

CPU-CPU Kommunikation mit SIMATIC Controllern

SIMATIC S7

Kompendium • September 2010

Applikationen & Tools

Answers for industry.

SIEMENS

Gewährleistung, Haftung und Support

Hinweis

Die Applikationsbeispiele sind unverbindlich und erheben keinen Anspruch auf Vollständigkeit hinsichtlich Konfiguration und Ausstattung sowie jeglicher Eventualitäten. Die Applikationsbeispiele stellen keine kundenspezifische Lösungen dar, sondern sollen lediglich Hilfestellung bieten bei typischen Aufgabenstellungen. Sie sind für den sachgemäßen Betrieb der beschriebenen Produkte selbst verantwortlich. Diese Applikationsbeispiele entheben Sie nicht der Verpflichtung zu sicherem Umgang bei Anwendung, Installation, Betrieb und Wartung. Durch Nutzung dieser Applikationsbeispiele erkennen Sie an, dass wir über die beschriebene Haftungsregelung hinaus nicht für etwaige Schäden haftbar gemacht werden können. Wir behalten uns das Recht vor, Änderungen an diesen Applikationsbeispielen jederzeit ohne Ankündigung durchzuführen. Bei Abweichungen zwischen den Vorschlägen in diesen Applikationsbeispiel und anderen Siemens Publikationen, wie z.B. Katalogen, hat der Inhalt der anderen Dokumentation Vorrang.

Für die in diesem Dokument enthaltenen Informationen übernehmen wir keine Gewähr. Unsere Haftung, gleich aus welchem Rechtsgrund, für durch die Verwendung der in diesem Applikationsbeispiel beschriebenen Beispiele, Hinweise, Programme, Projektierungs- und Leistungsdaten usw. verursachte Schäden ist ausgeschlossen, soweit nicht z.B. nach dem Produkthaftungsgesetz in Fällen des Vorsatzes, der groben Fahrlässigkeit, wegen der Verletzung des Lebens, des Körpers oder der Gesundheit, wegen einer Übernahme der Garantie für die Beschaffenheit einer Sache, wegen des arglistigen Verschweigens eines Mangels oder wegen Verletzung wesentlicher Vertragspflichten zwingend gehaftet wird. Der Schadensersatz wegen Verletzung wesentlicher Vertragspflichten ist jedoch auf den vertragstypischen, vorhersehbaren Schaden begrenzt, soweit nicht Vorsatz oder grobe Fahrlässigkeit vorliegt oder wegen der Verletzung des Lebens, des Körpers oder der Gesundheit zwingend gehaftet wird. Eine Änderung der Beweislast zu Ihrem Nachteil ist hiermit nicht verbunden. Weitergabe oder Vervielfältigung dieser Applikationsbeispiele oder Auszüge daraus sind nicht gestattet, soweit nicht ausdrücklich von Siemens Industry Sector zugestanden.

Bei Fragen zu diesem Beitrag wenden Sie sich bitte über folgende E-Mail-Adresse an uns:

online-support.automation@siemens.com

Industry Automation und Drives Technologies Service & Support Portal

Dieser Beitrag stammt aus dem Internet Serviceportal der Siemens AG, Industry Automation und Drives Technologies. Durch den folgenden Link gelangen Sie direkt zur Downloadseite dieses Dokuments.

<http://support.automation.siemens.com/WW/view/de/20982954>

Inhaltsverzeichnis

Gewährleistung, Haftung und Support	2
1 Vorbemerkungen zum Dokument	15
1.1 Gegenstand des Dokumentes.....	15
1.2 Veranlassung und Zielsetzung des Dokumentes.....	17
1.3 Eigenschaften und Nutzen des Dokumentes	18
1.4 Gültigkeitsbereich des Dokumentes.....	18
1.5 Einordnung des Dokumentes.....	19
2 Struktur des Dokumentes	20
2.1 Teil 1: Einführung	20
2.2 Teil 2: Auswahlhilfe	20
2.3 Teil 3: Kommunikationsarten.....	21
2.4 Teil 4: Kommunikation mit fremden Controllern ohne Verwendung eines offenen Standards	21
2.5 Teil 5: Anhang	21
3 Anwendung des Dokumentes	22
3.1 Hinweise zum Umgang mit dem Dokument.....	22
3.2 Beispiel zur Anwendung des Dokumentes.....	23
3.2.1 Aufgabenstellung.....	23
3.2.2 Übersicht Schnittstellen und Kommunikationsarten.....	24
3.2.3 Kombinationen Controller 1 / Controller 2.....	26
3.2.4 Kommunikationsarten.....	28
TEIL 1: Einführung	30
TEIL 1: Gliederung und Inhalt	31
4 Modelle zur CPU-CPU Kommunikation	32
4.1 Definition Controller.....	32
4.2 Definition CPU-CPU Kommunikation	32
4.3 CPUs in unterschiedlichen zentralen Stationen	33
4.4 CPUs in zentraler und dezentraler Station.....	34
4.5 CPUs innerhalb einer zentralen Station	35
5 Verbindungen bei SIMATIC	36
5.1 Verbindungen	36
5.2 Protokolle	37
5.3 Verbindungsressourcen	38
5.4 Einrichten von Verbindungen	39
5.5 Projektierte Verbindung.....	39
5.6 Nicht projektierte Verbindung.....	40
5.6.1 Automatisches Einrichten einer Verbindung	41
5.6.2 Programmiertes Einrichten einer Verbindung	41
6 Datenkonsistenz bei SIMATIC	42
6.1 Definitionen	42
6.2 Systembedingte Datenkonsistenz.....	43
6.2.1 S7-300.....	43
6.2.2 S7-400.....	43
6.2.3 S7-1200.....	43
6.3 Weitergehende Datenkonsistenz	44
6.3.1 Zusatzmaßnahmen	44
6.3.2 Fallunterscheidung.....	44

7	SIMATIC Controller	45
8	Medien zur SIMATIC Kommunikation	47
9	PROFINET/Industrial Ethernet (PN/IE)	48
9.1	Vorbemerkung.....	48
9.2	Ethernet.....	48
9.3	Industrial Ethernet (IE).....	50
9.4	PROFINET (PN).....	51
10	PROFIBUS (PB)	54
11	MPI	55
12	SIMATIC Rückwandbus	56
13	Serielle Schnittstelle (PtP)	57
14	Gegenüberstellung der Medien	58
15	Schnittstellen der SIMATIC Familien	59
16	Informationen Teil 1	60
	TEIL 2: Auswahlhilfe	61
	TEIL 2: Gliederung und Inhalt	62
17	Vorbemerkungen	63
17.1	Übersicht aller Kombinationen.....	63
17.1.1	Begriffe.....	63
17.1.2	Symmetrie der Kombinationen.....	64
17.1.3	Umsetzung im Dokument.....	65
17.2	Tabelle Schnittstellen.....	67
17.2.1	Zweck der Tabelle.....	67
17.2.2	Aufbau der Tabelle.....	67
17.2.3	Abkürzungen und Indizes.....	69
17.3	Tabelle Kombinationen.....	70
17.3.1	Zweck der Tabelle.....	70
17.3.2	Aufbau der Tabelle.....	70
17.3.3	Reales Beispiel.....	73
17.3.4	Abkürzungen und Indizes.....	73
17.4	Tabelle Kommunikationsarten-Kompakt.....	74
17.4.1	Zweck der Tabelle.....	74
17.4.2	Aufbau der Tabelle.....	74
17.5	Übersicht der Abkürzungen und Indizes.....	75
17.5.1	Abkürzungen.....	75
17.5.2	Übersicht der Indizes.....	76
18	***** Sprungverteiler *****	77
18.1	Netz PN/IE.....	77
18.2	Netz PB.....	78
18.3	Netz MPI.....	79
18.4	SIMATIC Rückwandbus.....	80
18.5	Serielle Schnittstellen.....	81
19	Auswahlhilfe: PROFINET/Industrial Ethernet (PN/IE)	82
19.1	PN/IE: Inhalt des Kapitels.....	82
19.2	PN/IE: Schnittstellen und Kommunikationsarten.....	83
19.2.1	ET 200 CPU an PN/IE.....	83
19.2.2	S7-300 an PN/IE.....	84

19.2.3	S7-400 an PN/IE	85
19.2.4	S7-1200 an PN/IE	86
19.2.5	S7-mEC an PN/IE	87
19.2.6	Box PC an PN/IE	88
19.2.7	Panel PC an PN/IE	90
19.2.8	WinAC MP an PN/IE	92
19.2.9	WinAC RTX an PN/IE	93
19.3	PN/IE: Controller 1 = ET 200 CPU	95
19.3.1	ET 200 CPU / ET 200 CPU	95
19.3.2	ET 200 CPU / S7-300	96
19.3.3	ET 200 CPU / S7-400	97
19.3.4	ET 200 CPU / S7-1200	98
19.3.5	ET 200 CPU / S7-mEC	99
19.3.6	ET 200 CPU / Box PC	100
19.3.7	ET 200 CPU / Panel PC	101
19.3.8	ET 200 CPU / WinAC MP	102
19.3.9	ET 200 CPU / WinAC RTX	103
19.4	PN/IE: Controller 1 = S7-300	104
19.4.1	(S7-300 / ET 200 CPU)	104
19.4.2	S7-300 / S7-300	105
19.4.3	S7-300 / S7-400	106
19.4.4	S7-300 / S7-1200	107
19.4.5	S7-300 / S7-mEC	108
19.4.6	S7-300 / Box PC	109
19.4.7	S7-300 / Panel PC	110
19.4.8	S7-300 / WinAC MP	111
19.4.9	S7-300 / WinAC RTX	112
19.5	PN/IE: Controller 1 = S7-400	113
19.5.1	(S7-400 / ET 200 CPU)	113
19.5.2	(S7-400 / S7-300)	113
19.5.3	S7-400 / S7-400	114
19.5.4	S7-400 / S7-1200	115
19.5.5	S7-400 / S7-mEC	116
19.5.6	S7-400 / Box PC	117
19.5.7	S7-400 / Panel PC	118
19.5.8	S7-400 / WinAC MP	119
19.5.9	S7-400 / WinAC RTX	120
19.6	PN/IE: Controller 1 = S7-1200	121
19.6.1	(S7-1200 / ET 200 CPU)	121
19.6.2	(S7-1200 / S7-300)	121
19.6.3	(S7-1200 / S7-400)	121
19.6.4	S7-1200 / S7-1200	122
19.6.5	S7-1200 / S7-mEC	123
19.6.6	S7-1200 / Box PC	124
19.6.7	S7-1200 / Panel PC	125
19.6.8	S7-1200 / WinAC MP	126
19.6.9	S7-1200 / WinAC RTX	127
19.7	PN/IE: Controller 1 = S7-mEC	128
19.7.1	(S7-mEC / ET 200 CPU)	128
19.7.2	(S7-mEC / S7-300)	128
19.7.3	(S7-mEC / S7-400)	128
19.7.4	(S7-mEC / S7-1200)	128
19.7.5	S7-mEC / S7-mEC	129
19.7.6	S7-mEC / Box PC	130
19.7.7	S7-mEC / Panel PC	131
19.7.8	S7-mEC / WinAC MP	132
19.7.9	S7-mEC / WinAC RTX	133

