1“, wenn der Wertebereich überschritten wird.
- 7) Fall below: Zustand „1“, wenn Istwert \leq unterer Sollwert
- 8) Overflow: Zustand „1“, wenn Istwert \geq oberer Sollwert



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Inkrementalzähler C11 bis C14

Operanden-Auswahl	Wert (hex)
Type	17
Instanz ¹⁾	01 - 04
Index	00 - 05, → Tabelle 288

1) Der Wert 01_{hex} für die Instanz wählt den Inkrementalzähler C11 aus, der Wert 04_{hex} den Inkrementalzähler C14.

Tabelle 288: Operandenübersicht

Index (hex)	Daten	Data 1, Data 3	Data 2, Data 4	Read/Write
00	Bit-IO	→ Tabelle 289	–	R
02	oberer Sollwert SH	DWORD oder UDINT ¹⁾		R/W ²⁾
03	unterer Sollwert SL	DWORD oder UDINT ¹⁾		R/W ²⁾
04	Vorgabe Istwert SV	DWORD oder UDINT ¹⁾		R/W ²⁾
05	Aktueller Istwert QV	DWORD oder UDINT ¹⁾		R

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte
- 2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 289: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	EN ¹⁾	SE ²⁾	RE ³⁾
FB-Ausgang Data 3		–	–	–	–	ZE ⁴⁾	CY ⁵⁾	FB ⁶⁾	OF ⁷⁾

- 1) Freigabe des Zählers
- 2) Bei positiver Flanke Vorgabe-Istwert übernehmen
- 3) Rücksetzen des Istwertes auf Null
- 4) Zero: Zustand „1“, wenn der Wert des Bausteinausganges QV (also der Zählerstand) gleich Null ist
- 5) Carry: Zustand „1“, wenn der Wertebereich überschritten wird.
- 6) Fall below: Zustand „1“, wenn Istwert ≤ unterer Sollwert
- 7) Overflow: Zustand „1“, wenn Istwert ≥ unterer Sollwert



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; vormals AWB2528-1423D) oder in der easySoft-Hilfe.

Vergleicher CP1 bis CP32

Operanden-Auswahl	Wert (hex)
Type	18
Instanz ¹⁾	01 - 20
Index	00 - 03, → Tabelle 290

1) Der Wert 01_{hex} für die Instanz wählt den Vergleicher CP1 aus, der Wert 20_{hex} den Vergleicher CP32.

Tabelle 290: Operandenübersicht

Index	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	–	–	→ Tab. 291	–	R
02	Vergleichswert I1	DWORD oder UDINT ¹⁾				R/W ²⁾
03	Vergleichswert I2	DWORD oder UDINT ¹⁾				R/W ²⁾

1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte

2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 291: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1
FB-Ausgang Data 3		–	–	–	–	GT ¹⁾	EQ ²⁾	LT ³⁾

- 1) greater than: Zustand „1“, wenn der Wert an I1 größer als der Wert an I2 ist (I1 > I2)
- 2) equal: Zustand „1“, wenn der Wert an I1 gleich dem Wert an I2 ist (I1 = I2)
- 3) less than: Zustand „1“, wenn der Wert an I1 kleiner als der Wert an I2 ist (I1 < I2).



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Textausgabe Bausteine D1 bis D32

Operanden-Auswahl	Wert (hex)
Type	19
Instanz ¹⁾	01 - 20
Index	00 - 33

1) Der Wert 01_{hex} für die Instanz wählt den Textausgabe Baustein D1 aus, der Wert 20_{hex} den Baustein D32.

Tabelle 292: Operandenübersicht

Index (hex)	Operand	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	→ Tab. 293	–	→ Tab. 293	–	R
02	Text Zeile 1, Spalte 1 bis 4	DWORD oder UDINT ¹⁾				R
03	Text Zeile 1, Spalte 5 bis 8	DWORD oder UDINT ¹⁾				R
04	Text Zeile 1, Spalte 9 bis 12	DWORD oder UDINT ¹⁾				R
05	Text Zeile 1, Spalte 13 bis 16	DWORD oder UDINT ¹⁾				R
06	Text Zeile 2, Spalte 1 bis 4	DWORD oder UDINT ¹⁾				R
07	Text Zeile 2, Spalte 5 bis 8	DWORD oder UDINT ¹⁾				R
08	Text Zeile 2, Spalte 9 bis 12	DWORD oder UDINT ¹⁾				R
09	Text Zeile 2, Spalte 13 bis 16	DWORD oder UDINT ¹⁾				R
10	Text Zeile 3, Spalte 1 bis 4	DWORD oder UDINT ¹⁾				R
11	Text Zeile 3, Spalte 5 bis 8	DWORD oder UDINT ¹⁾				R
12	Text Zeile 3, Spalte 9 bis 12	DWORD oder UDINT ¹⁾				R
13	Text Zeile 3, Spalte 13 bis 16	DWORD oder UDINT ¹⁾				R
14	Text Zeile 4, Spalte 1 bis 4	DWORD oder UDINT ¹⁾				R
15	Text Zeile 4, Spalte 5 bis 8	DWORD oder UDINT ¹⁾				R
16	Text Zeile 4, Spalte 9 bis 12	DWORD oder UDINT ¹⁾				R
17	Text Zeile 4, Spalte 13 bis 16	DWORD oder UDINT ¹⁾				R
18	Variable 1	DWORD oder UDINT ¹⁾				R/W ²⁾

Index (hex)	Operand	Data 1	Data 2	Data 3	Data 4	Read/Write
19	Variable 2	DWORD oder UDINT ¹⁾				R/W ²⁾
20	Variable 3	DWORD oder UDINT ¹⁾				R/W ²⁾
21	Variable 4	DWORD oder UDINT ¹⁾				R/W ²⁾
22	Skalierung Minimalwert 1	DWORD oder UDINT ¹⁾				R
23	Skalierung Minimalwert 2	DWORD oder UDINT ¹⁾				R
24	Skalierung Minimalwert 3	DWORD oder UDINT ¹⁾				R
25	Skalierung Minimalwert 4	DWORD oder UDINT ¹⁾				R
26	Skalierung Maximalwert 1	DWORD oder UDINT ¹⁾				R
27	Skalierung Maximalwert 2	DWORD oder UDINT ¹⁾				R
28	Skalierung Maximalwert 3	DWORD oder UDINT ¹⁾				R
29	Skalierung Maximalwert 4	DWORD oder UDINT ¹⁾				R
30	Steuerinformationen Zeile 1	DWORD oder UDINT ¹⁾				R
31	Steuerinformationen Zeile 2	DWORD oder UDINT ¹⁾				R
32	Steuerinformationen Zeile 3	DWORD oder UDINT ¹⁾				R
33	Steuerinformationen Zeile 4	DWORD oder UDINT ¹⁾				R

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte
- 2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 293: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	EN ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q1 ²⁾

- 1) Freigabe des Textbausteins
- 2) Zustand „1“, Textbaustein ist aktiv



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Datenbausteine DB1 bis DB32

Operanden-Auswahl	Wert (hex)
Type	1A
Instanz ¹⁾	01 - 20
Index	00 - 03, → Tabelle 294

1) Der Wert 01_{hex} für die Instanz wählt den Datenbaustein DB1 aus, der Wert 20_{hex} den Datenbaustein DB32.

Tabelle 294: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	→ Tab. 295	–	→ Tab. 295	–	R
02	Eingangswert I1	DWORD oder UDINT ¹⁾				R/W ²⁾
03	Ausgangswert QV	DWORD oder UDINT ¹⁾				R

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte
- 2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 295: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	T ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q ¹⁾

- 1) Übernahme des an I1 liegenden Wertes bei positiver Flanke.
- 2) Zustand „1“, wenn das Triggersignal den Zustand „1“ besitzt.



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

PID-Regler DC1 bis DC32

Operanden-Auswahl	Wert (hex)
Type	27
Instanz ¹⁾	01 - 20
Index	00 - 09, → Tabelle 296

1) Der Wert 01_{hex} für die Instanz wählt den PID Regler DC1 aus, der Wert 20_{hex} den PID Regler DC32.

Tabelle 296: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	→ Tab. 297	–	→ Tab. 297	–	R
01	Mode	→ Tab. 298	–	–	–	R
02	Sollwert I1: -32768 bis +32767	DWORD oder UDINT ¹⁾				R/W ²⁾
03	Istwert I2: -32768 bis +32767	DWORD oder UDINT ¹⁾				R/W ²⁾
04	Proportionalverstärkung KP [%], Wertebereich 0 bis 65535	DWORD oder UDINT ¹⁾				R/W ²⁾
05	Nachstellzeit TN [0,1s], Wertebereich 0 bis 65535	DWORD oder UDINT ¹⁾				R/W ²⁾
06	Vorhaltezeit TV [0,1s], Wertebereich 0 bis 65535	DWORD oder UDINT ¹⁾				R/W ²⁾
07	Abtastzeit TC Wertebereich 0,1s bis 65535,5s Wert 0: Zeit entspricht Programmzyklus	DWORD oder UDINT ¹⁾				R/W ²⁾
08	Handstellgröße MV Wertebereich -4096 bis +4095	DWORD oder UDINT ¹⁾				R/W ²⁾
09	Stellgröße QV Mode UNI, Wertebereich 0 bis +4095 (12 Bit) Mode BIP, Wertebereich -4096 bis +4095 (13 Bit)	DWORD oder UDINT ¹⁾				R

1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte

2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 297: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	SE ¹⁾	ED ²⁾	EI ³⁾	EP ⁴⁾	EN ⁵⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	LI ⁶⁾

- 1) Übernahme der Handstellgröße bei Zustand „1“
- 2) Aktivieren des D-Teils bei Zustand „1“
- 3) Aktivieren des I-Teils bei Zustand „1“
- 4) Aktivieren des P-Teils bei Zustand „1“
- 5) Aktiviert den Baustein bei Zustand „1“
- 6) Zustand „1“, wenn der Wertebereich der Stellgröße überschritten wurde

Tabelle 298: Index 1 – Mode

Data 1	Betriebsart
UNP unipolar	Die Stellgröße wird als unipolarer 12Bit-Wert ausgegeben. Entsprechender Wertebereich für QV 0 bis 4095.
BIP bipolar	Die Stellgröße wird als bipolarer 13Bit-Wert ausgegeben. Entsprechender Wertebereich für QV –4096 bis 4095



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Diagnose DG01...DG16

Operanden-Auswahl	Wert (hex)
Type	39
Instanz ¹⁾	01 - 10
Index	00 - 03
Data 1 - 4	abhängig vom Index, → Tabelle 299

- 1) Der Wert 01hex wählt den Baustein DG01 aus, der Wert 10hex den Baustein DG32.

Tabelle 299: Operandenübersicht

Index (hex)	Daten	Data 1, Data 3, Data 4	Data 2	Read/Write
0	Bit-IO	→ Tabelle 300	–	R
2	Diagnose-Register QV	DWORD oder UDINT ¹⁾		R
3	Ausgangs-Zustände ON	DWORD oder UDINT ¹⁾		R

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte.

Tabelle 300: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	EN ¹⁾
FB-Ausgang Data 3		Q8 ²⁾	Q7 ²⁾	Q6 ²⁾	Q5 ²⁾	Q4 ²⁾	Q3 ²⁾	Q2 ²⁾	Q1 ²⁾
FB-Ausgang Data 4		–	–	–	–	–	–	–	QC ³⁾

- 1) Im Zustand „1“ wird der Funktionsbaustein aktiviert.
- 2) Der Zustand „1“ wird eingenommen, wenn der ausgewählte Sicherheitsbaustein den gewählten Zustand besitzt.
- 3) Der Zustand „1“ wird eingenommen, wenn einer der Ausgänge Q1 bis Q8 den Zustand „1“ eingenommen hat.



Weitere Informationen zu diesem Baustein finden Sie im easy800-Handbuch (MN04902001Z-DE; vormals AWB2528-1423D) oder in der easySoft-Hilfe.

Signalglättungsfilter FT1 bis FT32

Operanden-Auswahl	Wert (hex)
Type	28
Instanz ¹⁾	01 - 20
Index	00 - 05, → Tabelle 301

1) Der Wert 01_{hex} für die Instanz wählt den Signalglättungsfilter FT1 aus, der Wert 20_{hex} den Signalglättungsfilter FT32.