19.8	PN/IE: Controller 1 = Box PC	134
19.8.1	(Box PC / ET 200 CPU).....	134
19.8.2	(Box PC / S7-300)	134
19.8.3	(Box PC / S7-400)	134
19.8.4	(Box PC / S7-1200)	134
19.8.5	(Box PC / S7-mEC)	134
19.8.6	Box PC / Box PC	135
19.8.7	Box PC / Panel PC	136
19.8.8	Box PC / WinAC MP.....	137
19.8.9	Box PC / WinAC RTX.....	138
19.9	PN/IE: Controller 1 = Panel PC	139
19.9.1	(Panel PC / ET 200 CPU)	139
19.9.2	(Panel PC / S7-300)	139
19.9.3	(Panel PC / S7-400)	139
19.9.4	(Panel PC / S7-1200)	139
19.9.5	(Panel PC / S7-mEC)	139
19.9.6	(Panel PC / Box PC).....	139
19.9.7	Panel PC / Panel PC	140
19.9.8	Panel PC / WinAC MP.....	141
19.9.9	Panel PC / WinAC RTX.....	142
19.10	PN/IE: Controller 1 = WinAC MP.....	143
19.10.1	(WinAC MP / ET 200 CPU)	143
19.10.2	(WinAC MP / S7-300).....	143
19.10.3	(WinAC MP / S7-400).....	143
19.10.4	(WinAC MP / S7-1200).....	143
19.10.5	(WinAC MP / S7-mEC).....	143
19.10.6	(WinAC MP / Box PC)	143
19.10.7	(WinAC MP / Panel PC)	143
19.10.8	WinAC MP / WinAC MP	144
19.10.9	WinAC MP / WinAC RTX	145
19.11	PN/IE: Controller 1 = WinAC RTX.....	146
19.11.1	(WinAC RTX / ET 200 CPU)	146
19.11.2	(WinAC RTX / S7-300).....	146
19.11.3	(WinAC RTX / S7-400).....	146
19.11.4	(WinAC RTX / S7-1200).....	146
19.11.5	(WinAC RTX / S7-mEC).....	146
19.11.6	(WinAC RTX / Box PC)	146
19.11.7	(WinAC RTX / Panel PC)	146
19.11.8	(WinAC RTX / WinAC MP).....	146
19.11.9	WinAC RTX / WinAC RTX.....	147
19.12	PN/IE: Übersicht Kommunikationsarten.....	148
20	Auswahlhilfe: PROFIBUS (PB)	150
20.1	PB: Inhalt des Kapitels	150
20.2	PB: Schnittstellen und Kommunikationsarten	151
20.2.1	ET 200 CPU an PB	151
20.2.2	S7-300 an PB	152
20.2.3	S7-400 an PB	153
20.2.4	S7-1200 an PB	154
20.2.5	S7-mEC an PB	155
20.2.6	Box PC an PB	156
20.2.7	Panel PC an PB	157
20.2.8	WinAC MP an PB	158
20.2.9	WinAC RTX an PB	159
20.3	PB: Controller 1 = ET 200 CPU.....	160
20.3.1	ET 200 CPU / ET 200 CPU	160
20.3.2	ET 200 CPU / S7-300.....	161

20.3.3	ET 200 CPU / S7-400.....	162
20.3.4	ET 200 CPU / S7-mEC.....	163
20.3.5	ET 200 CPU / Box PC.....	164
20.3.6	ET 200 CPU / Panel PC.....	165
20.3.7	ET 200 CPU / WinAC MP.....	166
20.3.8	ET 200 CPU / WinAC RTX.....	167
20.4	PB: Controller 1 = S7-300	168
20.4.1	(S7-300 / ET 200 CPU).....	168
20.4.2	S7-300 / S7-300	169
20.4.3	S7-300 / S7-400	170
20.4.4	S7-300 / S7-mEC	171
20.4.5	S7-300 / Box PC.....	172
20.4.6	S7-300 / Panel PC.....	173
20.4.7	S7-300 / WinAC MP	174
20.4.8	S7-300 / WinAC RTX	175
20.5	PB: Controller 1 = S7-400	176
20.5.1	(S7-400 / ET 200 CPU).....	176
20.5.2	(S7-400 / S7-300).....	176
20.5.3	S7-400 / S7-400	177
20.5.4	S7-400 / S7-mEC	178
20.5.5	S7-400 / Box PC.....	179
20.5.6	S7-400 / Panel PC.....	180
20.5.7	S7-400 / WinAC MP	181
20.5.8	S7-400 / WinAC RTX	182
20.6	PB: Controller 1 = S7-mEC	183
20.6.1	(S7-mEC / ET 200 CPU).....	183
20.6.2	(S7-mEC / S7-300).....	183
20.6.3	(S7-mEC / S7-400).....	183
20.6.4	S7-mEC / S7-mEC	184
20.6.5	S7-mEC / Box PC.....	185
20.6.6	S7-mEC / Panel PC.....	186
20.6.7	S7-mEC / WinAC MP	187
20.6.8	S7-mEC / WinAC RTX	188
20.7	PB: Controller 1 = Box PC.....	189
20.7.1	(Box PC / ET 200 CPU).....	189
20.7.2	(Box PC / S7-300)	189
20.7.3	(Box PC / S7-400)	189
20.7.4	(Box PC / S7-mEC)	189
20.7.5	Box PC / Box PC	190
20.7.6	Box PC / Panel PC	191
20.7.7	Box PC / WinAC MP.....	192
20.7.8	Box PC / WinAC RTX.....	193
20.8	PB: Controller 1 = Panel PC.....	194
20.8.1	(Panel PC / ET 200 CPU)	194
20.8.2	(Panel PC / S7-300)	194
20.8.3	(Panel PC / S7-400).....	194
20.8.4	(Panel PC / S7-mEC)	194
20.8.5	(Panel PC / Box PC).....	194
20.8.6	Panel PC / Panel PC	195
20.8.7	Panel PC / WinAC MP.....	196
20.8.8	Panel PC / WinAC RTX.....	197
20.9	PB: Controller 1 = WinAC MP	198
20.9.1	(WinAC MP / ET 200 CPU)	198
20.9.2	(WinAC MP / S7-300).....	198
20.9.3	(WinAC MP / S7-400).....	198
20.9.4	(WinAC MP / S7-mEC).....	198
20.9.5	(WinAC MP / Box PC)	198

20.9.6	(WinAC MP / Panel PC)	198
20.9.7	WinAC MP / WinAC MP	199
20.9.8	WinAC MP / WinAC RTX	200
20.10	PB: Controller 1 = WinAC RTX	201
20.10.1	(WinAC RTX / ET 200 CPU)	201
20.10.2	(WinAC RTX / S7-300)	201
20.10.3	(WinAC RTX / S7-400)	201
20.10.4	(WinAC RTX / S7-mEC)	201
20.10.5	(WinAC RTX / Box PC)	201
20.10.6	(WinAC RTX / Panel PC)	201
20.10.7	(WinAC RTX / WinAC MP)	201
20.10.8	WinAC RTX / WinAC RTX	202
20.11	PB: Übersicht Kommunikationsarten	203
21	Auswahlhilfe: MPI (MPI)	205
21.1	MPI: Inhalt des Kapitels	205
21.2	MPI: Schnittstellen und Kommunikationsarten	206
21.2.1	ET 200 CPU an MPI	206
21.2.2	S7-300 an MPI	207
21.2.3	S7-400 an MPI	208
21.3	MPI: Controller 1 = ET 200 CPU	209
21.3.1	ET 200 CPU / ET 200 CPU	209
21.3.2	ET 200 CPU / S7-300	210
21.3.3	ET 200 CPU / S7-400	211
21.4	MPI: Controller 1 = S7-300	212
21.4.1	(S7-300 / ET 200 CPU)	212
21.4.2	S7-300 / S7-300	212
21.4.3	S7-300 / S7-400	213
21.5	MPI: Controller 1 = S7-400	214
21.5.1	(S7-400 / ET 200 CPU)	214
21.5.2	(S7-400 / S7-300)	214
21.5.3	S7-400 / S7-400	214
21.6	MPI: Übersicht Kommunikationsarten	215
22	Auswahlhilfe: SIMATIC Rückwandbus	216
22.1	Inhalt des Kapitels	216
22.2	Schnittstellen und Kommunikationsarten	217
22.3	Controller 1 = S7-400 / Controller 2 = S7-400	217
22.4	Übersicht Kommunikationsarten	218
23	Auswahlhilfe: Serielle Schnittstelle (PtP)	219
23.1	Inhalt des Kapitels	219
23.2	ET 200 CPU an PtP	220
23.3	S7-300 an PtP	221
23.4	S7-400 an PtP	222
23.5	S7-1200 an PtP	223
23.6	S7-mEC an PtP	224
23.7	Box PC an PtP	225
23.8	Panel PC an PtP	225
23.9	WinAC MP an PtP	225
23.10	WinAC RTX an PtP	225
23.11	Dezentrale Station	226
24	Informationen Teil 2	227
24.1	Kommunikationsarten	227
24.2	Konkrete Anwendungsbeispiele	228
24.3	Leistungsdaten	228

TEIL 3: Kommunikationsarten	229
TEIL 3: Gliederung und Inhalt	230
25 Vorbemerkungen	231
25.1 Merkmale.....	231
25.2 Eigenschaften (Tabelle Kommunikationsarten-Detail).....	231
25.2.1 Zweck der Tabelle	231
25.2.2 Aufbau der Tabelle	232
25.2.3 Erläuterung der Kriterien	233
25.3 Anwendung	236
25.4 Überblick Anwenderschnittstellen	236
25.5 Anwenderschnittstellen	237
26 SIMATIC S7 spezifische Kommunikation	238
26.1 Merkmale.....	238
26.2 Übersicht	238
27 Globaldaten-Kommunikation	239
27.1 Merkmale.....	239
27.2 Eigenschaften.....	240
27.3 Anwendung	242
27.4 Überblick Anwenderschnittstellen	242
27.5 Anwenderschnittstelle GD_SND, GD_RCV	243
27.5.1 Beschreibung	243
27.5.2 Parameter für GD_SND	243
27.5.3 Parameter für GD_RCV	243
28 S7-Basiskommunikation	244
28.1 Merkmale.....	244
28.2 Eigenschaften.....	245
28.3 Anwendung	247
28.3.1 X-Bausteine.....	247
28.3.2 I-Bausteine	248
28.4 Überblick Anwenderschnittstellen	248
28.5 Anwenderschnittstelle X_SEND / X_RCV.....	249
28.5.1 Beschreibung	249
28.5.2 Parameter für X_SEND.....	249
28.5.3 Parameter für X_RCV	249
28.6 Anwenderschnittstelle X_PUT, X_GET.....	250
28.6.1 Beschreibung	250
28.6.2 Parameter für X_PUT	250
28.6.3 Parameter für X_GET.....	250
28.7 Anwenderschnittstelle I_PUT, I_GET.....	251
28.7.1 Beschreibung	251
28.7.2 Parameter für I_PUT	251
28.7.3 Parameter für I_GET	251
29 S7-Kommunikation	252
29.1 Merkmale.....	252
29.2 Eigenschaften.....	253
29.3 Anwendung	256
29.4 Überblick Anwenderschnittstellen	257
29.5 Anwenderschnittstelle: USEND / URCV	258
29.5.1 Beschreibung	258
29.5.2 Parameter für USEND.....	259
29.5.3 Parameter für URCV	259

29.6	Anwenderschnittstelle: BSEND / BRCV.....	260
29.6.1	Beschreibung	260
29.6.2	Parameter für BSEND.....	261
29.6.3	Parameter für BRCV	261
29.7	Anwenderschnittstelle PUT, GET.....	262
29.7.1	Beschreibung	262
29.7.2	Parameter für PUT	262
29.7.3	Parameter für GET	263
29.8	Anwenderschnittstelle: USEND_E / URCV_E	264
29.8.1	Beschreibung	264
29.8.2	Parameter für USEND_E	264
29.8.3	Parameter für URCV_E.....	264
29.9	Anwenderschnittstelle PUT_E, GET_E.....	265
29.9.1	Beschreibung	265
29.9.2	Parameter für PUT_E.....	265
29.9.3	Parameter für GET_E.....	265
30	PROFINET/Industrial Ethernet (PN/IE).....	266
30.1	Merkmale.....	266
30.2	Übersicht	266
31	PN/IE: Offene-Kommunikation mit Send/Receive-Bausteinen.....	267
31.1	Merkmale.....	267
31.2	Eigenschaften.....	269
31.3	Anwendung	272
31.4	Überblick Anwenderschnittstellen	273
31.5	Anwenderschnittstelle AG_xSEND, AG_xRECV	274
31.5.1	Beschreibung	274
31.5.2	Parameter für AG_SEND, AG_LSEND, AG_SSEND	275
31.5.3	Parameter für AG_RECV, AG_LRECV, AG_SSRECV.....	276
31.6	Anwenderschnittstelle FETCH, WRITE (Server)	277
32	PN/IE: Offene-Kommunikation mit T-Bausteinen.....	278
32.1	Merkmale.....	278
32.2	Eigenschaften.....	280
32.3	Anwendung	283
32.4	Überblick Anwenderschnittstellen	284
32.5	Anwenderschnittstelle TSEND / TRCV	286
32.5.1	Beschreibung	286
32.5.2	Parameter für TSEND	286
32.5.3	Parameter für TRCV.....	287
32.6	Anwenderschnittstelle TUSEND / TURCV	288
32.6.1	Beschreibung	288
32.6.2	Parameter für TUSEND.....	288
32.6.3	Parameter für TURCV	289
32.7	Anwenderschnittstelle Verbindungsbausteine	290
32.7.1	Beschreibung	290
32.7.2	Parameter für TCON	290
32.7.3	Parameter für TDISCON	290
32.8	Anwenderschnittstelle TSEND_C / TRCV_C.....	291
32.8.1	Beschreibung	291
32.8.2	Parameter für TSEND_C.....	291
32.8.3	Parameter für TRCV_C.....	292
32.9	Anwenderschnittstelle FETCH, WRITE (Server)	293
32.9.1	Beschreibung	293
32.9.2	Parameter für FW_TCP.....	293
32.9.3	Parameter für FW_IOT	293

33	PN/IE: CBA	294
33.1	Merkmale.....	294
33.2	Anwendung	296
33.3	Anwenderschnittstellen	296
34	PN/IE: PNIO	297
34.1	Merkmale.....	297
34.2	Eigenschaften.....	298
34.3	Anwendung	300
34.4	Überblick Anwenderschnittstellen	300
34.5	Anwenderschnittstelle PNIO_SEND, PNIO_RECV	301
34.5.1	Beschreibung	301
34.5.2	Parameter für PNIO_SEND	302
34.5.3	Parameter für PNIO_RECV	302
34.6	Anwenderschnittstelle DPRD_DAT, DPWR_DAT	303
34.6.1	Beschreibung	303
34.6.2	Parameter für DPRD_DAT	303
34.6.3	Parameter für DPWR_DAT	303
35	PROFIBUS (PB)	304
35.1	Merkmale.....	304
35.2	Übersicht	304
36	PB: Offene-Kommunikation mit Send/Receive-Bausteinen	305
36.1	Merkmale.....	305
36.2	Eigenschaften.....	306
36.3	Anwendung	308
36.4	Überblick Anwenderschnittstellen	308
36.5	Anwenderschnittstelle: AG_xSEND, AG_xRECV	309
36.5.1	Beschreibung	309
36.5.2	Parameter für AG_SEND, AG_LSEND.....	310
36.5.3	Parameter für AG_RECV, AG_LRECV.....	310
37	PB: FMS-Kommunikation	311
37.1	Merkmale.....	311
37.2	Eigenschaften.....	312
37.3	Anwendung	314
37.4	Überblick Anwenderschnittstellen	314
37.5	Anwenderschnittstelle: READ	315
37.5.1	Beschreibung	315
37.5.2	Parameter.....	315
37.6	Anwenderschnittstelle WRITE.....	316
37.6.1	Beschreibung	316
37.6.2	Parameter.....	316
37.7	Anwenderschnittstelle REPORT	317
37.7.1	Beschreibung	317
37.7.2	Parameter.....	317
38	PB: DP-Kommunikation	318
38.1	Merkmale.....	318
38.2	Eigenschaften.....	319
38.3	Anwendung	321
38.4	Überblick Anwenderschnittstellen	321
38.5	Anwenderschnittstelle DP_SEND, DP_RECV	322
38.5.1	Beschreibung	322
38.5.2	Parameter für DP_SEND	323

38.5.3	Parameter für DP_RECV	323
38.6	Anwenderschnittstelle DPRD_DAT, DPWR_DAT	324
38.6.1	Beschreibung	324
38.6.2	Parameter für DPRD_DAT	324
38.6.3	Parameter für DPWR_DAT	324
39	Serielle Schnittstelle.....	325
39.1	Merkmale	325
39.2	Übersicht	325
40	ASCII und 3964(R) und RK 512.....	326
40.1	Merkmale	326
40.1.1	Abgrenzung	326
40.1.2	Merkmale ASCII	327
40.1.3	Merkmale 3964(R).....	328
40.1.4	Merkmale RK 512.....	329
40.2	Anwendung	330
40.3	Eigenschaften.....	331
40.4	Überblick Anwenderschnittstellen ASCII und 3964(R).....	333
40.5	ASCII und 3964(R): Anwenderschnittstelle ET 200S.....	334
40.5.1	Beschreibung	334
40.5.2	Parameter für S_SEND.....	334
40.5.3	Parameter für S_RCV	335
40.6	ASCII / 3964(R): Anwenderschnittstelle S7-300 CPU	336
40.6.1	Beschreibung	336
40.6.2	Parameter für SEND_PTP	336
40.6.3	Parameter für RCV_PTP.....	337
40.7	ASCII / 3964(R): Anwenderschnittstelle CP 340.....	338
40.7.1	Beschreibung	338
40.7.2	Parameter für P_SEND.....	338
40.7.3	Parameter für P_RCV	339
40.8	ASCII / 3964(R): Anwenderschnittstelle CP 341.....	340
40.8.1	Beschreibung	340
40.8.2	Parameter für P_SND_RK	340
40.8.3	Parameter für P_RCV_RK	341
40.9	ASCII / 3964(R): Anwenderschnittstelle CP 440.....	342
40.9.1	Beschreibung	342
40.9.2	Parameter für SEND_440	342
40.9.3	Parameter für REC_440.....	343
40.10	ASCII / 3964(R): Anwenderschnittstelle CP 441.....	344
40.10.1	Beschreibung	344
40.10.2	Parameter für BSEND.....	345
40.10.3	Parameter für BRCV	345
40.11	Überblick Anwenderschnittstellen RK 512	346
40.12	RK 512: Anwenderschnittstelle S7-300 CPU	347
40.12.1	Beschreibung	347
40.12.2	Daten senden	347
40.12.3	Daten holen	347
40.12.4	Parameter für SEND_RK	348
40.12.5	Parameter für SERVE_RK	349
40.12.6	Parameter für FETCH_RK	350
40.12.7	Parameter für SERVE_RK	351
40.13	RK 512: Anwenderschnittstelle CP 341	352
40.13.1	Beschreibung	352
40.13.2	Daten senden	352
40.13.3	Daten holen	352
40.13.4	Parameter für P_SND_RK	353

40.13.5	Parameter für P_RCV_RK	354
40.13.6	Parameter für P_SND_RK	355
40.13.7	Parameter für P_RCV_RK	356
40.14	RK 512: Anwenderschnittstelle CP 441	357
40.14.1	Beschreibung	357
40.14.2	Daten senden	357
40.14.3	Daten senden	357
40.14.4	Parameter für BSEND	359
40.14.5	Parameter für BRCV	359
40.14.6	Parameter für BSEND	360
40.14.7	Parameter für PUT	360
40.14.8	Parameter für GET	361
41	Anwenderdefiniertes Protokoll	362
41.1	Merkmale	362
41.2	Anwendung	362
41.3	Anwenderschnittstelle S7-1200	362
41.3.1	Beschreibung	362
41.3.2	Parameter für SEND_PTP	363
41.3.3	Parameter für RCV_PTP	363
42	Informationen Teil 3	364
TEIL 4:	Kommunikation mit fremden Controllern ohne Verwendung eines offenen Standards	365
TEIL 4:	Gliederung und Inhalt	366
43	Vorbemerkungen	367
44	Modbus/TCP	368
44.1	Merkmale	368
44.2	Überblick Anwenderschnittstellen	368
44.2.1	Allgemeines	368
44.2.2	Anschluss über SIMATIC CPU	369
44.2.3	Anschluss über SIMATIC CP	369
44.2.4	Einrichten der TCP-Verbindung	369
45	Modbus Seriell (RTU Format)	370
45.1	Merkmale	370
45.2	Überblick Anwenderschnittstellen	371
45.2.1	Allgemeines	371
45.2.2	Anschluss über SIMATIC CP	371
45.2.3	Modbus Master	371
45.2.4	Modbus Slave	372
45.3	Modbus Master: Anwenderschnittstelle ET 200S	373
45.4	Modbus Master: Anwenderschnittstelle CP 341	373
45.4.1	Beschreibung	373
45.4.2	Parameter P_SND_RK	373
45.4.3	Parameter P_RCV_RK	374
45.5	Modbus Master: Anwenderschnittstelle CP 441-2	375
45.5.1	Beschreibung	375
45.5.2	Parameter BSEND	375
45.5.3	Parameter BRCV	376
45.6	Modbus Master: Anwenderschnittstelle CM 1241	377
45.6.1	Beschreibung	377
45.6.2	Parameter MB_MASTER	377
45.7	Modbus Slave: Anwenderschnittstelle ET 200S	378