Tabelle 301: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/write
00	Bit-IO	–	–	→ Tab. 302	–	R
02	Eingangswert I1: -32768 bis +32767	DWORD oder UDINT ¹⁾				R/W ²⁾
03	Ausgleichszeit TG [0,1s], Wertebereich 0 bis 65535	DWORD oder UDINT ¹⁾				R/W ²⁾
04	Proportionalverstärkung KP [%], Wertebereich 0 bis 65535	DWORD oder UDINT ¹⁾				R/W ²⁾
05	Verzögerter Ausgabewert QV, Wertebereich -32768 bis +32767	DWORD oder UDINT ¹⁾				R

1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte

2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 302: Index 0 – Bit-IO

FB-Ausgang Data 3	Bit	7	6	5	4	3	2	1	0
			–	–	–	–	–	–	–

1) Aktiviert den Baustein bei Zustand „1“



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; vormals AWB2528-1423D) oder in der easySoft-Hilfe.

Bausteine Empfang Netzdaten GT1 bis GT32

Operanden-Auswahl	Wert (hex)
Type	1B
Instanz ¹⁾	01 - 20
Index	00 - 02, → Tabelle 303

1) Der Wert 01_{hex} für die Instanz wählt den Baustein Empfang Netzdaten GT1 aus, der Wert 20_{hex} den Baustein Empfang Netzdaten GT32.

Tabelle 303: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	–	–	→ Tab. 304	–	R
01	Mode/Parameter	→ Tab. 305	–	→ Tab. 305	–	R
02	Istwert aus dem Netzwerk QV	DWORD oder UDINT ¹⁾				R

1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte

Tabelle 304: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q ¹⁾

1) Zustand „1“, wenn ein neuer Wert anliegt, der vom Netzwerk NET übertragen wird.

Tabelle 305: Index 1 – Mode/Parameter (Bezeichnung des FB PUT, dessen Daten zu empfangen sind)

Mode (Data 1)	NET-ID¹⁾	
	0	NET-ID 1

	7	NET-ID 8
Parameter (Data 3)	Instanz²⁾	
	0	PT01

	31	PT32

1) Nummer des Teilnehmers, der den Wert sendet.
Mögliche Teilnehmernummer: 01 bis 08

2) Sendebaustein (z. B. PT 20) des NET-Teilnehmers, der sendet. Mögliche Bausteinnummer: 01 - 32



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Wochenschaltuhren HW1 bis HW32

Operanden-Auswahl	Wert (hex)
Type	1C
Instanz ¹⁾	01 - 20
Index	00 - 05, → Tabelle 306

1) Der Wert 01_{hex} für die Instanz wählt die Wochenschaltuhr HW1 aus, der Wert 20_{hex} die Wochenschaltuhr HW32.

Tabelle 306: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	–	–	→ Tabelle 307	–	R
02	Parameter Kanal A	→ Tabelle 308				R
03	Parameter Kanal B	→ Tabelle 308				R
04	Parameter Kanal C	→ Tabelle 308				R
05	Parameter Kanal D	→ Tabelle 308				R

Tabelle 307: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q ¹⁾

1) Zustand „1“, wenn die Einschaltbedingung erfüllt ist.

Die Daten sind in der nachfolgenden Tabelle im Motorola-format dargestellt, obwohl sie tatsächlich im Intelformat übergeben werden.

Tabelle 308: Index 2 - 5, Parameterkanäle A - D

Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
	Date 2								Date 1							
ON	d4	d3	d2	d1	d0	h4	h3	h2	h1	h0	m5	m4	m3	m2	m1	m0
	Wochentag					Stunde					Minute					

Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
	Date 4								Date 3							
OFF	d4	d3	d2	d1	d0	h4	h3	h2	h1	h0	m5	m4	m3	m2	m1	m0
	Wochentag					Stunde					Minute					

m5 bis m0: Minute (0 bis 59)

h4 bis h0: Stunde (0 bis 23)

d5 bis d0: Wochentag (0 = Sonntag bis 6 = Samstag)



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Jahresschaltuhren HY1 bis HY32

Operanden-Auswahl	Wert (hex)
Type	1D
Instanz ¹⁾	01 - 20
Index	00 - 05, → Tabelle 309

1) Der Wert 01_{hex} für die Instanz wählt die Jahresschaltuhr HY1 aus, der Wert 20_{hex} die Jahresschaltuhr HY32.

Tabelle 309: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	–	–	→ Tab. 310	–	R
02	Parameter Kanal A	→ Tab. 311				R
03	Parameter Kanal B	→ Tab. 311				R
04	Parameter Kanal C	→ Tab. 311				R
05	Parameter Kanal D	→ Tab. 311				R

Tabelle 310: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q ¹⁾

1) Zustand „1“, wenn die Einschaltbedingung erfüllt ist.

Die Daten sind in der nachfolgenden Tabelle im Motorola-format dargestellt, obwohl sie tatsächlich im Intelformat übergeben werden.

Tabelle 311: Index 2 - 5, Parameterkanäle A - D

Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
	Date 2								Date 1							
ON	y6	y5	y4	y3	y2	y1	y0	m3	m2	m1	m0	d4	d3	d2	d1	d0
	Jahr							Monat				Tag				

Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
	Date 4								Date 3							
OFF	y6	y5	y4	y3	y2	y1	y0	m3	m2	m1	m0	d4	d3	d2	d1	d0
	Jahr							Monat				Tag				

d4...d0: Tag (1...31), m3...m0: Monat (1...12), y6...y0: Jahr (0: 2000...99: 2099)



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Bedingter Sprung JC01...JC32

Operanden-Auswahl	Wert (hex)
Type	2F
Instanz ¹⁾	01 - 20
Index	00
Data 1 - 4	abhängig vom Index, → Tabelle 312

1) Der Wert 01hex wählt den Baustein JC01 aus, der Wert 20hex den Baustein JC32.

Tabelle 312: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
0	Bit-IO	→ Tabelle 313	–	→ Tabelle 313	–	R

Tabelle 313: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	EN ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	E1 ²⁾

- 1) Im Zustand „1“ wird im Programm auf die zugehörige Sprungmarke verzweigt.
- 2) Zustand „1“ wird eingenommen, wenn die zugehörige Sprungmarke nicht gefunden wurde.



Weitere Informationen zu diesem Baustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easySoft-Hilfe.

Wertskalierungen LS1 bis LS32

Operanden-Auswahl	Wert (hex)
Type	29
Instanz ¹⁾	01 - 20
Index	00 - 07, → Tabelle 314

1) Der Wert 01_{hex} für die Instanz wählt die Wertskalierung LS1 aus, der Wert 20_{hex} die Wertskalierung LS32.

Tabelle 314: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	–	–	→ Tab. 315	–	R
02	Eingangswert I1: Wertebereich 32 Bit	DWORD oder UDINT ¹⁾				R/W ²⁾
03	Stützpunkt X1: (X-Koordinate) Wertebereich 32 Bit	DWORD oder UDINT ¹⁾				R/W ²⁾
04	Stützpunkt Y1: (Y-Koordinate) Wertebereich 32 Bit	DWORD oder UDINT ¹⁾				R/W ²⁾
05	Stützpunkt X2: (X-Koordinate) Wertebereich 32 Bit	DWORD oder UDINT ¹⁾				R/W ²⁾
06	Stützpunkt Y2: (Y-Koordinate) Wertebereich 32 Bit	DWORD oder UDINT ¹⁾				R/W ²⁾
07	Skalierter Eingangswert QV	DWORD oder UDINT ¹⁾				R

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte
- 2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 315: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		–	–	–	–	–	–	–	EN ¹⁾

1) Aktiviert den Baustein bei Zustand „1“



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Masterreset MR1 bis MR32

Operanden-Auswahl	Wert (hex)
Type	0F
Instanz ¹⁾	01 - 20
Index	00 - 01, → Tabelle 316

1) Der Wert 01_{hex} für die Instanz wählt den Masterreset MR1 aus, der Wert 20_{hex} den Masterreset MR32.

Tabelle 316: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
0	Bit-IO	→ Tab. 317	–	→ Tab. 317	–	R
1	Mode/Parameter	→ Tab. 318	–	–	–	R

Tabelle 317: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	T ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q ¹²⁾

- 1) Triggerspule. Wird die Spule getriggert (erhält eine positive Flanke), wird der entsprechende Reset durchgeführt.
- 2) Zustand „1“, wenn die Triggerspule MR...T den Zustand „1“ besitzt.

Tabelle 318: Index 1 – Mode

Data 1 (hex)		
00	Q	die Ausgänge Q..., *Q..., S..., *S..., *SN..., QA01 werden auf den Zustand „0“ zurückgesetzt. * entsprechend der NET-ID
01	M	der Merkerbereich MD01 bis MD48 wird auf Zustand „0“ zurückgesetzt
02	ALL	wirkt auf Q und M.



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Datenmultiplexer MX01...MX32

Operanden-Auswahl	Wert (hex)
Type	31
Instanz ¹⁾	01 - 20
Index	00 - 0B
Data 1 - 4	abhängig vom Index, → Tabelle 319

1) Der Wert 01hex wählt den Baustein MX01 aus, der Wert 20hex den Baustein MX32.

Tabelle 319: Operandenübersicht

Index (hex)	Daten	Data 1 Data 3	Data 2 Data 4	Read/Write
0	Bit-IO	→ Tabelle 320	–	R
2	Kanalauswahl: 0 bis 7	DWORD oder UDINT ¹⁾		R/W ²⁾
3	Eingangswert Kanal 1	DWORD oder UDINT ¹⁾		R/W ²⁾
4	Eingangswert Kanal 2	DWORD oder UDINT ¹⁾		R/W ²⁾
5	Eingangswert Kanal 3	DWORD oder UDINT ¹⁾		R/W ²⁾
6	Eingangswert Kanal 4	DWORD oder UDINT ¹⁾		R/W ²⁾
7	Eingangswert Kanal 5	DWORD oder UDINT ¹⁾		R/W ²⁾
8	Eingangswert Kanal 6	DWORD oder UDINT ¹⁾		R/W ²⁾
9	Eingangswert Kanal 7	DWORD oder UDINT ¹⁾		R/W ²⁾
A	Eingangswert Kanal 8	DWORD oder UDINT ¹⁾		R/W ²⁾
B	Ausgangswert QV	DWORD oder UDINT ¹⁾		R

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte.
- 2) Der Wert kann nur beschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 320: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	EN ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	E1 ²⁾

- 1) Im Zustand „1“ wird der gewählte Eingangswert in den Ausgangswert eingetragen.
- 2) Zustand „1“ wird eingenommen, wenn die Kanalauswahl ungültig ist.



Weitere Informationen zu diesem Baustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easySoft-Hilfe.

Zahlenwandler NC1 bis NC32

Operanden-Auswahl	Wert (hex)
Type	2A
Instanz ¹⁾	01 - 20
Index	00 - 03, → Tabelle 321

1) Der Wert 01_{hex} für die Instanz wählt den Zahlenwandler NC1 aus, der Wert 20_{hex} den Zahlenwandler NC32.

Tabelle 321: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	–	–	→ Tab. 322	–	R
01	Mode	→ Tab. 323	–	–	–	R
02	Eingangswert I1	DWORD oder UDINT ¹⁾				R/W ²⁾
03	Ausgangswert QV	DWORD oder UDINT ¹⁾				R

1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte

2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 322: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		–	–	–	–	–	–	–	EN ¹⁾

1) Aktiviert den Baustein bei Zustand „1“

Tabelle 323: Index 1 – Mode

Data 1 (hex)		
00	BCD	Wandelt einen BCD-codierten Dezimalwert in einen ganzzahligen Wert um.
01	BIN	Wandelt einen ganzzahligen Wert in einen BCD-codierten Dezimalwert um.



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Betriebsstundenzähler OT1 bis OT4

Operanden-Auswahl	Wert (hex)
Type	1E
Instanz ¹⁾	01 - 04
Index	00 - 03, → Tabelle 324

1) Der Wert 01_{hex} für die Instanz wählt den Betriebsstundenzähler OT1 aus, der Wert 04_{hex} den Betriebsstundenzähler OT4.