45.7.1	Beschreibung	378
45.7.2	Parameter S_MODB	378
45.8	Modbus Slave: Anwenderschnittstelle CP 341	379
45.8.1	Beschreibung	379
45.8.2	Parameter FB80	379
45.9	Modbus Slave: Anwenderschnittstelle CP 441-2	380
45.9.1	Beschreibung	380
45.9.2	Parameter FB180	380
45.10	Modbus Slave: Anwenderschnittstelle CM 1241	381
45.10.1	Beschreibung	381
45.10.2	Parameter MB_SLAVE	381
46	Informationen Teil 4.....	382
	TEIL 5: Anhang.....	383
	TEIL 5: Gliederung und Inhalt.....	384
47	Literaturangaben	385
47.1	Informationen	385
47.2	FAQ	388
47.3	Applikationen aus Service & Support Portal	388
48	Begriffe	389
49	Abkürzungen.....	391
50	Hintergrundwissen	392
50.1	ISO/OSI-Referenzmodell.....	392
50.2	Kommunikationsmodelle	393
50.2.1	Client und Server.....	393
50.2.2	Master und Slave	394
50.2.3	Consumer und Provider	395
50.3	Quittierung.....	396
51	Betrachtete Komponenten.....	397
51.1	Programmierwerkzeuge	397
51.2	SIMATIC CPU	398
51.3	SIMATIC CP	400
52	Historie.....	401
52.1	Versionen	401
52.2	Wesentliche Änderungen	401

1 Vorbemerkungen zum Dokument

1.1 Gegenstand des Dokumentes

Kommunikationsaufgaben

Im Bereich der Automatisierungstechnik spielt die Kommunikation von Controllern eine zentrale Rolle. Controller führen unterschiedliche Kommunikationsaufgaben aus. Die folgende Tabelle zeigt diese Kommunikationsaufgaben.

Tabelle 1-1

Kommunikationsaufgabe	Kommunikationspartner	Kommunikation	Daten (Beispiele)	Netz (Beispiele)
Feld- und Prozess-Kommunikation	Controller dezentrale Peripherie (Aktoren, Sensoren)	innerhalb eines Netzes	Endschalter- positionen Tempera- turwerte	PROFINET/ Industrial Ethernet PROFIBUS
Daten-Kommunikation	Controller 1 Controller 2	innerhalb eines Netzes, oder über Netzwerkgrenzen hinweg	Sollwerte Rezepte	PROFINET/ Industrial Ethernet PROFIBUS
IT-Kommunikation	Controller PC	weltweit	E-Mail Datei	PROFINET/ Industrial Ethernet Internet

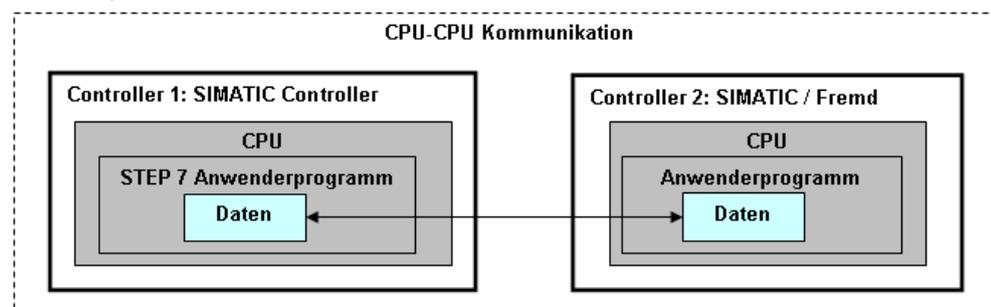
Gegenstand des vorliegenden Dokumentes ist die Daten-Kommunikation zwischen folgenden Kommunikationspartnern:

- SIMATIC Controller / SIMATIC Controller
- SIMATIC Controller / fremder Controller

CPU-CPU Kommunikation

Bei der Daten-Kommunikation werden Daten (Datenbausteine, Merker, ...) zwischen Controllern ausgetauscht. Diese Daten liegen in den Anwenderprogrammen der CPUs. Zur Verdeutlichung wird deswegen im vorliegenden Dokument für den Begriff „Daten-Kommunikation“ der treffendere Begriff „CPU-CPU Kommunikation“ verwendet. Das folgende Bild verdeutlicht dies.

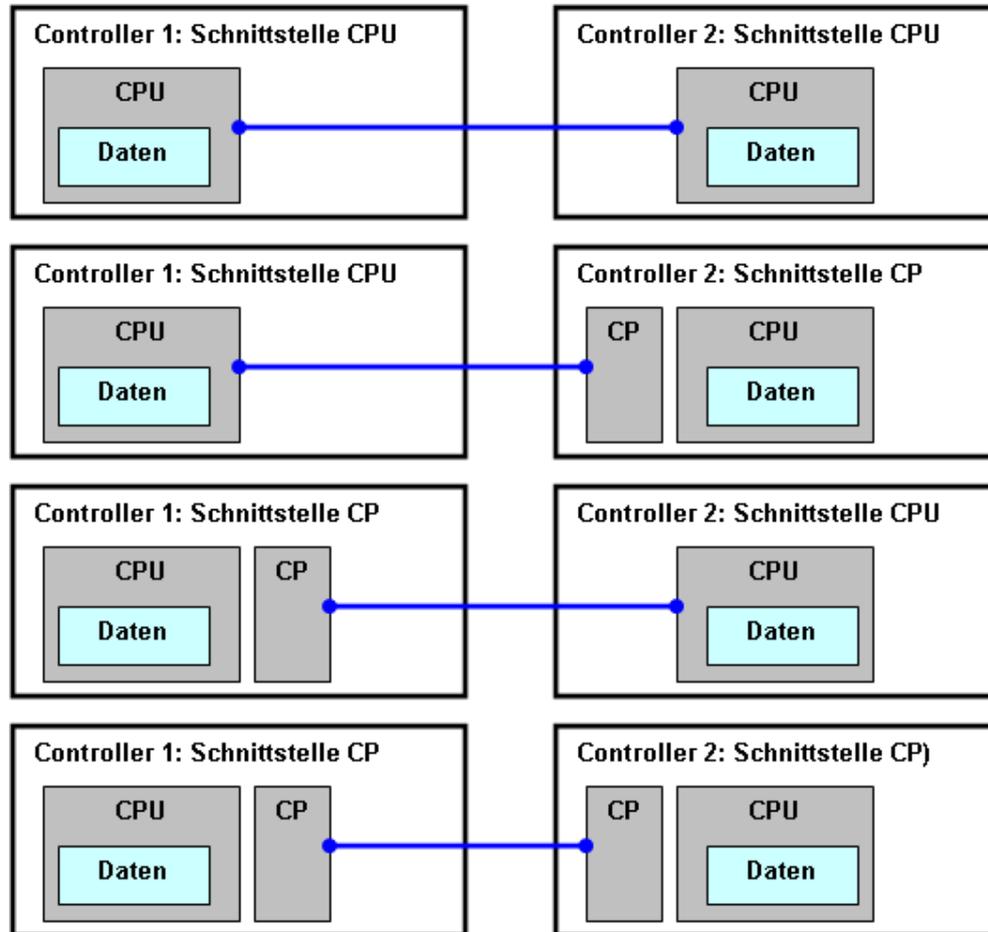
Abbildung 1-1



Kommunikationswege

Zur Kommunikation kann die integrierte Schnittstelle einer SIMATIC CPU, oder ein SIMATIC CP verwendet werden. Das folgende Bild zeigt die im Dokument betrachteten Varianten.

Abbildung 1-2



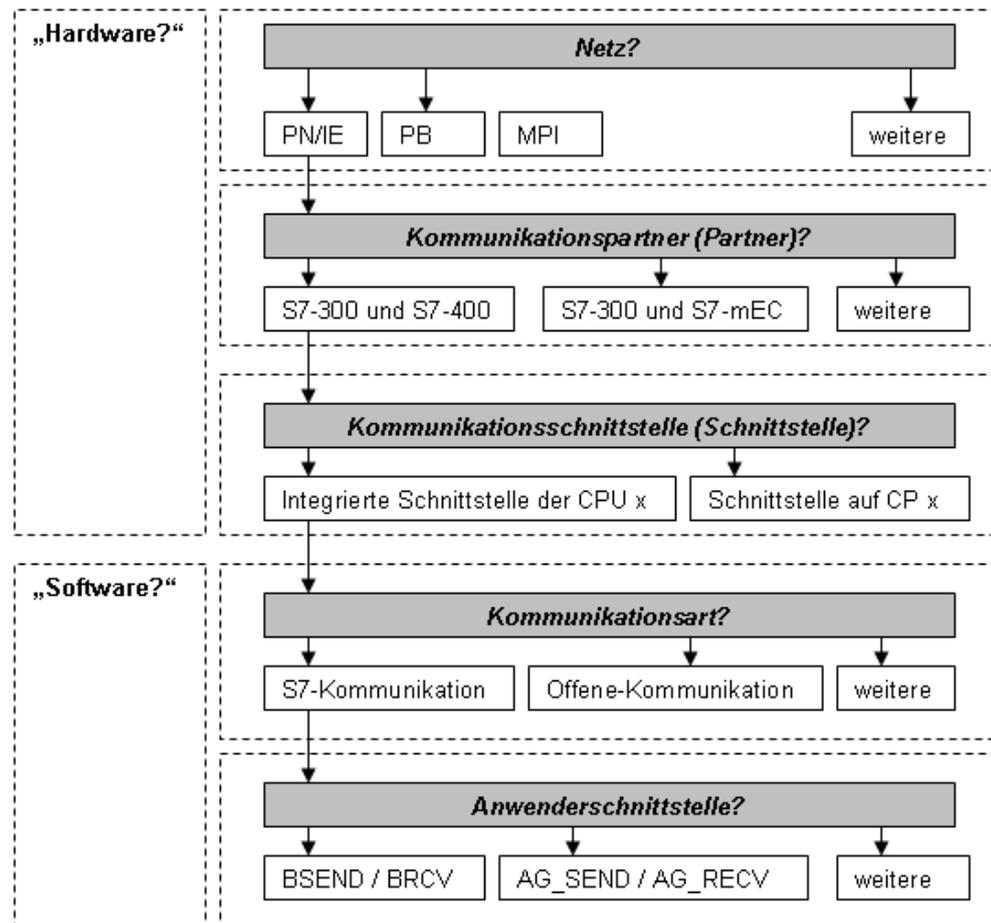
1.2 Veranlassung und Zielsetzung des Dokumentes

Veranlassung

Zur Realisierung einer CPU-CPU Kommunikation gibt es zahlreiche unterschiedliche Möglichkeiten. Für den Anwender stellen sich daher bei der Suche nach einer optimalen Lösung folgende Fragen:

- Welche Lösungen gibt es überhaupt?
- Wie unterscheiden sich die Lösungen?

Abbildung 1-3



Zielsetzung

Das Dokument hilft beim Finden einer optimalen Lösung für eine CPU-CPU Kommunikation zwischen zwei SIMATIC Controllern bzw. zwischen einem SIMATIC Controller und einem Controller eines Fremdherstellers („fremder Controller“).

Zusammengefasst beantwortet das Dokument die Frage:

Wer kann mit wem wie kommunizieren?

1.3 Eigenschaften und Nutzen des Dokumentes

Eigenschaften

Das Dokument hat folgende Eigenschaften:

- übersichtlicher und kompakter Aufbau
- stichpunktartiger Inhalt
- Es werden keine Details beschrieben, die auch in anderen Dokumenten zu finden sind. So wird zum Beispiel die exakte Arbeitsweise von Kommunikationsbausteinen nicht beschrieben (*1).

(*1): Details zu den Kommunikationsbausteinen sind zu finden in

- STEP 7 Online Hilfe
- Gerätehandbücher zu den S7-CPU's und S7-CP's
- System- und Standardfunktionen für SIMATIC S7-CPU's (/6/)
- Funktionen und Funktionsbausteine für SIMATIC NET S7-CP's (/13/)

Nutzen

Das Dokument bietet dem Leser folgenden Nutzen:

- Unterstützung bei Planung und Projektierung
- Schnelles Finden von Informationen (Nachschlagewerk)
- Vermittlung von Basiswissen
- Verweis auf weitere hilfreiche Informationen (Handbücher, Applikationsbeispiele, FAQs, ...)

1.4 Gültigkeitsbereich des Dokumentes

Alle Aussagen im Dokument beziehen sich ausschließlich auf die aktuellsten Komponenten der SIMATIC:

- Stand September 2010
- Programmierwerkzeug STEP 7 V5.5 (außer S7-1200)
- Programmierwerkzeug STEP 7 Basic V10.5 (für S7-1200)

Im Dokument werden folgende Themen nicht betrachtet:

- Feld- und Prozess-Kommunikation (Sensoren, ...)
- IT-Kommunikation (E-Mail, ...)
- Kommunikation zu Standard PC (OPC, ...)
- Kommunikation über Modem
- F-Kommunikation
- H-Kommunikation

1.5 Einordnung des Dokumentes

Zum Thema Kommunikation gibt es bei SIMATIC zahlreiche Dokumente. Die folgende Tabelle zeigt, wie das vorliegende Dokument einzuordnen ist.

Tabelle 1-2

Dokument	Zielsetzung des Dokumentes	Verweis
Vorliegendes Dokument: CPU-CPU Kommunikation mit SIMATIC Controllern	Auswahlhilfe zur CPU-CPU Kommunikation	---
Handbücher zu den Komponenten (S7-CPU, S7-CP, ...)	Technische Dokumentation der Komponenten. (Eigenschaften der Schnittstellen, ...)	/0/
Applikationsbeispiele zur Kommunikation	Lösungen zu konkreten Aufgabenstellungen (Dokumentation und STEP 7 Projekt)	Service & Support Portal
FAQs zur Kommunikation	Antworten auf häufig gestellte Fragen	
Systemhandbuch Kommunikation mit SIMATIC	Basiswissen zur industriellen Kommunikation	/3/
Katalog Produkte für Totally Integrated Automation und Micro Automation	Bestellunterlage für SIMATIC Controller	/4/
Katalog Industrielle Kommunikation	Bestellunterlage für SIMATIC Net Produkte (kurze Beschreibung der Netze, ...)	/5/

2 Struktur des Dokumentes

Das Dokument besteht aus mehreren Teilen (Teil 1 bis Teil 5).
Ziel und Inhalt der einzelnen Teile werden im Folgenden kurz erläutert.

2.1 Teil 1: Einführung

Ziel

Teil 1 dient der Einführung ins Thema CPU-CPU Kommunikation:

- Kompakter Einstieg in das Thema Kommunikation mit SIMATIC
- Erläuterung von Begriffen und Zusammenhängen, die für das Verständnis des Dokumentes wichtig sind.

Inhalt

Folgende Themen werden angesprochen:

- Funktionsmodelle zur CPU-CPU Kommunikation
- Verbindungen bei SIMATIC
- Datenkonsistenz bei SIMATIC
- Übersicht der SIMATIC Controller
- Medien zur SIMATIC Kommunikation
- Schnittstellen der SIMATIC Familien

Am Ende steht ein Kapitel mit Hinweisen auf weitere Informationen.

2.2 Teil 2: Auswahlhilfe

Ziel

Teil 2 ist der zentrale Teil des Dokumentes:

- Übersichtliche Darstellung aller Möglichkeiten für eine CPU-CPU Kommunikation mit SIMATIC Controllern

Inhalt

Pro Medium (PN/IE, PB, MPI, ...) wird beschrieben:

- Für jede SIMATIC Familie (ET 200 CPU, S7-300, ...):
 - Welche Schnittstellen (CPU, CP) gibt es?
 - Welche Kommunikationsarten (S7-Kommunikation, ...) gibt es?
- Wie können die SIMATIC Familien miteinander kommunizieren?
 - Über welche Schnittstellen?
 - Mit welchen Kommunikationsarten?
 - Welche Besonderheiten (Client, Server, ...) sind zu beachten?
- Gegenüberstellung aller zur Verfügung stehenden Kommunikationsarten

Am Ende steht ein Kapitel mit Hinweisen auf weitere Informationen.

2.3 Teil 3: Kommunikationsarten

Ziel

Teil 3 dient der Vertiefung:

- Detaillierte Informationen zu allen Kommunikationsarten

Inhalt

Pro Kommunikationsart (S7-Kommunikation, ...) wird beschrieben:

- Merkmale
- Eigenschaften (Tabelle mit einheitlichen Kriterien)
- Prinzipielle Anwendung (Projektierung, Programmierung)
- Anwenderschnittstelle (Kommunikationsbausteine)

Am Ende steht ein Kapitel mit Hinweisen auf weitere Informationen.

2.4 Teil 4: Kommunikation mit fremden Controllern ohne Verwendung eines offenen Standards

Teil 4 beschreibt Beispiele für die Kommunikation von SIMATIC Controllern mit Controllern von Fremdherstellern (fremde Controller) über offengelegte Protokolle.

Beispiel: Kommunikation zu fremden Controllern über Modbus/TCP

Am Ende steht ein Kapitel mit Hinweisen auf weitere Informationen.

2.5 Teil 5: Anhang

Inhalt von Teil 5:

- Literaturangaben
- Begriffe und Abkürzungen
- Hintergrundwissen zu ausgesuchten Themen
- Übersicht der im Dokument betrachteten Komponenten
- Historie des Dokumentes

3 Anwendung des Dokumentes

Das Dokument kann unterschiedlich genutzt werden:

- Lesen des Dokumentes direkt am PC (online)
- Lesen des ausgedruckten Dokumentes (offline)

In der folgenden Beschreibung wird dies berücksichtigt. In Klammern wird ergänzt, ob die beschriebene Aktion online oder offline möglich ist.

3.1 Hinweise zum Umgang mit dem Dokument

Navigation im Dokument

Da das Dokument sehr umfangreich ist, wurden Vorkehrungen getroffen, die den Umgang mit dem Dokument erleichtern.

Inhaltsverzeichnis

Über das ausführliche Inhaltsverzeichnis können gezielt Kapitel ausgewählt werden (online, offline).

Sprungverteiler

Am Anfang von Teil 2 der Dokumentation befindet sich das Kapitel Sprungverteiler. Dort existiert für jedes Medium eine Seite mit einer Sammlung von Querverweisen. Es sind dort alle wichtigen Kapitel zum betreffenden Medium aufgeführt. Querverweise sind im Dokument gekennzeichnet (grau hinterlegt, oder mit einem Rahmen versehen).

Ein Anklicken eines Querverweises (online) führt in das entsprechende Kapitel. Am Ende dieser Kapitel befindet sich ein Rücksprung, mit dem man schnell wieder zum Sprungverteiler zurückkehren kann (online). Ein Rücksprung wird durch einen blauen und unterstrichenen Text gekennzeichnet.

Beispiel: [Zurück zum Sprungverteiler PN/IE](#)

Literaturangaben

Literaturangaben werden im Text mit /x/ gekennzeichnet. Im Kapitel 47 befindet sich eine Sammlung von Links zu den betreffenden Quellen. Durch Anklicken eines Links (online) gelangt man direkt zur gewünschten Information.

Begriffe und Abkürzungen

Im Kapitel 48 werden wichtige Begriffe erläutert. Das Kapitel 49 enthält eine Beschreibung aller Abkürzungen.

Hintergrundwissen

Im Kapitel 50 werden wichtige Zusammenhänge erläutert.

3.2 Beispiel zur Anwendung des Dokumentes

Im Folgenden wird die Anwendung des Dokumentes anhand eines konkreten Beispiels gezeigt.

3.2.1 Aufgabenstellung

Bekannt

Zwei SIMATIC Controller sollen über das Netz PN/IE kommunizieren:

- Controller 1: aus Familie S7-300
- Controller 2: aus Familie S7-400

Gesucht

Gesucht sind Antworten auf die Fragen:

- Frage 1: Welche Schnittstellen und Kommunikationsarten stehen pro Familie zur Verfügung?
- Frage 2: Welche Komponenten können mit einander kommunizieren, und welche Kommunikationsarten sind möglich?
- Frage 3: Welche Eigenschaften haben die zur Verfügung stehenden Kommunikationsarten?
- Frage 4: Wie sehen die Anwenderschnittstellen (Kommunikationsbausteine) konkret aus?