Tabelle 324: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	→ Tab. 325	–	→ Tab. 325	–	R
02	Oberer Schwellwert I1	DWORD oder UDINT ¹⁾				R/W ²⁾
03	Istwert QV	DWORD oder UDINT ¹⁾				R

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte
- 2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 325: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	RE ¹⁾	EN ²⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q ³⁾

- 1) Resetspule: Zustand „1“ setzt den Zähler-Istwert auf Null zurück.
- 2) Freigabespule
- 3) Zustand „1“, wenn Sollwert erreicht wurde (größer/gleich)



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Impulsausgabe PO01...PO02

Operanden-Auswahl	Wert (hex)
Type	32
Instanz ¹⁾	01 - 02
Index	00 - 0A
Data 1 - 4	abhängig vom Index, → Tabelle 326

1) Der Wert 01hex wählt den Baustein PO01 aus, der Wert 02hex den Baustein PO02.

Tabelle 326: Operandenübersicht

Index (hex)	Daten	Data 1 Data 3	Data 2 Data 4	Read/ Write
0	Bit-IO	→ Tabelle 327	–	R
2	Impulsanzahl im Fahrbetrieb I1: 0 bis 2147483647	DWORD oder UDINT ¹⁾		R/W ²⁾
3	Startfrequenz FS: 0 bis 5000 Hz	DWORD oder UDINT ¹⁾		R/W ²⁾
4	Betriebsfrequenz FO: 0 bis 5000 Hz	DWORD oder UDINT ¹⁾		R/W ²⁾
5	Frequenzänderung im Hochlauf RF: 0 bis 65535 mHz	DWORD oder UDINT ¹⁾		R/W ²⁾
6	Frequenzänderung beim Bremsen BF: 0 bis 65535 mHz	DWORD oder UDINT ¹⁾		R/W ²⁾
7	Schrittzahl im Tippbetrieb P1: 0 bis 65535	DWORD oder UDINT ¹⁾		R/W ²⁾
8	Frequenz im Tippbetrieb PF: 0 bis 5000 Hz	DWORD oder UDINT ¹⁾		R/W ²⁾
9	Aktuelle Schrittzahl QV	DWORD oder UDINT ¹⁾		R
A	Aktuelle Frequenz QF	DWORD oder UDINT ¹⁾		R

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte.
- 2) Der Wert kann nur beschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 327: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	TP ¹⁾	BR ²⁾	ST ³⁾	EN ⁴⁾
FB-Ausgang Data 3		–	–	–	–	–	–	E1 ⁵⁾	AC ⁶⁾

- 1) Bei Erhalt einer positiven Flanke wird der Tippbetrieb gestartet.
- 2) Bei Erhalt einer positiven Flanke wird der Fahrauftrag abgebrochen.
- 3) Bei Erhalt einer positiven Flanke wird der Fahrauftrag gestartet.
- 4) Im Zustand „1“ wird der Funktionsbaustein aktiviert.
- 5) Zustand „1“ wird eingenommen, wenn die Parametereingabe ungültig ist.
- 6) Zustand „1“ wird eingenommen, wenn ein Fahrauftrag aktiv ist.



Weitere Informationen zu diesem Baustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easySoft-Hilfe.

Bausteine Senden Netzdaten PT1 bis PT32

Operanden-Auswahl	Wert (hex)
Type	1F
Instanz ¹⁾	01 - 20
Index	00 - 02
Data 1 - 4	abhängig vom Index, → Tabelle 328

1) Der Wert 01_{hex} wählt den Baustein PT1 aus, der Wert 20_{hex} den Baustein PT32.

Tabelle 328: Operandenübersicht

Index (hex)	Daten	Data 1 Data 3	Data 2 Data 4	Read/Write
0	Bit-IO	→ Tab. 329	–	R
2	Sollwert QV für das Netzwerk	DWORD oder UDINT ¹⁾		R

1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte.

Tabelle 329: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	T ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	E ¹⁾²⁾	AC ³⁾	Q ¹⁾⁴⁾

- 1) Triggerspule. Wird die Spule getriggert (erhält eine positive Flanke), so wird der entsprechende Wert dem NET zur Verfügung gestellt.
- 2) Zustand „1“ wird eingenommen, wenn der Sendeauftrag wegen eines Fehlers abgebrochen wurde.
- 3) Zustand „1“ wird eingenommen, wenn die Triggerspule getriggert wird. Zustand „0“ wird eingenommen, wenn der Sendeauftrag erfolgreich durchgeführt wurde oder wegen eines Fehlers abgebrochen wurde.
- 4) Zustand „1“, wenn der Zustand der Triggerspule ebenfalls „1“ ist.



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; vormals AWB2528-1423D) oder in der easySoft-Hilfe.

Pulsweitenmodulation PW1 bis PW2

Operanden-Auswahl	Wert (hex)
Type	2B
Instanz ¹⁾	01 - 02
Index	00 - 04, → Tabelle 330

1) Der Wert 01_{hex} für die Instanz wählt die Pulsweitenmodulation PW1 aus, der Wert 02_{hex} die Pulsweitenmodulation PW2.

Tabelle 330: Operandenübersicht

Index (hex)	Daten	Data 1 Data 3	Data 2 Data 4	Read/Write
00	Bit-IO	→ Tab. 331	–	R
02	Stellgröße SV Wertebereich 0 bis 4095 (12 Bit)	DWORD oder UDINT ¹⁾		R/W ²⁾
03	Periodendauer PD [ms], Wertebereich 0 bis 65535	DWORD oder UDINT ¹⁾		R/W ²⁾
04	Mindesteinschaltdauer ME [ms], Wertebereich 0 bis 65535	DWORD oder UDINT ¹⁾		R/W ²⁾

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte
- 2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 331: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	EN ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	E1 ²⁾

- 1) Aktiviert den Baustein bei Zustand „1“.
- 2) Zustand „1“, wenn die Mindesteinschaltdauer oder die Mindestausschaltdauer unterschritten wird



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Baustein Uhr synchronisieren SC1

Operanden-Auswahl	Wert (hex)
Type	20
Instanz	01
Index	00, → Tabelle 332

Tabelle 332: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	→ Tab. 333	–	→ Tab. 333	–	R

Tabelle 333: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	–	T ¹⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q1 ²⁾

- 1) Triggerspule. Wird die Spule getriggert (erhält eine positive Flanke), so wird automatisch das aktuelle Datum, der Wochentag und die Uhrzeit des sendenden Teilnehmers in das Netzwerk NET gestellt.
- 2) Zustand „1“, wenn der Zustand der Triggerspule SC01T_ ebenfalls „1“ ist.



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Serielle Ausgabe SP01...SP32

Operanden-Auswahl	Wert (hex)
Type	35
Instanz ¹⁾	01 - 20
Index	00
Data 1 - 4	abhängig vom Index, → Tabelle 334

1) Der Wert 01hex wählt den Baustein SP01 aus, der Wert 20hex den Baustein SP32.

Tabelle 334: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
0	Bit-IO	→ Tabelle 335	–	→ Tabelle 335	–	R

Tabelle 335: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	–	T ¹⁾	EN ²⁾
FB-Ausgang Data 3		–	–	–	–	–	–	E ¹³⁾	AC ⁴⁾

- 1) Bei Erhalt einer positiven Flanke wird der Sendevorgang ausgelöst.
- 2) Im Zustand „1“ wird der Funktionsbaustein aktiviert.
- 3) Der Zustand „1“ wird eingenommen, wenn beim Sendevorgang ein Fehler aufgetreten ist.
- 4) Der Zustand „1“ wird eingenommen, wenn der Sendevorgang aktiv ist.



Weitere Informationen zu diesem Baustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easySoft-Hilfe.

Schieberegister SR01...SR32

Operanden-Auswahl	Wert (hex)
Type	33
Instanz ¹⁾	01 - 20
Index	00 - 0B
Data 1 - 4	abhängig vom Index, → Tabelle 336

1) Der Wert 01hex wählt den Baustein SR01 aus, der Wert 20hex den Baustein SR32.

Tabelle 336: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2 Data 4	Data 3	Read/ Write
0	Bit-IO	→ Tabelle 337	–	→ Tabelle 337	R
1	Mode	→ Tabelle 338	–	–	R
2	Dateneingang vorwärts I1	DWORD oder UDINT ¹⁾			R/W ²⁾
3	Dateneingang rückwärts I2	DWORD oder UDINT ¹⁾			R/W ²⁾
4	Datenausgang 1 (D1)	DWORD oder UDINT ¹⁾			R
5	Datenausgang 2 (D2)	DWORD oder UDINT ¹⁾			R
6	Datenausgang 3 (D3)	DWORD oder UDINT ¹⁾			R
7	Datenausgang 4 (D4)	DWORD oder UDINT ¹⁾			R
8	Datenausgang 5 (D5)	DWORD oder UDINT ¹⁾			R
9	Datenausgang 6 (D6)	DWORD oder UDINT ¹⁾			R
A	Datenausgang 7 (D7)	DWORD oder UDINT ¹⁾			R
B	Datenausgang 8 (D8)	DWORD oder UDINT ¹⁾			R

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte.
- 2) Der Wert kann nur beschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 337: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	BD ¹⁾	FD ²⁾	RE ³⁾	BP ⁴⁾	FP ⁵⁾	EN ⁶⁾
FB-Ausgang Data 3		Q8 ⁷⁾	Q8 ⁷⁾	Q6 ⁷⁾	Q5 ⁷⁾	Q4 ⁷⁾	Q3 ⁷⁾	Q2 ⁷⁾	Q1 ⁷⁾

- 1) Eingangsbittwert für die Rückwärts-Schiebeoperation im Mode BIT
- 2) Eingangsbittwert für die Vorwärts-Schiebeoperation im Mode BIT
- 3) Im Zustand „1“ wird der Funktionsbaustein zurückgesetzt.
- 4) Bei Erhalt einer positiven Flanke wird im Mode BIT der Wert von BD in das letzte Register-Feld Q8 eingetragen und die ursprünglichen Inhalte der Registerfelder werden um ein Feld in Richtung niedrigerer Feldnummern verschoben. Bei Erhalt einer positiven Flanke wird im Mode DW der Wert von I2 in das letzte Register-Feld D8 eingetragen und die ursprünglichen Inhalte der Registerfelder werden um ein Feld in Richtung niedrigerer Feldnummern verschoben.
- 5) Bei Erhalt einer positiven Flanke wird im Mode BIT der Wert von FD in das erste Register-Feld Q1 eingetragen und die ursprünglichen Inhalte der Registerfelder werden um ein Feld in Richtung höherer Feldnummern verschoben. Bei Erhalt einer positiven Flanke wird im Mode DW der Wert von I1 in das erste Register-Feld D1 eingetragen und die ursprünglichen Inhalte der Registerfelder werden um ein Feld in Richtung höherer Feldnummern verschoben.
- 6) Im Zustand „1“ wird der Funktionsbaustein aktiviert.
- 7) Zustand der acht Felder des Bit-Schieberegisters

Tabelle 338: Index 1 – Mode

Data 1 (hex)		
00	BIT	Betriebsart: Bit schieben
01	DW	Betriebsart: Doppelwort schieben



Weitere Informationen zu diesem Baustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easySoft-Hilfe.

Baustein Sollzykluszeit ST1

Operanden-Auswahl	Wert (hex)
Type	2C
Instanz	01
Index	00 - 02, → Tabelle 339

Tabelle 339: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
0	Bit-IO	–	–	→ Tab. 340	–	R
2	Zykluszeit [ms], Wertebereich 0 bis 1000	DWORD oder UDINT ¹⁾				R

1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte

Tabelle 340: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		–	–	–	–	–	–	–	EN ¹⁾

1) Aktiviert den Baustein bei Zustand „1“



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Zeitrelais T1 bis T32

Operanden-Auswahl	Wert (hex)
Type	21
Instanz ¹⁾	01 - 20
Index	00 - 04, → Tabelle 341

1) Der Wert 01_{hex} für die Instanz wählt das Zeitrelais T1 aus, der Wert 20_{hex} das Zeitrelais T32.