Lösung

Für jedes Medium (PNIE, PB, MPI, ...) gibt es im Dokument einen sogenannten Sprungverteiler.

Der Sprungverteiler besteht aus den Abschnitten:

- Übersicht Schnittstellen und Kommunikationsarten
- Kombinationen Controller 1 / Controller 2
- Kommunikationsarten

Mit dem Sprungverteiler werden die obigen Fragen schnell beantwortet. Dies wird in den folgenden Kapiteln am Beispiel gezeigt.

3.2.2 Übersicht Schnittstellen und Kommunikationsarten

Die folgende Abbildung zeigt den Abschnitt „Übersicht Schnittstellen und Kommunikationsarten“ aus dem Sprungverteiler für PN/IE (Kapitel 18.1).

Abbildung 3-1

Familie		Kapitel
Modulare Controller	ET 200 CPU	19.2.1
	S7-300	19.2.2
	S7-400	19.2.3
	S7-1200	19.2.4
Embedded Controller	S7-mEC	19.2.5
	Box PC	19.2.6
	Panel PC	19.2.7
	WinAC MP	19.2.8
PC-based Controller	WinAC RTX	19.2.9

Vorgehensweise zur Beantwortung der Frage 1 (Welche Schnittstellen und Kommunikationsarten stehen pro Familie zur Verfügung?):

Das Kapitel zur gesuchten SIMATIC Familie anklicken (online) oder das entsprechende Kapitel aufschlagen (offline).

Ergebnis:

Im Kapitel ist eine Tabelle zu finden, in der alle Schnittstellen und Kommunikationsarten der entsprechenden Familie aufgeführt sind. Die folgende Abbildung zeigt einen Ausschnitt dieser Tabelle für die Familie S7-300 (Tabelle 19-2).

Abbildung 3-2

Controller an PN/IE: S7-300		Kommunikationsart	
		SIMATIC spezifisch	Offener Standard
		S7-Kommunikation	Offene-Kommunikation
CPU	alle mit Schnittstelle: PN (*2)	(1) (IoT) (14) (IoT)	(3) (IoT, TCP, UDP) (13) (IoT, TCP) (*1)
CP	343-1 Lean	PUT, GET Server (IoT)	(8) (IoT, TCP, UDP) (13) (IoT, TCP)
	343-1	(1) (IoT, ISO)	(8) (IoT, TCP, UDP, ISO) (13) (IoT, TCP, ISO)
	343-1 Advanced	(1) (IoT, ISO)	(8) (IoT, TCP, UDP, ISO) (13) (IoT, TCP, ISO)
	343-1 ERPC ^(*)	(1) (IoT)	(8) (IoT, TCP, UDP) (13) (TCP)

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- | | |
|--------------------------------------|-------------------------------------|
| (1) USEND/URCV, BSEND/BRCV, PUT, GET | (2) Lade/Transfer Befehle, DPRD_DAT |
| (3) TSEND/TRCV, TUSEND/TURCV | (6) PNIO_SEND, PNIO_RECV |
| (8) AG_SEND/AG_RECV | (13) Server für Fetch/Write |

3.2 Beispiel zur Anwendung des Dokumentes

So wird die Tabelle gelesen (Beispiel in der roten Umrandung):

- Mit dem CP 343-1 kann S7-300 am PN/IE betrieben werden.
- Eine mögliche Kommunikationsart ist S7-Kommunikation (Server und Client).
- Die unter dem Index (1) aufgeführten Kommunikationsbausteine können eingesetzt werden: USEND/URC, BSEND/BRCV, PUT, GET
- In Klammern stehen die zur Verfügung stehenden Protokolle:
ISO on TCP, ISO

Hinweis

An dieser Stelle soll nur das Prinzip der Tabelle gezeigt werden. Eine ausführliche Beschreibung des Aufbaus der Tabelle ist im Kapitel (17.2) zu finden.

3.2.3 Kombinationen Controller 1 / Controller 2

Die folgende Abbildung zeigt den Abschnitt „Kombinationen Controller 1 / Controller 2“ aus dem Sprungverteiler für PN/IE (Kapitel 18.1).

Abbildung 3-3

Controller 2		Controller 1								
		Modular					Embedded			PC-based
		ET 200 CPU	S7-300	S7-400	S7-1200	S7-mEC	Box PC	Panel PC	WinAC MP	WinAC RTX
Modular	ET 200 CPU	19.3.1	19.3.2	19.3.3	19.3.4	19.3.5	19.3.6	19.3.7	19.3.8	19.3.9
	S7-300	19.3.2	19.4.2	19.4.3	19.4.4	19.4.5	19.4.6	19.4.7	19.4.8	19.4.9
	S7-400	19.3.3	19.4.3	19.5.3	19.5.4	19.5.5	19.5.6	19.5.7	19.5.8	19.5.9
	S7-1200	19.3.4	19.4.4	19.5.4	19.6.4	19.6.5	19.6.6	19.6.7	19.6.8	19.6.9
Embedded	S7-mEC	19.3.5	19.4.5	19.5.5	19.6.5	19.7.5	19.7.6	19.7.7	19.7.8	19.7.9
	Box PC	19.3.6	19.4.6	19.5.6	19.6.6	19.7.6	19.8.6	19.8.7	19.8.8	19.8.9
	Panel PC	19.3.7	19.4.7	19.5.7	19.6.7	19.7.7	19.8.7	19.9.7	19.9.8	19.9.9
	WinAC MP	19.3.8	19.4.8	19.5.8	19.6.8	19.7.8	19.8.8	19.9.8	19.10.8	19.10.9
PC-based	WinAC RTX	19.3.9	19.4.9	19.5.9	19.6.9	19.7.9	19.8.9	19.9.9	19.10.9	19.11.9

Vorgehensweise zur Beantwortung der Frage 2 (Welche Komponenten können mit einander kommunizieren, und welche Kommunikationsarten sind möglich?):

Das Kapitel zur gesuchten Kombination zweier SIMATIC Familien anklicken (online) oder das entsprechende Kapitel aufschlagen (offline).

Ergebnis:

Im Kapitel ist eine Tabelle zu finden, in der alle Kombinationen der Schnittstellen (CPU, CP) beider Familien eingetragen sind. Für jede Kombination sind die möglichen Kommunikationsarten eingetragen. Die folgende Abbildung zeigt einen Ausschnitt dieser Tabelle für die Familien S7-300 und S7-400 (Tabelle 19-23).

Abbildung 3-4

Controller 2: S7-400			Controller 1: S7-300 an PN/IE							
			CPU				CP			
			alle mit Schnittstelle: PN				343-1 Lean			
			IOC, IOD				IOD			
			S7	OC	PN (*1)		S7	OC	PN (*1)	
					PNIO	CBA			PNIO	CBA
CPU	alle mit Schnittstelle: PN	IOC, IOD	(1)	(3)	(2)	x	(21)	(8)/(3)	(6)/(2)	---
CP	443-1	IOC, IOD	(1)	(3)/(4)+(9)	(2)	---	(21)	(8)/(4)+(9)	(6)/(2)	---
	443-1 Advanced	IOC, IOD	(1)	(3)/(4)+(9)	(2)	x	(21)	(8)/(4)+(9)	(6)/(2)	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DA
- (3) TSEND/TRCV, TUSEND/TURCV
- (4) TSEND/TRCV
- (6) PNIO_SEND, PNIO_RECV
- (21) Controller 1 ist Server (für PUT, G
- (8) AG_SEND/AG_RECV
- (9) AG_SEND/AG_RECV, AG_LSEND/AG_LRECV, A

So wird die Tabelle gelesen (Beispiel in der roten Umrandung):

- Alle S7-300 CPUs mit PN Schnittstelle (Controller 1) können mit CP 443-1 Advanced (Controller 2) kommunizieren.
- Mögliche Kommunikationsarten:
 - S7-Kommunikation (S7)
 - Offene-Kommunikation (OC)
 - PN-Kommunikation (PN)
- Mögliche Kommunikationsbausteine bei Offene-Kommunikation:
 - Controller 1: T-Bausteine (3)
 - Controller 2: T-Bausteine (4) und Send/Receive-Bausteine (9)
- Mögliche Kommunikationsarten bei PN-Kommunikation:
 - PNIO mit den Anwenderschnittstellen (2)
 - CBA

Hinweis

An dieser Stelle soll nur das Prinzip der Tabelle gezeigt werden. Eine ausführliche Beschreibung des Aufbaus der Tabelle ist im Kapitel (17.3) zu finden.

3.2.4 Kommunikationsarten

Die folgende Abbildung zeigt den Abschnitt „Kommunikationsarten“ aus dem Sprungverteiler für PN/IE (Kapitel 18.1).

Abbildung 3-5

Kommunikationsart		Kapitel
Alle Kommunikationsarten (Tabelle Kompakt)		19.12
Tabelle mit Details	S7-Kommunikation	29.2
	Offene-Kommunikation mit Send/Receive-Bausteinen	31.2
	Offene-Kommunikation mit T-Bausteinen	32.2
	PNIO	34.2
Modbus/TCP (SIMATIC / fremder Controller)		44

Übersicht aller Kommunikationsarten

Vorgehensweise zur Beantwortung der Frage 3 (Welche Eigenschaften haben die zur Verfügung stehenden Kommunikationsarten?):

Das Kapitel (1) anklicken (online) oder aufschlagen (offline).

Ergebnis:

Die Tabelle in diesem Kapitel zeigt alle über PN/IE möglichen Kommunikationsarten im Vergleich.

Die folgende Abbildung zeigt einen Ausschnitt aus der Tabelle. (Tabelle 19-58).

Abbildung 3-6

	SIMATIC spezifisch		Offener Standard	
	S7-Kommunikation		Offene-Kommunikation	
			Send/Receive-Bausteine	T-Bausteine
Protokolle	ISO, IoT		ISO, IoT, TCP, UDP	IoT, TCP, UDP
Schnittstellen	ISO: CP IoT: CPU, CP		CP	CPU, CP
Kommunikationsbausteine (max. Daten)	BSEND (≤ 64 KByte) USEND, USEND_E (≥ 160 Byte) PUT, PUT_E, GET, GET_E (≥ 160 Byte)		AG_xSEND (ISO, IoT, TCP ≤ 8 KByte) (UDP ≤ 2 KByte) ----- Server für FETCH, WRITE (nicht bei UDP)	TSEND, TUSEND, ... (IoT ≤ 32 KByte) (TCP ≤ 64 KByte) (UDP = 1472 Byte)
remote Quittierung	BSEND: Applikation USEND: Transport PUT, GET: Applikation		ISO, IoT, TCP: Transport UDP: keine	IoT, TCP: Transport UDP: keine
Routingfähig?	ISO: nein IoT: ja		ISO: nein sonst: ja	ja
Verbindungen?	ja		UDP: nein sonst: ja	UDP: nein sonst: ja

Details zu einer speziellen Kommunikationsart

Vorgehensweise zur Beantwortung der Frage 4 (Wie sehen die Anwenderschnittstellen (Kommunikationsbausteine) konkret aus?):

Das Kapitel (2) anklicken (online) oder aufschlagen (offline).