Tabelle 341: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
0	Bit-IO	→ Tab. 342	–	→ Tab. 342	–	R
01	Mode/Parameter	→ Tab. 343	–	→ Tab. 343	–	R
02	Zeitsollwert 1	DWORD oder UDINT ¹⁾				R/W ²⁾
03	Zeitsollwert 2	DWORD oder UDINT ¹⁾				R/W ²⁾
04	Abgelaufene Istzeit	DWORD oder UDINT ¹⁾				R

1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte

2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 342: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	–	–	ST ¹⁾	EN ²⁾	RE ³⁾
FB-Ausgang Data 3		–	–	–	–	–	–	–	Q1 ⁴⁾

1) Stopp, das Zeitrelais wird gestoppt (Stoppspule)

2) Enable, das Zeitrelais wird gestartet (Triggerspule)

3) Reset, das Zeitrelais wird zurückgesetzt (Resetspule)

4) Schaltkontakt

Tabelle 343: Index 1 – Mode/Parameter

Mode	Data 1	Betriebsart
	0	Ansprechverzögert
	1	Ansprechverzögert mit Zufalls-Sollwert
	2	Rückfallverzögert
	3	Rückfallverzögert mit Zufalls-Sollwert
	4	Ansprechverzögert und rückfallverzögert (zwei Zeitsollwerte)
	5	Ansprechverzögert und rückfallverzögert, jeweils mit Zufalls-Sollwert (zwei Zeitsollwerte)
	6	Impulsgeber
	7	Blink-Relais (zwei Zeitsollwerte)
	8	Rückfallverzögert, retriggerbar (easy600 Mode)
	9	Rückfallverzögert mit Zufalls-Sollwert, retriggerbar (easy600 Mode)
Para- meter	Data 3	Betriebsart
	0	S (Millisekunden)
	1	M:S (Sekunden)
	2	H:M (Minuten)



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

Tabellenfunktion TB01...TB32

Operanden-Auswahl	Wert (hex)
Type	34
Instanz ¹⁾	01 - 20
Index	00 - 04
Data 1 - 4	abhängig vom Index, → Tabelle 344

1) Der Wert 01hex wählt den Baustein TB01 aus, der Wert 20hex den Baustein TB32.

Tabelle 344: Operandenübersicht

Index (hex)	Daten	Data 1 Data 3	Data 2 Data 4	Read/ Write
0	Bit-IO	→ Tabelle 345	–	R
2	Eingangswert I1 für Tabelle von TB...	DWORD oder UDINT ¹⁾		R/W ²⁾
3	Ausgangswert QV aus Tabelle von TB...	DWORD oder UDINT ¹⁾		R
4	Anzahl Einträge QN in Tabelle von TB...	DWORD oder UDINT ¹⁾		R

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte.
- 2) Der Wert kann nur beschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 345: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Eingang Data 1		–	–	–	RE ¹⁾	RL ²⁾	RF ³⁾	WP ⁴⁾	EN ⁵⁾
FB-Ausgang Data 3		–	–	–	–	–	–	TF ⁶⁾	TE ⁷⁾

- 1) Bei Erhalt einer positiven Flanke werden alle Einträge aus der Tabelle entfernt. Die Anzahl der Tabelleneinträge QN wird auf „0“ gesetzt.
- 2) Bei Erhalt einer positiven Flanke wird der neuste Eintrag in der Tabelle am Ausgang QV ausgegeben und aus der Tabelle entfernt. Die Anzahl der Tabelleneinträge QN wird um eins erniedrigt.
- 3) Bei Erhalt einer positiven Flanke wird der älteste Eintrag in der Tabelle am Ausgang QV ausgegeben und aus der Tabelle entfernt. Die Anzahl der Tabelleneinträge QN wird um eins erniedrigt.
- 4) Bei Erhalt einer positiven Flanke wird der Wert von I1 in die Tabelle übernommen und die Anzahl der Tabelleneinträge QN um eins erhöht, solange die maximale Anzahl von Einträgen nicht überschritten wird. In diesem Fall wird der Wert von I1 am Ausgang QV ausgegeben.
- 5) Im Zustand „1“ wird der Funktionsbaustein aktiviert.
- 6) Der Zustand „1“ wird eingenommen, wenn die Tabelle voll ist.
- 7) Der Zustand „1“ wird eingenommen, wenn die Tabelle leer ist



Weitere Informationen zu diesem Baustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easySoft-Hilfe.

Wertbegrenzung VC1 bis VC32

Operanden-Auswahl	Wert (hex)
Type	2D
Instanz ¹⁾	01 - 20
Index	00 - 05, → Tabelle 346

1) Der Wert 01_{hex} für die Instanz wählt die Wertbegrenzung VC1 aus, der Wert 20_{hex} die Wertbegrenzung VC32.

Tabelle 346: Operandenübersicht

Index (hex)	Daten	Data 1	Data 2	Data 3	Data 4	Read/Write
00	Bit-IO	–	–	→ Tab. 347	–	R
02	Eingangswert I1	DWORD oder UDINT ¹⁾				R/W ²⁾
03	Oberer Grenzwert SH	DWORD oder UDINT ¹⁾				R/W ²⁾
04	Unterer Grenzwert SL	DWORD oder UDINT ¹⁾				R/W ²⁾
05	Ausgangswert QV	DWORD oder UDINT ¹⁾				R

- 1) Wert wird im Intel-Format übergeben: Data 1 enthält Low-Byte, Data 4 enthält High-Byte
- 2) Der Wert kann nur geschrieben werden, wenn er im Programm mit einer Konstanten beschaltet ist.

Tabelle 347: Index 0 – Bit-IO

	Bit	7	6	5	4	3	2	1	0
FB-Ausgang Data 3		–	–	–	–	–	–	–	EN ¹⁾

1) Aktiviert den Baustein bei Zustand „1“



Weitere Informationen zu diesem Funktionsbaustein finden Sie im easy800-Handbuch (MN04902001Z-DE; frühere Bezeichnung AWB2528-1423D) oder in der easy-Soft-Hilfe.

**Datum und Uhrzeit lesen/
schreiben**

Objekt „Easy Uhr“

Die Beschreibung dieses Objektes ist identisch zum easy600 (DPV1), → Seite 294.

**Sommer-/Winterzeit lesen/
schreiben**

Objekte „Easy 800 Sommer-/Winterzeit Einstellung“

Dieses Objekt enthält die Sommer-/Winterzeit Einstellung des easy800/MFD.

Dieses Objekt ist lesbar (Read) und schreibbar (Write). Es kann von einem DPV1-Master der Klasse 1 und/oder Klasse 2 angesprochen werden.

Verwendungszweck

Verwenden Sie dieses Objekt, um

- die aktuelle Sommer-/Winterzeit Einstellung der Uhr eines easy800/MFD auszulesen
- eine neue Sommer-/Winterzeit Einstellung der Uhr eines easy800/MFD zu übergeben

Adressierung des Objektes

Für die Adressierung des Objektes verwenden Sie folgende Informationen:

- API ist gleich 0 (nur für DPV1-Master Klasse 2 erforderlich)
- Slot Number ist gleich 0
- Index ist gleich 209

Länge des Objektes

Die Länge der auszulesenden bzw. der zu schreibenden Daten beträgt 20 Octets. Tragen Sie beim Aufruf des Dienstes „Write“ keine andere Länge ein und beim Aufruf des Dienstes „Read“ keine kleinere Länge; ansonsten erhalten Sie eine Fehlermeldung.

Variablen-Definitionen (Beispiel) für das Objekt

In einem IEC 61131-3 basierenden System deklarieren Sie für das Lesen und Schreiben der Sommer-/Winterzeit Einstellung folgende Variable (Datenbaustein):

Name	Datentyp
Easy_800_Uhr_Umstellung	ARRAY [1...20] OF BYTE oder ARRAY [1...20] OF USINT

Dateninhalte des Objektes

Die folgende Tabelle zeigt die Adresslage und Bedeutung der Dateninhalte des Objektes beim Lesen oder Schreiben. Des Weiteren gibt sie beispielhaft an, wie Sie auf diese Dateninhalte mit Hilfe der beispielhaft definierten Variable zugreifen.

Tabelle 348: Adresslage und Bedeutung der Dateninhalte

Variablen-Zugriff (Beispiel)	Daten- position	Bedeutung	Wertebereich (dezimal)
Easy_800_Uhr_Umstellung[1]	Octet 1	Umschaltbedingung → Tabelle 349	0 bis 5
Easy_800_Uhr_Umstellung[2]	Octet 2	Tag der Umstellung für Wechsel auf Sommerzeit (Der Wechsel erfolgt um 2.00 Uhr auf 3.00 Uhr)	1 bis 31 BCD-Kodierung
Easy_800_Uhr_Umstellung[3]	Octet 3	Monat der Umstellung für Wechsel auf Sommerzeit	1 bis 12 BCD-Kodierung
Easy_800_Uhr_Umstellung[4]	Octet 4	Tag der Umstellung für Wechsel auf Winterzeit (Der Wechsel erfolgt um 3.00 Uhr auf 2.00 Uhr)	1 bis 31 BCD-Kodierung
Easy_800_Uhr_Umstellung[5]	Octet 5	Monat der Umstellung für Wechsel auf Winterzeit	1 bis 12 BCD-Kodierung
Easy_800_Uhr_Umstellung[6]	Octet 6	Regel 1 für Wechsel auf Sommerzeit → Tabelle 350	0 bis 5
Easy_800_Uhr_Umstellung[7]	Octet 7	Wochentag für Wechsel auf Sommerzeit → Tabelle 352	0 bis 6
Easy_800_Uhr_Umstellung[8]	Octet 8	Regel 2 für Wechsel auf Sommerzeit → Tabelle 351	0 bis 2
Easy_800_Uhr_Umstellung[9]	Octet 9	Tag der Umstellung für Wechsel auf Sommerzeit	1 bis 31 BCD-Kodierung
Easy_800_Uhr_Umstellung[10]	Octet 10	Monat der Umstellung für Wechsel auf Sommerzeit	1 bis 12 BCD-Kodierung

Variablen-Zugriff (Beispiel)	Daten- position	Bedeutung	Wertebereich (dezimal)
Easy_800_Uhr_ Umstellung[11]	Octet 11	Stunde der Umstellung für Wechsel auf Sommerzeit	0 bis 23 BCD-Kodierung
Easy_800_Uhr_ Umstellung[12]	Octet 12	Minute der Umstellung für Wechsel auf Sommerzeit	0 bis 59 BCD-Kodierung
Easy_800_Uhr_ Umstellung[13]	Octet 13	Zeitdifferenz	0 bis 5
Easy_800_Uhr_ Umstellung[14]	Octet 14	Regel 1 für Wechsel auf Winterzeit → Tabelle 350	0 bis 5
Easy_800_Uhr_ Umstellung[15]	Octet 15	Wochentag für Wechsel auf Winterzeit → Tabelle 352	0 bis 6
Easy_800_Uhr_ Umstellung[16]	Octet 16	Regel 2 für Wechsel auf Winterzeit → Tabelle 351	0 bis 2
Easy_800_Uhr_ Umstellung[17]	Octet 17	Tag der Umstellung für Wechsel auf Winterzeit	1 bis 31 BCD-Kodierung
Easy_800_Uhr_ Umstellung[18]	Octet 18	Monat der Umstellung für Wechsel auf Winterzeit	1 bis 12 BCD-Kodierung
Easy_800_Uhr_ Umstellung[19]	Octet 19	Stunde der Umstellung für Wechsel auf Winterzeit	0 bis 23 BCD-Kodierung
Easy_800_Uhr_ Umstellung[20]	Octet 20	Minute der Umstellung für Wechsel auf Winterzeit	0 bis 59 BCD-Kodierung

Tabelle 349: Umschaltbedingungen für Sommer-/Winterzeit
Wechsel

Wert (hex)	Bedeutung	Hinweis
00	keine Umschaltung	Octet 2 bis 20 sind ohne Bedeutung
01	Umschaltung erfolgt gemäß Octet 2 und 3 für Wechsel auf Sommerzeit Octet 4 und 5 für Wechsel auf Winterzeit	Octet 6 bis 20 sind ohne Bedeutung
02	Umschaltung erfolgt gemäß Regeln für EU	Octet 2 bis 20 sind ohne Bedeutung
03	Umschaltung erfolgt gemäß Regeln für GB	Octet 2 bis 20 sind ohne Bedeutung
04	Umschaltung erfolgt gemäß Regeln für USA	Octet 2 bis 20 sind ohne Bedeutung
05	Umschaltung erfolgt gemäß Octet 6 bis 13 für Wechsel auf Sommerzeit Octet 13 bis 20 für Wechsel auf Winterzeit	Octet 2 bis 5 sind ohne Bedeutung

Ist in Octet 1 der Wert 0 eingetragen, erfolgt keine Umstellung der Uhr auf Sommer-/Winterzeit. Bei Auswahl der Werte 2 bis 4 erfolgt die Umstellung der Uhr automatisch gemäß den gesetzlichen Bestimmungen für das ausgewählte Land.

Wird in Octet 1 der Wert 1 angegeben, erfolgt die Umstellung der Uhr auf Sommer-/Winterzeit an den in Octet 2 bis 5 definierten Daten.

Wird in Octet 1 der Wert 5 angegeben, erfolgt die Umstellung der Uhr auf Sommer-/Winterzeit gemäß den selbst definierten Regeln in Octet 6 bis 20.

Octet 6 enthält die erste Regel für die Umstellung auf Sommerzeit, Octet 14 für die Umstellung auf Winterzeit. Tabelle 350 zeigt die möglichen Regeln an.

Tabelle 350: Regel 1 für Sommer-/Winterzeit Wechsel

Wert (hex)	Bedeutung	
00	Umschaltung erfolgt an dem Zeitpunkt, der definiert ist in...	... • Octet 9 bis 12 für Wechsel auf Sommerzeit • Octet 17 bis 20 für Wechsel auf Winterzeit In diesem Fall sind ohne Bedeutung: • Octet 7 und 8 bei Wechsel auf Sommerzeit • Octet 15 und 16 bei Wechsel auf Winterzeit
01	Umschaltung erfolgt am ersten Wochentag, der definiert ist in...	... • Octet 7 für Wechsel auf Sommerzeit • Octet 15 für Wechsel auf Winterzeit in Zusammenhang mit der definierten Regel 2 in: • Octet 8 bei Wechsel auf Sommerzeit • Octet 16 bei Wechsel auf Winterzeit
02 ¹⁾	Umschaltung erfolgt am zweiten Wochentag, der definiert ist in...	
03 ¹⁾	Umschaltung erfolgt am dritten Wochentag, der definiert ist in...	
04 ¹⁾	Umschaltung erfolgt am vierten Wochentag, der definiert ist in...	
05 ¹⁾	Umschaltung erfolgt am letzten Wochentag, der definiert ist in...	

1) In der Regel 2 muss die Bedeutung „im“ eingetragen sein. Die Regel 2 wird definiert in:

- Octet 8 bei Wechsel auf Sommerzeit
- Octet 16 bei Wechsel auf Winterzeit

Enthält Octet 6 oder Octet 14 einer der Werte 1 bis 5, muss auch die zweite Regel für die Umstellung auf Sommerzeit (Octet 8) oder Winterzeit (Octet 16) definiert werden. Tabelle 351 zeigt die möglichen Regeln an.

Tabelle 351: Regel 2 für Sommer-/Winterzeit Wechsel

Wert (hex)	Bedeutung	Hinweis
00 ¹⁾	Umschaltung erfolgt an dem definierten Wochentag und Monat	Der Wochentag wird definiert im: <ul style="list-style-type: none"> • Octet 7 bei Wechsel auf Sommerzeit • Octet 15 beim Wechsel auf Winterzeit Monat bzw. Datum werden definiert im: <ul style="list-style-type: none"> • Octet 9 bzw. 10 bei Wechsel auf Sommerzeit • Octet 17 bzw. 18 bei Wechsel auf Winterzeit
01	Umschaltung erfolgt an dem definierten Wochentag nach dem definiertem Datum	
02	Umschaltung erfolgt an dem definierten Wochentag vor dem definiertem Datum	

1) In diesem Fall sind ohne Bedeutung:

- Octet 9 bei Wechsel auf Sommerzeit
- Octet 17 bei Wechsel auf Winterzeit

Tabelle 352: Kodierung des Wochentags

Wert (hexadezimal)	Bedeutung
00	Sonntag
01	Montag
02	Dienstag
03	Mittwoch
04	Donnerstag
05	Freitag
06	Samstag

Tabelle 353: Kodierung der Zeitdifferenz (Octet 13)

Wert (hexadezimal)	Zeitdifferenz der Umstellung beträgt
01	30 Minuten
02	1 Stunde
03	90 Minuten
04	2 Stunden
05	150 Minuten
06	3 Stunden

DPV1-Fehlermeldungen Das EASY204-DP erzeugt die in der nachfolgenden Tabelle enthaltenen DPV1-Fehlermeldungen.

Tabelle 354: DPV1-Fehlermeldungen

Fehlermeldung/-code	Bedeutung	Erläuterung/Abhilfe
Access denied (Code B6 hexadezimal)	Das angesprochene Objekt ist für den Typ des DP-Masters gesperrt	Sie haben mit einem DP-Master der Klasse 1 ein Objekt angesprochen, das nur von einem DPV1-Master der Klasse 2 angesprochen werden kann. Beachten Sie die Objektbeschreibung.
	Das angesprochene Objekt ist für den ausgewählten Dienst gesperrt	Sie haben entweder mit einem „Write“-Dienst auf ein Objekt zugegriffen, das nur lesbar ist, oder mit einem „Read“-Dienst auf ein Objekt zugegriffen, das nur schreibbar ist. Beachten Sie die Objektbeschreibung.
	Das angeschlossene Easy/MFD hat für den gewählten Dienst einen Fehlercode erzeugt	Beachten Sie die Objektbeschreibung
Invalid index (Code B0 hexadezimal)	Der im „Read“- oder „Write“-Dienst angegebene Index ist für das EASY204-DP ungültig	Verwenden Sie den in der Objektbeschreibung angegebenen Index
Invalid parameter (Code B8 hexadezimal)	Die beim „Read“-Dienst verwendete Datenlänge ist kleiner als die Datenlänge des angesprochenen Objektes	Verwenden Sie die in der Objektbeschreibung angegebene Datenlänge
	Die beim Verbindungsaufbau des DPV1-Master der Klasse 2 verwendeten Parameter stimmen nicht mit denen des EASY204-DP überein	Der DPV1-Master der Klasse 2 muss beim Verbindungsaufbau die Parameter „Features Supported“ auf 1 und „Profile Ident Number“ auf 0 setzen

Fehlermeldung/-code	Bedeutung	Erläuterung/Abhilfe
Invalid range (Code B7 hexadezimal)	Beim „Write“-Dienst wurde für einen Dateninhalt ein unzulässiger Wert benutzt	Beachten Sie die Wertebereiche der Dateninhalte gemäß Objektbeschreibung
Invalid slot (Code B2 hexadezimal)	Der im „Read“- oder „Write“-Dienst angegebene Slot ist für das EASY204-DP ungültig	Verwenden Sie den in der Objektbeschreibung angegebenen Slot
Resource unavailable (Code C3 hexadezimal)	Das angesprochene Objekt ist zurzeit nicht erreichbar, da die Kommunikation zwischen EASY204-DP und angeschlossenem Easy/MFD gestört ist.	Überprüfen Sie die Verbindung zwischen den Geräten
Write length error (Code B1 hexadezimal)	Die beim „Write“-Dienst verwendete Datenlänge stimmt nicht mit der Datenlänge des angesprochenen Objektes überein	Verwenden Sie die in der Objektbeschreibung angegebene Datenlänge

Anhang

Was ist wenn...?

Ereignis	Erklärung	Abhilfe
POW-LED leuchtet nicht	Keine Versorgungsspannung	Stromversorgung anschließen und einschalten
POW-LED blinkt	Datentransfer über den easyLink in Ordnung	
BUS-LED leuchtet nicht	Keine PROFIBUS-DP-Datenkommunikation	PROFIBUS-DP anschließen und betreiben
BUS-LED leuchtet	Datentransfer über den PROFIBUS-DP ist in Ordnung	
Slave meldet sich nicht	<ul style="list-style-type: none"> – Keine Slaveadresse eingestellt – Abschlusswiderstand nicht vorhanden – Leitung, Stecker defekt – Keine Versorgungsspannung 	<ul style="list-style-type: none"> – Slaveadresse einstellen – Abschlusswiderstände setzen – Anschluss überprüfen – Gerät mit Spannung versorgen
Schreibenbefehl zurückgewiesen	<ul style="list-style-type: none"> – Befehl ist nicht erlaubt – easy-Display steht nicht auf der Statusanzeige 	<ul style="list-style-type: none"> – Befehl ändern – Statusdisplay anzeigen
Istwert ist gleich Null	Kein Istwert vorhanden	Funktionsrelais besitzt keinen Istwert oder ist nicht getriggert

Übersicht der Befehle

easy600

Die Befehle sind in aufsteigender Reihenfolge sortiert:

Befehl Wert (hex)	
01	T1 Zeitsollwert schreiben
02	T2 Zeitsollwert schreiben
03	T3 Zeitsollwert schreiben
04	T4 Zeitsollwert schreiben
05	T5 Zeitsollwert schreiben
06	T6 Zeitsollwert schreiben
07	T7 Zeitsollwert schreiben
08	T8 Zeitsollwert schreiben
09	C1 Zählerrelais Sollwert schreiben
0A	C2 Zählerrelais Sollwert schreiben
0B	C3 Zählerrelais Sollwert schreiben
0C	C4 Zählerrelais Sollwert schreiben
0D	C5 Zählerrelais Sollwert schreiben
0E	C6 Zählerrelais Sollwert schreiben
0F	C7 Zählerrelais Sollwert schreiben
10	C8 Zählerrelais Sollwert schreiben
12	Schaltuhr 1 Kanal A schreiben
13	Schaltuhr 1 Kanal B schreiben
14	Schaltuhr 1 Kanal C schreiben
15	Schaltuhr 1 Kanal D schreiben
16	Schaltuhr 2 Kanal A schreiben
17	Schaltuhr 2 Kanal B schreiben
18	Schaltuhr 2 Kanal C schreiben
19	Schaltuhr 2 Kanal D schreiben
1A	Schaltuhr 3 Kanal A schreiben

Befehl Wert (hex)	
1B	Schaltuhr 3 Kanal B schreiben
1C	Schaltuhr 3 Kanal C schreiben
1D	Schaltuhr 3 Kanal D schreiben
1E	Schaltuhr 4 Kanal A schreiben
1F	Schaltuhr 4 Kanal B schreiben
20	Schaltuhr 4 Kanal C schreiben
21	Schaltuhr 4 Kanal D schreiben
22	A1 Analogwertvergleich schreiben
23	A2 Analogwertvergleich schreiben
24	A3 Analogwertvergleich schreiben
25	A4 Analogwertvergleich schreiben
26	A5 Analogwertvergleich schreiben
27	A6 Analogwertvergleich schreiben
28	A7 Analogwertvergleich schreiben
29	A8 Analogwertvergleich schreiben
2A	Uhrzeit schreiben
2B	Schaltuhr 1 Kanal A lesen
2C	Schaltuhr 1 Kanal B lesen
2D	Schaltuhr 1 Kanal C lesen
2E	Schaltuhr 1 Kanal D lesen
2F	Schaltuhr 2 Kanal A lesen
30	Schaltuhr 2 Kanal B lesen
31	Schaltuhr 2 Kanal C lesen
32	Schaltuhr 2 Kanal D lesen
33	Schaltuhr 3 Kanal A lesen
34	Schaltuhr 3 Kanal B lesen
35	Schaltuhr 3 Kanal C lesen
36	Schaltuhr 3 Kanal D lesen

Befehl Wert (hex)	
37	Schaltuhr 4 Kanal A lesen
38	Schaltuhr 4 Kanal B lesen
39	Schaltuhr 4 Kanal C lesen
3A	Schaltuhr 4 Kanal D lesen
3C	Uhrzeit lesen
3D	Wert und Zustand Analog- und Digital-Eingänge lesen
3E	Zustand P- und bedientasten lesen
3F	Zustand Zeitrelais, Zählerrelais, Zeitschaltuhren, Analogwertvergleicher lesen
40	Zustand Hilfsrelais (Merker), digitale Ausgänge und Textanzeige lesen
41	T1 Istwert lesen
42	T2 Istwert lesen
43	T3 Istwert lesen
44	T4 Istwert lesen
45	T5 Istwert lesen
46	T6 Istwert lesen
47	T7 Istwert lesen
48	T8 Istwert lesen
49	C1 Zählerrelais Istwert lesen
4A	C2 Zählerrelais Istwert lesen
4B	C3 Zählerrelais Istwert lesen
4C	C4 Zählerrelais Istwert lesen
4D	C5 Zählerrelais Istwert lesen
4E	C6 Zählerrelais Istwert lesen
4F	C7 Zählerrelais Istwert lesen
50	C8 Zählerrelais Istwert lesen

easy800/MFD

Datum und Uhrzeit	Byte 1 Befehl (hex)	Byte 2 Len¹⁾ (hex)	Byte 3 Index (hex)
Datum und Uhrzeit lesen/schreiben	93/B3	05	00
Winter-/Sommerzeit, Zeitumstellung			01

Abbilddaten	Byte 1 Befehl (hex)	Byte 2 Len¹⁾ (hex)	Byte 3 Type (hex)	Byte 4 Index (dez)
Abbilddaten lesen/schreiben	91/B1			
„Lokale Eingänge: I1 - I16“		2	01	0
„Eingänge der Netzwerkteilnehmer IW1 bis IW8 lesen“				1 - 8
„Lokale Analog-Eingänge IA1 bis IA4 lesen“			02	1 - 4
„Lokale Diagnose ID1 bis ID16 lesen“			03	0
„Lokale Ausgänge QW0/Ausgänge der Netzwerkteilnehmer QW1 bis QW8 lesen und schreiben“			04	0/1 - 8
„Lokalen Analog-Ausgang QA1 lesen und schreiben“			05	0
„Lokale P-Tasten lesen“		1	06	0
„Eingänge RW.../Ausgänge SW... von easyLink lesen“		2	07/09	0
„Receive-Data Netz RN1...RN32/ Send-Data Netz SN1...SN32 lesen“				1 - 8
„Receive-Data Netz RN1...RN32/ Send-Data Netz SN1...SN32 lesen“		4	08/0A	1 - 8
Merker-Bit M1...M96		1	0B	1 - 96
Merker-Byte MB1...MB96			0C	1 - 96
Merker-Word MW1...MW96		2	0D	1 - 96
Merker-Doppelwort MD1...MD96		4	0E	1 - 96

1) Len... steht für die Anzahl der zu sendenden Datenbytes.