Ergebnis:

Die Tabelle in diesem Kapitel zeigt alle wichtigen Eigenschaften der S7-Kommunikation.

Die folgende Abbildung zeigt einen Ausschnitt aus der Tabelle (Tabelle 29-1).

Abbildung 3-7

Kommunikationsart:		S7-Kommunikation	
Protokoll:		S7-Protokoll (*1)	
Allgemeines			
Medien		MPI, PB, PN/E, Rückwandbus (*3)	
Schnittstellen		CPU, CP	
Anbindung	SIMATIC S5	nein	
	Fremd (offene Standards)	nein	
Anwenderschnittstelle			
Kommunikationsbausteine		BSEND / BRCV	USEND / URCV USEND_E / URCV_E (*4)
maximale Anzahl Daten (*2)		<= 64 KByte	>= 160 Byte
Anzahl Variablen bei Aufruf Kommunikationsbaustein		1	S7-300: USEND/URCV: 1 USEND_E/URCV_E: 1 bis 4
			S7-400: 1 bis 4
dynamische Adressierung Daten		S7-300: ja	S7-300: ja
		S7-400: nein	S7-400: nein
remote Quittierung		Applikation	Transport
Modell		Client / Client	Client / Client

Werden weitere Informationen zu den Kommunikationsbausteinen benötigt, dann können diese im Kapitel zur S7-Kommunikation nachgelesen werden.

Beispiel: Parameter des Kommunikationsbausteines BSEND (Tabelle 29-12):

Abbildung 3-8

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Sendeauftrag
ID	WORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
R_ID	DWORD	Zuordnung Sende SFB/FB und Empfangs SFB/FB. Dies ermöglicht die Kommunikation mehrerer SFB/FB Paare über dieselbe logische Verbindung.
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
SD_1	ANY	Sendebereich (*2)
LEN	WORD	Länge der zu sendenden Daten

(*1): remote Quittierung: Applikation

(*2): SIMATIC S7 Speicherbereiche:

- S7-300: M, D
- S7-400: E, A, M, D, T, Z

TEIL 1: Einführung

Teil 1 dient der Einführung ins Thema CPU-CPU Kommunikation

Erläuterung von Begriffen und Zusammenhängen,
die für das Verständnis des Dokumentes wichtig sind.

TEIL 1: Gliederung und Inhalt

Tabelle 3-1

Kapitel	Gliederung	Inhalt
4	Funktionsmodell CPU-CPU Kommunikation	Übersicht Funktionsmodelle
5	Verbindungen bei SIMATIC	Das Wichtigste über Verbindungen
6	Datenkonsistenz bei SIMATIC	Erläuterungen zur Datenkonsistenz
7	SIMATIC Controller	Übersicht SIMATIC Controller und Familien
8	Medien zur SIMATIC Kommunikation	Übersicht aller Medien zur SIMATIC Kommunikation
9	PROFINET/Industrial Ethernet (PN/IE)	Pro Medium wird beschrieben: <ul style="list-style-type: none"> • Merkmale • ISO/OSI Referenzmodell
10	PROFIBUS (PB)	
11	MPI	
12	SIMATIC Rückwandbus	
13	Serielle Schnittstelle (PtP)	
14	Gegenüberstellung der Medien	Tabellarische Gegenüberstellung der Medien
15	Schnittstellen der SIMATIC Familien	Übersicht aller SIMATIC Schnittstellen
16	Informationen	Hinweis auf weitere Informationen

4 Modelle zur CPU-CPU Kommunikation

4.1 Definition Controller

Im Dokument wird die folgende Definition verwendet:

Ein Controller ist eine zentrale oder dezentrale Automatisierungsstation (Station) mit den Komponenten CPU, CP (optional) und Peripherie. Die Komponenten sind innerhalb der Station über den Rückwandbus verbunden.

Zentrale Station:

- enthält zentrale Peripherie
- kommuniziert mit dezentralen Stationen über PROFINET IO oder PROFIBUS DP

Dezentrale Station:

- enthält dezentrale Peripherie
- kommuniziert mit zentraler Station über PROFINET IO oder PROFIBUS DP

4.2 Definition CPU-CPU Kommunikation

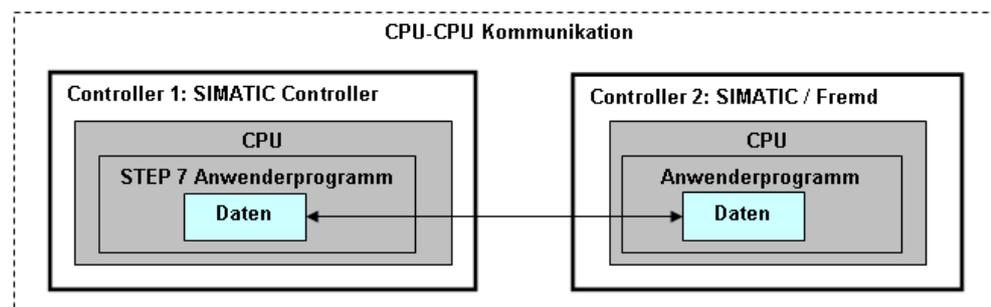
Bei der CPU-CPU Kommunikation werden Daten zwischen zwei Controllern ausgetauscht:

- Controller 1: SIMATIC Controller
- Controller 2: SIMATIC Controller oder fremder Controller

Quelle oder Ziel der Daten ist der Anwenderdatenbereich der CPU des Controllers:

- Datenbaustein, Merker, Eingänge, Ausgänge, ...

Abbildung 4-1



Bei der CPU-CPU Kommunikation werden folgende Fälle unterschieden:

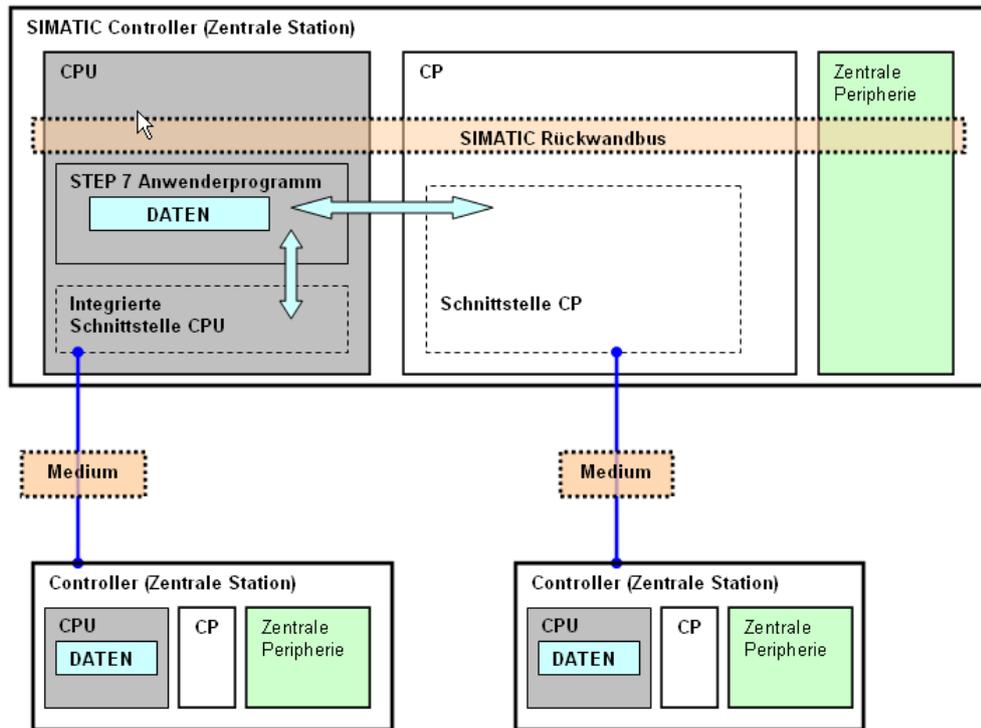
- CPUs in unterschiedlichen zentralen Stationen
- CPUs in zentraler und dezentraler Station (*1)
- CPUs innerhalb einer zentralen SIMATIC Station

Zu (*1): Eine dezentrale Station mit CPU wird auch mit I-Slave (bei PROFIBUS) bzw. I-Device (bei PROFINET) bezeichnet.

4.3 CPUs in unterschiedlichen zentralen Stationen

Das Bild zeigt das Funktionsmodell für die CPU-CPU Kommunikation zwischen zentralen Stationen.

Abbildung 4-2



Schnittstellen zur Kommunikation:

- Schnittstelle auf CPU (integrierte Schnittstelle)
- Schnittstelle auf CP (externe Schnittstelle)

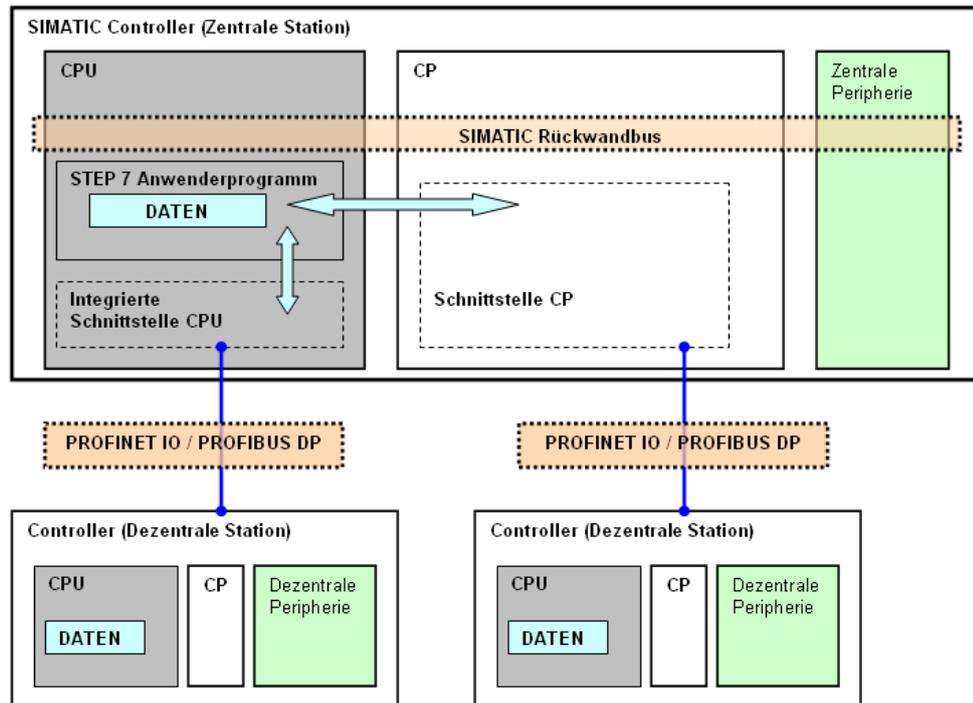
Medien zur Kommunikation:

- Netz (PROFINET/Industrial Ethernet, PROFIBUS, MPI)
- Serielle Schnittstelle (ASCII, 3964(R), RK 512, ...)