Funktionsbausteine	Byte 1 Befehl (hex)	Byte 2 Type (hex)	Byte 3 Instanz (hex)
Funktionsbausteine lesen/schreiben	92/B2		
„Empfang von Netzdaten GT01...GT32“		0F	1 - 20
„Analogwertvergleichler A01...A32“		11	1 - 20
„Arithmetikbaustein AR01...AR32“		12	1 - 20
„Boolesche Verknüpfung BV01...BV32“		13	1 - 20
„Zähler C01...C32“		14	1 - 20
„Frequenzzähler CF01...CF04“		15	1 - 20
„Schneller Zähler CH01...CH04“		16	1 - 4
„Inkrementalzähler CI01...CI02“		17	1 - 2
„Vergleicher CP01...CP32“		18	1 - 20
„Textausgabe-Baustein D01...D32“		19	1 - 20
„Datenbaustein DB01...DB32“		1A	1 - 20
„Empfang von Netzdaten GT01...GT32“		1B	1 - 20
„Wochenzeitschaltuhr HW01...HW32“		1C	1 - 20
„Jahreszeitschaltuhr HY01...HY32“		1D	1 - 20
„Betriebsstundenzähler OT01...OT04“		1E	1 - 4
„Senden von Netzdaten PT01...PT32“		1F	1 - 20
„Uhr synchronisieren SC01“		20	1
„Sollzykluszeit ST01“		21	1 - 20
„Block Compare BC01...BC32“		25	1 - 20
„Block Transfer BT01...BT32“		26	1 - 20
„PID-Regler DC01...DC32“		27	1 - 20
„Signalglättungsfilter FT01...FT32“		28	1 - 20
„Wertskalierung LS01...LS32“		29	1 - 20
„Zahlenwandler NC01...NC32“		2A	1 - 20
„Pulsweitenmodulation PW01...PW02“		2B	1 - 2
„Sollzykluszeit ST01“		2C	1
„Wertbegrenzung VC01...VC32“		2D	1 - 20

Funktionsbausteine	Byte 1	Byte 2	Byte 3
	Befehl (hex)	Type (hex)	Instanz (hex)
„Bedingter Sprung JC01...JC32“		2F	1 - 20
„Datenmultiplexer MX01...MX32“		31	1 - 20
„Impulsausgabe PO01...PO02“		32	1 - 2
„Schieberegister SR01...SR32“		33	1 - 20
„Tabellenfunktion TB01...TB32“		33	1 - 20
„Serielle Ausgabe SP01...SP32“		34	1 - 20
„Diagnose DG01...DG16“		39	1 - 10

Technische Daten

Allgemeines

Normen und Bestimmungen	EN 55011, EN 55022, IEC/EN 61-4, IEC 60068-2-27, IEC 61158
Abmessungen (B × H × T)	35,5 × 90 × 56,5
Gewicht	150 g
Montage	Hutschiene DIN 50022, 35 mm Schraubmontage mit Gerätefüße ZB4-101-GF1 (Zubehör)

Klimatische Umgebungstemperaturen

Betriebsumgebungstemperatur waagerechter/senkrechter Einbau	Kälte nach IEC 60068-2-1 Wärme nach IEC 60068-2-2	-25 bis 55 °C
Betauung		Betauung durch geeignete Maßnahmen verhindern
Lager-/Transporttemperatur		-40 bis +70 °C
Relative Luftfeuchte	IEC 60068-2-30	5 bis 95 %, keine Betauung
Luftdruck (Betrieb)		795 bis 1080 hPa
Korrosionsunempfindlichkeit	IEC 60068-2-42 IEC 60068-2-43	SO ₂ 10 cm ³ /m ³ , 4 Tage H ₂ S 1 cm ³ /m ³ , 4 Tage

Mechanische Umgebungsbedingungen

Verschmutzungsgrad		2
Schutzart	EN 50 178 IEC 60 529 VBG4	IP20
Schwingungen	IEC 60068-2-6	10 bis 57 Hz (konstante Amplitude 0,15 mm) 57 bis 150 Hz (konstante Beschleunigung 2 g)
Schocken	IEC 60068-2-27	18 Schocks (Halbsinus 15 g/11 ms)
Kippfallen	IEC 60068-2-31	Fallhöhe 50 mm
Freier Fall, verpackt	IEC 60068-2-32	1 m

Elektromagnetische Verträglichkeit (EMV)

Elektrostatische Entladung	IEC/EN 61 000-4-2, Schärfegrad 3	8 kV Luftentladung 6 kV Kontaktentladung
Elektromagnetische Felder	IEC/EN 61 000-4-3	Feldstärke 10 V/m
Funkentstörung	EN 55 011, EN 55 022	Grenzwertklasse A
Burst Impulse	IEC/EN 61 000-4-4, Schärfegrad 3	2 kV Versorgungs-, 1 kV Signalleitungen
Energiereiche Impulse (Surge)		
EASY...-DC...	IEC/EN 61 000-4-5, Schärfegrad 2	0,5 kV Versorgungsleitung symmetrisch
Einströmung	IEC/EN 61 000-4-6	10 V

Isolationsfestigkeit

Bemessung der Luft- und Kriechstrecken	EN 50 178, UL 508, CSA C22.2 No 142
Isolationsfestigkeit	EN 50 178

Werkzeug und Anschlussquerschnitte

eindrätig	
min.	0,2 mm ² , AWG 22
max.	4 mm ² , AWG 12
feindrätig mit Aderendhülse	
min.	0,2 mm ² , AWG 22
max.	2,5 mm ² , AWG 12
Schlitzschraubendreherbreite in	3,5 × 0,8 mm
Anzugsdrehmoment max.	0,5 Nm

Stromversorgung

Bemessungsspannung	
Nennwert	24 V DC, -15 %, +20 %
Zulässiger Bereich	20,4 bis 28,8 V DC
Restwelligkeit	< 5 %
Eingangsstrom bei 24 V DC	typ. 200 mA
Spannungseinbrüche (IEC/EN 61 131-2)	10 ms
Verlustleistung bei 24 V DC	typ. 4,8 W

LED-Anzeigen

LED-Power (POW)	grün
LED-PROFIBUS-DP (BUS)	grün

PROFIBUS-DP

Anschluss Gerät	SUB-D 9-polig, Buchse
Potentialtrennung	Bus zu Versorgungsspannung (einfach) Bus und Stromversorgung zu „easy“-Basisgerät (sichere Trennung)
Funktion	PROFIBUS-DP-Slave
Schnittstelle	RS 485
Busprotokoll	PROFIBUS-DP
Baudraten	Automatische Suche bis 12 MBd
Busabschlusswiderstände	Über Stecker zuschaltbar
Busadressen	1 bis 126 über „easy“-Basisgerät mit Display oder easySoft adressierbar
Dienste	
Modul Eingänge	alle Daten S1 bis S8 (EASY6...)
Modul Ausgänge	alle Daten R1 bis R16 (EASY6...)
Modul Steuerbefehle	Read/Write Uhrzeit, Tag, Sommer/Winterzeit Alle Parameter der easy-Funktionsrelais

Abmessungen

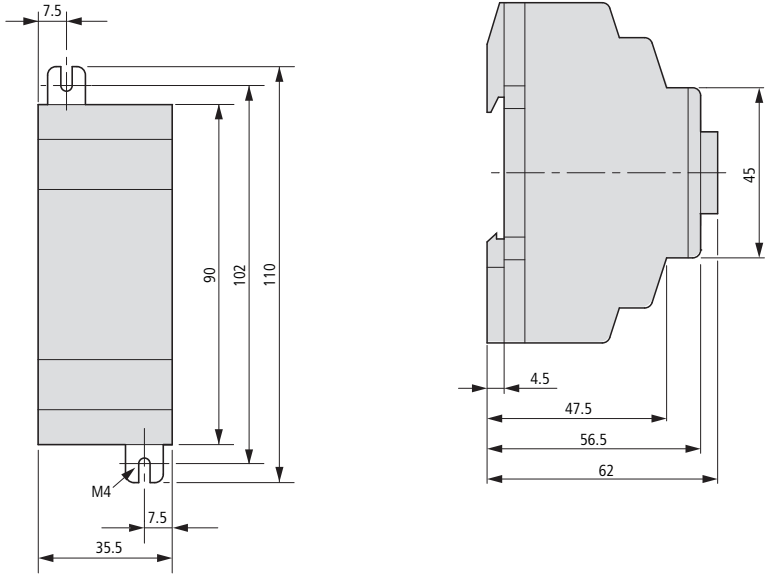


Abbildung 9: Maße EASY204-DP

Glossar

Das Glossar bezieht sich auf Themen rund um PROFIBUS-DP.

Abschlusswiderstand	Widerstand am Anfang und am Ende einer Bus-Leitung, der störende Signalreflexionen verhindert und zur Leistungsanpassung bei Busleitungen dient. Abschlusswiderstände müssen immer die letzte Einheit am Ende eines Bussegments sein.
Acknowledge	Quittung des Empfängers für ein empfangenes Signal.
Adresse	Nummer zur Kennzeichnung z. B. eines Speicherplatzes, eines Systems oder eines Moduls innerhalb eines Netzwerks.
Adressierung	Zuweisung bzw. Einstellung einer Adresse, z. B. für ein Modul in einem Netzwerk.
aktives Metallteil	Leiter oder leitfähiges Bauteil, das im Betrieb unter Spannung steht.
Automatisierungsgerät	Gerät zur Steuerung mit Eingängen und Ausgängen, das an einen technischen Prozess angeschlossen wird. Speicherprogrammierbare Steuerungen (SPS) sind eine spezielle Gruppe von Automatisierungsgeräten.
analog	Wert – z. B. einer Spannung – der sich stufenlos proportional verhält. Bei analogen Signalen kann der Wert des Signals innerhalb bestimmter Grenzen jeden beliebigen Wert annehmen.
Baud	Maß für die Übertragungsgeschwindigkeit von Daten. Ein Baud entspricht der Übertragung von einem Bit pro Sekunde (Bit/s).
Baud-Rate	Maßeinheit für die Geschwindigkeit bei der Datenübertragung in Bit/s.
Betriebsmittel, elektrische	Alle Gegenstände, die für die Erzeugung, Umwandlung, Übertragung, Verteilung und Anwendung von elektrischer Energie eingesetzt werden, z. B. Leitungen, Kabel, Maschinen, Steuergeräte.

Bezugserde	Potenzial des Erdreichs im Bereich von Erdungseinrichtungen. Kann im Gegensatz zur „Erde“, deren Potenzial immer Null ist, ein von Null verschiedenes Potenzial haben.
Bezugspotenzial	Potenzial, von dem aus die Spannungen aller angeschlossenen Stromkreise betrachtet und/oder gemessen werden.
bidirektional	In beiden Richtungen arbeitend.
Blitzschutz	Alle Maßnahmen, die dazu dienen, ein System vor Schäden durch Überspannungen zu schützen, die von Blitzen hervorgerufen werden können.
Bus	Sammelleitungssystem für den Datenaustausch, z. B. zwischen CPU, Speicher und I/O-Ebene. Ein Bus kann aus mehreren parallelen Leitungen für Datenübertragung, Adressierung, Steuerung und Stromversorgung bestehen.
Buslinie	Kleinste mit einem Bus verbundene Einheit bestehend aus einer SPS, einem Kopplungselement für Module an den Bus und einem Modul.
Bussystem	Die Gesamtheit aller Einheiten, die über einen Bus miteinander kommunizieren.
Buszykluszeit	Zeitintervall, in dem ein Master alle Slaves bzw. Teilnehmer in einem Bussystem bedient, d. h. deren Ausgänge schreibt und Eingänge liest.
CPU	Abkürzung für englisch „Central Processing Unit“. Zentrale Einheit zur Datenverarbeitung, das Kernstück eines Rechners.
digital	Wert – z. B. einer Spannung – der innerhalb einer endlichen Menge nur bestimmte Zustände annehmen kann, meist definiert als „0“ und „1“.
DIN	Abkürzung für „Deutsches Institut für Normungen e.V.“.
EMV	Abkürzung für „Elektromagnetische Verträglichkeit“. Die Fähigkeit eines elektrischen Betriebsmittels, in einer bestimmten Umgebung fehlerfrei zu funktionieren, ohne negativen Einfluss auf die Umgebung zu haben.
EN	Abkürzung für „Europäische Norm“.

Erde	In der Elektrotechnik die Bezeichnung für leitfähiges Erdreich, dessen elektrisches Potenzial an jedem Punkt gleich Null ist. In der Umgebung von Erdungseinrichtungen kann das elektrische Potenzial der Erde ungleich Null sein, dann spricht man von „Bezugserde“.
erden	Verbinden eines elektrisch leitfähigen Teils über eine Erdungseinrichtung mit dem Erder.
Erder	Eine oder mehrere Komponenten, die mit dem Erdreich direkten und guten Kontakt haben.
ESD	Abkürzung für englisch “Electro Static Discharge”, elektrostatische Entladung.
Feldbus	Datennetz auf der Sensor-/Aktorebene. Ein Feldbus verbindet die Geräte in der Feldebene. Kennzeichnend für einen Feldbus sind hohe Übertragungssicherheit und Echtzeitverhalten.
Feldeinspeisung	Einspeisung der Spannung zur Versorgung der Feldgeräte sowie der Signalspannung.
galvanische Kopplung	Eine galvanische Kopplung tritt generell auf, wenn zwei Stromkreise eine gemeinsame Leitung benutzen. Typische Störquellen sind z. B. anlaufende Motoren, statische Entladungen, getaktete Geräte und ein unterschiedliches Potenzial der Gehäuse von Komponenten und der gemeinsamen Stromversorgung.
GND	Abkürzung für englisch “GROUND”, deutsch Masse (Potenzial 0).
GSD	Die Gerätestammdaten (GSD) enthalten vereinheitlichte PROFIBUS-Teilnehmer-Beschreibungen. Sie dienen zur Vereinfachung der Projektierung des DP-Masterss und der DP-Slaves.
hexadezimal	Zahlensystem mit der Basis 16. Gezählt wird von 0 bis 9 und weiter mit den Buchstaben A, B, C, D, E und F.
I/O	Abkürzung für englisch “Input/Output”, Eingabe/Ausgabe.
Impedanz	Scheinwiderstand, den ein Bauelement oder eine Schaltung aus mehreren Bauelementen für einen Wechselstrom einer bestimmten Frequenz besitzt.

impedanzarme Verbindung	Verbindung mit geringem Wechselstromwiderstand.
inaktive Metallteile	Nicht berührbare leitfähige Elemente, die von den aktiven Metallteilen durch eine Isolierung elektrisch getrennt sind, im Fehlerfall jedoch Spannung annehmen können.
induktive Kopplung	Eine induktive (magnetische) Kopplung tritt zwischen zwei stromdurchflossenen Leitern auf. Die durch die Ströme hervorgerufene magnetische Wirkung induziert eine Störspannung. Typische Störquellen sind z. B. Transformatoren, Motoren, parallel laufende Netzkabel und HF-Signalkabel.
kapazitive Kopplung	Eine kapazitive (elektrische) Kopplung tritt zwischen Leitern auf, die sich auf unterschiedlichen Potenzialen befinden. Typische Störquellen sind z. B. parallel verlaufende Signalkabel, Schütze und statische Entladungen.
Kodierelement	Zweiteiliges Element zur eindeutigen Zuordnung von Elektronik- und Basismodul.
kommandofähige Module	Kommandofähige Module sind Module mit internem Speichersatz, die in der Lage sind, bestimmte Befehle (z. B. Ersatzwerte auszugeben) auszuführen.
Konfigurieren	Systematisches Anordnen der I/O-Module einer Station.
kurzschlussfest	Eigenschaft von elektrischen Betriebsmitteln. Ein kurzschlussfestes Betriebsmittel hält den thermischen und dynamischen Belastungen, die an seinem Installationsort aufgrund eines Kurzschlusses auftreten können, stand.
LSB	Abkürzung für englisch "Least Significant Bit". Bit mit dem niedrigsten Stellenwert.
Masse	Gesamtheit aller untereinander verbundenen inaktiven Teile eines Betriebsmittels, die auch im Fehlerfall keine Berührungsspannung annehmen.
Masseband	Flexibler Leiter, meist geflochten, der die inaktiven Teile eines Betriebsmittels verbindet, z. B. die Tür eines Schaltschranks mit dem Schaltschrankkorpus.
Master	Station oder Teilnehmer in einem Bussystem, die/der die Kommunikation zwischen den anderen Teilnehmern des Bussystems steuert.

Master-Slave Mode	Betriebsart, bei der eine Station oder ein Teilnehmer im System als Master die Kommunikation über den Bus leitet.
Mode	englisch, deutsch Betriebsart (Modus).
Modulbus	Der Modulbus ist der interne Bus einer XI/ON-Station. Über ihn kommunizieren die XI/ON-Module mit dem Gateway. Er ist unabhängig vom Feldbus.
MSB	Abkürzung für englisch "Most Significant Bit". Bit mit dem höchsten Stellenwert.
Multimaster Mode	Betriebsart, bei der alle Stationen oder Teilnehmer im System gleichberechtigt über den Bus kommunizieren können.
Namur	„Normen-Arbeitsgemeinschaft für Mess- und Regeltechnik“. Namur-Initiatoren sind Sonderausführungen der Zweidrahtinitiatoren. Aufgrund der besonderen Konstruktion – niedriger Innenwiderstand, wenige Bauteile, kurze Bauform – zeichnen sich Namur-Initiatoren durch eine hohe Stör- und Betriebssicherheit aus.
Overhead	Systemverwaltungszeit, die bei jedem Übertragungszyklus einmal im System benötigt wird.
Parametrieren	Festlegen von Parametern der einzelnen Busteilnehmer bzw. ihrer Module in der Konfigurationssoftware des DP-Masters.
Potenzialausgleich	Die Angleichung der elektrischen Niveaus der Körper elektrischer Betriebsmittel und fremder, leitfähiger Körper durch eine elektrische Verbindung.
potenzialfrei	Galvanische Trennung der Bezugspotenziale von Steuer- und Laststromkreisen bei I/O-Modulen.
potenzialgebunden	Elektrische Verbindung der Bezugspotenziale von Steuer- und Laststromkreisen bei I/O-Modulen.

PROFIBUS-DP	<p>PROFIBUS-Bussystem mit DP-Protokoll. DP steht für „dezentrale Peripherie“.</p> <p>Der PROFIBUS-DP basiert auf DIN 19245 Teil 1+4 und wurde in die europäische Feldbusnorm EN 50 170 integriert. Er dient zum schnellen zyklischen Datenaustausch zwischen dem zentralen DP-Master und den dezentralen Peripheriegeräten, den DP-Slaves. Der durchgängige Einsatz wird durch ein Multi-Master-Konzept realisiert.</p>
PROFIBUS-DP-Adresse	Jedem PROFIBUS-DP-Teilnehmer wird eine eindeutige PROFIBUS-DP-Adresse zugeordnet, über die er vom Master angesprochen werden kann.
PROFIBUS-DP-Master	Der PROFIBUS-DP-Master regelt als zentraler Busteilnehmer den Zugriff aller PROFIBUS-DP-Slaves auf den PROFIBUS.
PROFIBUS-DP-Slave	PROFIBUS-DP-Slaves werden vom PROFIBUS-DP-Master angesprochen und tauschen, auf dessen Anforderung hin, Daten mit ihm aus.
Reaktionszeit	In einem Bussystem das Zeitintervall zwischen dem Absenden eines Leseauftrags und dem Erhalt einer Antwort. Innerhalb eines Eingabemoduls das Zeitintervall von der Signaländerung am Eingang des Moduls bis zur Ausgabe derselben an das Bussystem.
Repeater	Verstärker für die über einen Bus übertragenen Signale.
RS 485	Serielle Schnittstelle nach EIA-Norm zur schnellen Datenübertragung durch mehrere Sender.
Schirm	Bezeichnung für die leitfähige Hülle von Leitungen, Gehäusen und Schränken.
Schirmung	Gesamtheit der Maßnahmen und Betriebsmittel, die zur Verbindung von Anlagenteilen mit dem Schirm dienen.
Schutzleiter	Ein für den Schutz gegen gefährliche Körperströme notwendiger Leiter, dargestellt durch das Kürzel PE (Abkürzung für englisch "Protective Earth").
seriell	Bezeichnung für eine Art der Informationsübertragung, bei der die Daten nacheinander – Bit für Bit – über eine Leitung übertragen werden.

Slave	Station oder Teilnehmer in einem Bussystem, die/der dem Master untergeordnet ist/sind.
SPS	Abkürzung für Speicherprogrammierbare Steuerung.
Station	Funktionseinheit oder Baugruppe, bestehend aus mehreren Elementen.
Strahlungskopplung	Eine Strahlungskopplung tritt auf, wenn eine elektromagnetische Welle auf eine Leiterstruktur trifft. Durch das Auftreffen der Welle werden Ströme und Spannungen induziert. Typische Störquellen sind z. B. Funkenstrecken (Zündkerzen, Kollektoren von Elektromotoren) und Sender (z. B. Funkgeräte), die nahe bei der entsprechenden Leiterstruktur betrieben werden.
SUB-D-Stecker	9-poliger Stecker zum Anschluss des Feldbusses.
Topologie	Geometrischer Aufbau eines Netzes bzw. Anordnung der Schaltungen.
UART	Abkürzung für englisch "Universal Asynchronous Receiver/Transmitter", deutsch universeller asynchroner Empfänger/Sender. Ein „UART“ ist ein Logikschaltkreis, der zur Umwandlung einer asynchronen seriellen Datenfolge in eine bitparallele Datenfolge oder umgekehrt eingesetzt wird.
unidirektional	In einer Richtung arbeitend.

Stichwortverzeichnis

A	Abbilddaten	
	Allgemeine Informationen	64
	Übersicht easy600	64
	Übersicht easy600 (DPV1)	260
	Übersicht easy700	109
	Übersicht easy700 (DPV1)	300
	Übersicht easy800/MFD	160
	Übersicht easy800/MFD (DPV1)	364
	Abschlusswiderstände	24
	Adressbereich	30
	Analog-Ausgang	
	easy800/MFD (DPV1), lesen	367
	easy800/MFD, lesen und schreiben	170
	Analog-Eingänge	
	easy600 (DPV1), lesen	267
	easy600, lesen	81
	easy700, lesen	115
	easy700/800/MFD (DPV1), lesen	306
	easy800/MFD, lesen	164
	Analogwertvergleich	
	easy600 (DPV1), lesen	272
	easy600 (DPV1), schreiben	289
	easy600, lesen	69
	easy600, schreiben	73
	easy700 (DPV1), lesen	312
	easy700 (DPV1), lesen und schreiben	343
	easy700, lesen	110
	easy700, lesen und schreiben	132
	easy800/MFD (DPV1), lesen und schreiben	404
	easy800/MFD, lesen und schreiben	184
	Anschlussbelegung PROFIBUS-DP	23
	Arithmetikbaustein	
	easy800/MFD (DPV1), lesen und schreiben	406
	easy800/MFD, lesen und schreiben	186

Ausgänge von easyLink	
easy600 (DPV1), lesen	279
easy700, lesen	125
easy800/MFD, lesen	173
Ausgänge, easyNet-Teilnehmer	
easy800/MFD (DPV1), lesen	376
Ausgänge, lokale	
easy800/MFD, lesen und schreiben	170
Ausgänge, lokale und Netzwerkteilnehmer	
easy800/MFD, lesen und schreiben	168
Ausgangsdaten, Definition	47

B	Bedientasten	
	easy600	68
	easy600 (DPV1), lesen	275
	Bedingter Sprung	
	easy800/MFD	224
	easy800/MFD (DPV1), lesen und schreiben ..	434
	Betriebsart festlegen	
	easy600/700 (DPV1)	265
	easy600/700/800/MFD	51
	Betriebsmodus, ungültig	150
	Betriebsstundenzähler	
	easy700	138
	easy700 (DPV1), lesen	326
	easy700 (DPV1), lesen und schreiben	347
	easy800/MFD (DPV1), lesen und schreiben ..	443
	easy800/MFD, lesen und schreiben	234
	Betriebssystem-Voraussetzungen	19
	Bitfeld	106
	Bit-Merker	
	easy800/MFD (DPV1), lesen und schreiben ..	384
	Block Compare	
	easy800/MFD (DPV1), lesen und schreiben ..	408
	easy800/MFD, lesen und schreiben	188
	Block Transfer	
	easy800/MFD (DPV1), lesen und schreiben ..	410
	easy800/MFD, lesen und schreiben	190

Boolsche Verknüpfung	
easy800/MFD (DPV1), lesen und schreiben	.412
easy800/MFD, lesen und schreiben192
BUS-LED33
Busleitungslängen26
Byte-Merker	
easy800/MFD (DPV1), lesen und schreiben	.386

D	Daten-	
	ausgang50
	eingang48
	Datenaustausch	
	easy800/MFD (DPV1), lesen und schreiben	392, 394,396, 399
	Datenaustauschverfahren	
	easy60059
	easy700101
	easy800/MFD153
	Datenbaustein	
	easy800/MFD (DPV1), lesen und schreiben	.423
	Datenbaustein, lesen und schreiben (easy800/MFD)	207
	Datenmultiplexer	
	easy800/MFD (DPV1), lesen und schreiben	.439
	easy800/MFD, lesen und schreiben230
	Datum lesen/schreiben	
	easy60061
	easy600/700 (DPV1)294
	easy700104
	easy800156
	Diagnose	
	easy800/MFD (DPV1), lesen und schreiben	.426
	easy800/MFD, lesen212
	Diagnose, lokale	
	easy800/MFD, lesen166
	Diagnose, Net-Teilnehmer	
	easy800/MFD (DPV1), lesen370

Digital-Ausgänge	
easy600 (DPV1), lesen	269
easy600, lesen	65
easy700, lesen	124
easy700/800/MFD (DPV1), lesen	308
Digital-Eingänge	
easy600 (DPV1), lesen	267
easy600, lesen	81
easy700, lesen	113
easy700/800/MFD (DPV1), lesen	304
easy800/MFD, lesen	161
Doppelwort-Merker	
easy800/MFD (DPV1), lesen und schreiben	390
E	
Echtzeituhr	156
Eingänge	
easy800/MFD, lesen	164
Eingänge von easyLink	
easy600 (DPV1), lesen	277
easy700, lesen	125
easy800/MFD, lesen	173
Eingänge, easyNet-Teilnehmer	
easy800/MFD (DPV1), lesen	372
Eingänge, lokale	
easy800/MFD, lesen	161
Eingänge, Netzwerkteilnehmer	
easy800/MFD, lesen	163
Eingangsdaten auf easyLink schreiben	
easy600 (DPV1), lesen	277
Eingangsdaten, Definition	47
Eingangsverzögerung	48
Empfangsdaten, Netzwerkteilnehmer	
easy800/MFD	175, 216
easy800/MFD (DPV1)	380

F	Fehlercodes, über easyLink	
	easy700	150
	Fehlermeldungen, DPV1	464
	Frequenzzähler	
	easy800/MFD (DPV1), lesen und schreiben .	416
	easy800/MFD, lesen und schreiben	196
	Funktionsbaustein-Daten	
	auswählen, easy700 (DPV1)	337
	lesen, easy700 (DPV1)	339
	schreiben, easy700 (DPV1)	341
	Funktionsbaustein-Daten lesen/schreiben, Verfahren bei easy700 (DPV1)	336
	Funktionsbausteine, Übersicht	
	easy600	72
	easy600 (DPV1)	260
	easy700	131
	easy700 (DPV1)	301
	easy800/MFD	182
	easy800/MFD (DPV1)	365
G	GSD-Datei	43
H	Hardware-Voraussetzungen	19
	Hilfsrelais	
	easy600, lesen	65
I	Impulsausgabe	
	easy800/MFD (DPV1), lesen und schreiben .	444
	easy800/MFD, lesen und schreiben	236
	Inkrementalzähler	
	easy800/MFD (DPV1), lesen und schreiben .	419
	easy800/MFD, lesen und schreiben	200

J	Jahresschaltuhr	
	easy700	144
	easy700 (DPV1), lesen	316
	easy700 (DPV1), lesen und schreiben	352
	easy700, lesen	128
	easy800/MFD (DPV1), lesen und schreiben	432
	Jahreszeitschaltuhr	
	easy800/MFD, lesen	221
<hr/>		
L	Link-Ausgänge, easyNet-Teilnehmer	
	easy800/MFD (DPV1), lesen	378
	Link-Eingänge, easyNet-Teilnehmer	
	easy800/MFD (DPV1), lesen	374
	Lokale Ausgänge	
	easy700, lesen	124
	Lokale Eingänge	
	easy700, lesen	113
<hr/>		
M	Masterreset	
	easy700	129
	easy700 (DPV1), lesen	318
	easy800/MFD (DPV1), lesen und schreiben	437
	Merker	
	easy600 (DPV1), lesen	269
	easy700 (DPV1), lesen	328, 330
	easy700 (DPV1), schreiben	332, 334
	easy800/MFD, lesen und schreiben	177
	Minute	
	easy600	63
	Modul	
	Ausgaenge 1 Byte	54
	Ausgaenge 3 Byte	50
	Eingaenge 1 Byte	50
	Eingaenge 3 Byte	48

N	Netzdaten	
	easy800/MFD (DPV1), lesen und schreiben	428, 446
	easy800/MFD, lesen216, 239
<hr/>		
P	PID-Regler	
	easy800/MFD (DPV1), lesen und schreiben	.424
	easy800/MFD, lesen und schreiben209
	Potentialtrennungen25
	POW-LED33
	P-Tasten	
	easy600 (DPV1), lesen275
	easy600, lesen68
	easy700, lesen122
	easy700/800/MFD (DPV1), lesen310
	P-Tasten, lokale	
	easy800/MFD, lesen171
	Pulsweitenmodulation	
	easy800/MFD (DPV1), lesen und schreiben	.447
	easy800/MFD, lesen und schreiben241
<hr/>		
R	Reaktionszeiten (Basisgerät)34
	Receive-Data, Netzwerkteilnehmer	
	easy800/MFD175
	easy800/MFD (DPV1), lesen380
	Rücksetzen, easy/MFD-Ein-/Ausgänge50
<hr/>		
S	Schaltregel106
	Schieberegister	
	easy800/MFD (DPV1), lesen und schreiben	.450
	easy800/MFD, lesen und schreiben246
	Schneller Zähler	
	easy800/MFD (DPV1), lesen und schreiben	.417
	easy800/MFD, lesen und schreiben198
	Schwellwertschalter	
	easy700132
	easy700 (DPV1), lesen und schreiben343

Schwellwertvergleich	
easy700	
lesen	110
Send-Data, easyNet-Teilnehmer	
easy800/MFD (DPV1), lesen	382
Send-Data, Netzwerkteilnehmer	
easy800/MFD	175
Sendedaten, Netzwerkteilnehmer	
easy800/MFD	175, 239
easy800/MFD (DPV1), lesen	382
Serielle Ausgabe	
easy800/MFD (DPV1), lesen und schreiben ..	449
Setzen, easy/MFD-Ein-/Ausgänge	50
Signalglättungsfilter	
easy800/MFD (DPV1), lesen und schreiben ..	427
easy800/MFD, lesen und schreiben	214
Sollzykluszeit	
easy800/MFD (DPV1), lesen und schreiben ..	452
easy800/MFD, lesen und schreiben	248
Sommerzeit	
easy600	61
easy600 (DPV1)	297
easy700	105
easy700 (DPV1)	356
easy800/MFD	157
easy800/MFD (DPV1), lesen und schreiben ..	458
Statusanzeige	
easy/MFD-Eingänge	52
easy-Ausgänge S1 bis S8	49
Steuerbefehle	
easy600	59
easy700	101
easy800/MFD	153
Stunde	
easy600	62

T	Tabellenfunktion	
	easy800/MFD (DPV1), lesen und schreiben	.455
	easy800/MFD, lesen und schreiben253
	Teilnehmeradresse einstellen30
	Telegramm, ungültig150
	Textanzeige	
	easy600, lesen65
	Textausgabe	
	easy800/MFD (DPV1), lesen und schreiben	.421
	Textausgabe-Baustein	
	easy800/MFD, lesen und schreiben204
	Textbausteine	
	easy700 (DPV1), lesen320
	easy700, lesen112
	Textmerker	
	easy600 (DPV1), lesen269
	Toggelbyte	
	easy60060
	easy700102
	easy800/MFD154
U	Übertragungsraten26
	Uhr synchronisieren	
	easy800/MFD (DPV1), lesen und schreiben	.448
	easy800/MFD, lesen243
	Uhrzeit lesen/schreiben	
	easy60061
	easy600/700 (DPV1)294
	easy700104
	easy800156
	ungültiger Betriebsmodus150
	ungültiges Telegramm150

V	Vergleicher	
	easy800/MFD (DPV1), lesen und schreiben .	420
	easy800/MFD, lesen und schreiben	202
	Vergleichswerte	
	easy600	73
	Versionsgeschichte, easy800	155
	Versorgungsspannung	22

W	Wertbegrenzung	
	easy800/MFD (DPV1), lesen und schreiben .	457
	easy800/MFD, lesen und schreiben	255
	Wertskalierung	
	easy800/MFD (DPV1), lesen und schreiben .	435
	easy800/MFD, lesen und schreiben	226
	Winterzeit	
	easy600	61
	easy600 (DPV1)	297
	easy700	106
	easy700 (DPV1)	356
	easy800/MFD	157
	easy800/MFD (DPV1), lesen und schreiben .	458
	Wochenschaltuhr	
	easy600 (DPV1), lesen	272
	easy600 (DPV1), lesen und schreiben	291
	easy700	130, 147
	easy700 (DPV1), lesen	314
	easy700 (DPV1), lesen und schreiben	354
	easy800/MFD (DPV1), lesen und schreiben .	430
	easy800/MFD, lesen	218
	Wochentag	
	easy600	62
	Wort-Merker	
	easy800/MFD (DPV1), lesen und schreiben .	388

Z	Zahlenwandler	
	easy800/MFD (DPV1), lesen und schreiben	.441
	easy800/MFD, lesen und schreiben232
	Zähler	
	easy600 (DPV1), lesen272
	easy600 (DPV1), lesen und schreiben286
	easy700 (DPV1), lesen324
	easy700 (DPV1), lesen und schreiben345
	easy700, lesen111
	easy800/MFD (DPV1), lesen und schreiben	.414
	easy800/MFD, lesen und schreiben194
	Zählerrelais	
	easy600 (Istwert lesen)76
	easy600 (Sollwert schreiben)78
	easy600, lesen69
	easy700135
	Zeitglieder	
	easy700 (DPV1), lesen322
	easy700, lesen127
	Zeitrelais	
	easy600 (DPV1), lesen272
	easy600 (DPV1), lesen und schreiben281
	easy600 (Istwert lesen)84
	easy600 (Sollwert schreiben)87
	easy600, lesen69
	easy700140
	easy700 (DPV1), lesen und schreiben349
	easy800/MFD (DPV1), lesen und schreiben	.453
	easy800/MFD, lesen und schreiben250
	Zeitschaltuhr	
	easy600, lesen69, 94
	easy600, schreiben98
	Zeitumstellung	
	easy800157
	Zykluszeit34