4.4 CPUs in zentraler und dezentraler Station

Das Bild zeigt das Funktionsmodell für die CPU-CPU Kommunikation zwischen zentraler und dezentraler Station.

Abbildung 4-3



Schnittstellen zur Kommunikation:

- Schnittstelle auf CPU (integrierte Schnittstelle)
- Schnittstelle auf CP (externe Schnittstelle)

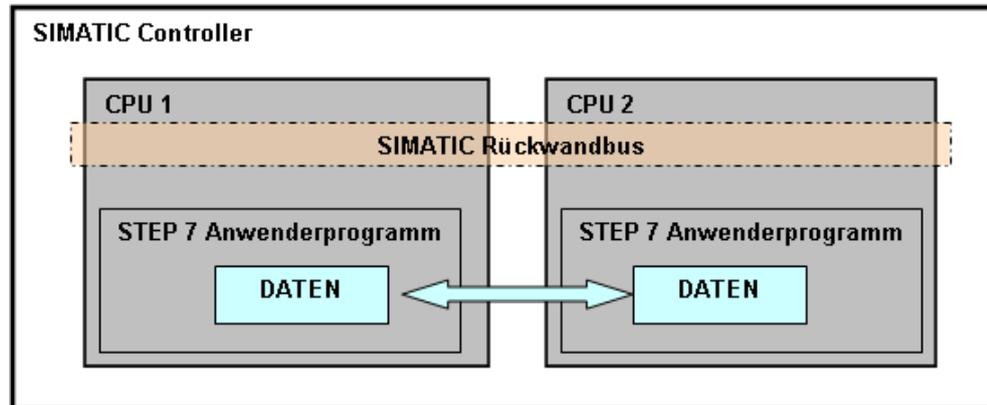
Medien zur Kommunikation:

- PROFINET/Industrial Ethernet (PROFINET IO)
- PROFIBUS (PROFIBUS DP)

4.5 CPUs innerhalb einer zentralen Station

Das Bild zeigt das Funktionsmodell für die CPU-CPU Kommunikation zwischen CPUs innerhalb einer zentralen SIMATIC Station.

Abbildung 4-4



Medium zur Kommunikation:

- SIMATIC Rückwandbus

Hinweis

Dies ist nur bei S7-400 möglich, und wird dort mit „Multicomputing“ bezeichnet. Es können bis zu 4 S7-CPU's in einer zentralen SIMATIC Station gleichzeitig betrieben werden.

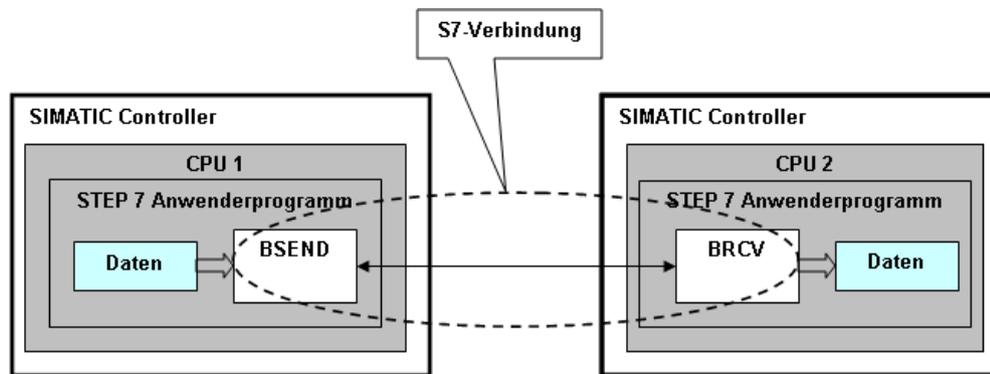
5 Verbindungen bei SIMATIC

5.1 Verbindungen

Einführung

CPU-CPU Kommunikation erfolgt bei den meisten Kommunikationsarten der SIMATIC über Verbindungen. Eine Verbindung legt fest, welche Kommunikationspartner (CPUs) miteinander kommunizieren. Das folgende Bild zeigt ein Beispiel einer Verbindung: CPU 1 kommuniziert mit CPU 2 über eine S7-Verbindung.

Abbildung 5-1



Verbindungen müssen eingerichtet werden (Kapitel 5.4). Dies kann zum Beispiel über Projektierung erfolgen. Das folgende Bild zeigt eine in STEP 7 (NetPro) projektierte S7-Verbindung.

Abbildung 5-2

Local ID	Partner ID	Partner	Type	Active connection partner	Subnet
1	2	SIMATIC 300(1) / CPU 317-2 DP	S7 connection	Yes	Ethernet(1) [IE]

Copyright © Siemens AG 2011 All rights reserved
20982954_SIMATIC_Comm_DOKU_v201_d.doc

Eigenschaften einer Verbindung

Eine Verbindung ist definiert durch folgende Eigenschaften (Beispiele):

- beteiligte CPUs (CPU 1, CPU 2)
- verwendetes Protokoll (TCP, ISO on TCP, ...)
- Verhalten nach der Datenübertragung:
Verbindung bleibt aufgebaut, oder wird abgebaut

Funktion einer Verbindung

Beim Verbindungsaufbau handeln die Schnittstellen der beiden Kommunikationspartner (integrierte Schnittstelle der CPU oder Schnittstelle des CP) die Verbindungsparameter (maximale Telegrammlänge, ...) aus.

Der aktive Kommunikationspartner schlägt einen Wert vor. Der andere Kommunikationspartner bestätigt oder macht einen anderen Vorschlag. Der ausgehandelte Wert gilt dann für die Dauer der Verbindung.

Bei aufgebauter Verbindung werden vom Betriebssystem der CPU oder des CP folgende Aufgaben ausgeführt:

- Flusssteuerung (Vermeidung von Überlast in den Partnern, ...)
- Überwachung der Verbindung (Partner noch erreichbar, ...)
- Austausch von Quittungen (Daten fehlerfrei angekommen, ...)

5.2 Protokolle

Ein Protokoll legt fest, nach welchen Regeln die Kommunikation zwischen zwei Kommunikationspartnern abläuft. Zur Beschreibung eines Protokolls wird häufig das ISO/OSI-Referenzmodell verwendet (Kapitel 50.1).

Bei den Protokollen werden zwei Klassen unterschieden.

Verbindungsorientierte Protokolle

Diese Protokolle bauen zwischen beiden Kommunikationspartnern eine Verbindung auf.

Beispiele: TCP, ISO on TCP

Die Protokolle werden eingesetzt, wenn die Priorität bei einer zuverlässigen Übertragung der Daten liegt

Verbindungslose Protokolle

Diese Protokolle bauen zwischen beiden Kommunikationspartnern keine Verbindung auf.

Beispiel: UDP

Die Protokolle werden eingesetzt, wenn die Priorität bei einer schnellen Übertragung der Daten liegt.

5.3 Verbindungsressourcen

Bedeutung

Verbindungen belegen Ressourcen auf der CPU oder dem CP (z.B. Speicherbereiche des Betriebssystems). Diese Ressourcen werden bei der SIMATIC mit „Verbindungsressourcen“ bezeichnet.

Die Anzahl der maximal möglichen Verbindungen pro CPU oder CP ist begrenzt. Sie hängt von den zur Verfügung stehenden Verbindungsressourcen der CPU oder des CP ab. In den Handbüchern für CPUs und CPs ist deswegen in den technischen Daten angegeben, wie viele Verbindungen pro Kommunikationsart möglich sind.

Das folgende Bild zeigt dazu ein Beispiel aus dem Handbuch S7-300.

Abbildung 5-3

Technische Daten	
<i>Technische Daten der CPU 31x</i>	
<i>8.5 CPU 315-2 PN/DP</i>	
Technische Daten	
Diagnosepuffer	Ja
• Anzahl der Einträge (nicht einstellbar)	max. 500
• davon netzausfallsicher	100 letzten Einträge sind remanent
Kommunikationsfunktionen	
Offene IE-Kommunikation	
Anzahl Verbindungen / Zugangspunkte, gesamt	8
TCP/IP	
	Ja (über integrierte PROFINET-Schnittstelle und ladbare FBs)
• Anzahl Verbindungen, max.	8
• Datenlänge bei Verbindungstyp 01H, max.	1460 Byte
• Datenlänge bei Verbindungstyp 11H, max.	8192 Byte
ISO on TCP	
	Ja (über integrierte PROFINET-Schnittstelle und ladbare FBs)
• Anzahl Verbindungen, max.	8
• Datenlänge, max.	8192 Byte
UDP	
	Ja (über integrierte PROFINET-Schnittstelle und ladbare FBs)
• Anzahl Verbindungen, max.	8
• Datenlänge, max.	1472 Byte

Belegung und Freigabe

Verbindungsressourcen werden beim Verbindungsaufbau belegt, und beim Verbindungsabbau wieder freigegeben.

Werden Verbindungen in STEP 7 (NetPro) projektiert, dann überwacht STEP 7 die Einhaltung der technischen Daten bezüglich der maximal möglichen Verbindungen.

Werden Verbindungen im STEP 7 Anwenderprogramm aufgebaut bzw. abgebaut, muss der Anwender die Verbindungsressourcen selbst verwalten. D.h. der Anwender muss sicherstellen, dass die Anzahl maximal möglicher Verbindungen, der an der Kommunikation beteiligten Baugruppen, eingehalten wird.

5.4 Einrichten von Verbindungen

Verbindungen müssen eingerichtet werden.
Dabei sind folgende Fälle zu unterscheiden:

- projektierte Verbindung
- nicht projektierte Verbindung

In den folgenden Kapiteln werden die beiden Fälle beschrieben.

Tabelle 5-1

	Kapitel
Projektierte Verbindung	5.5
Nicht projektierte Verbindung	5.6

5.5 Projektierte Verbindung

Projektierte Verbindungen sind Verbindungen, die mit STEP 7 (NetPro) eingerichtet werden. Das folgende Bild zeigt ein Beispiel.

Abbildung 5-4

Local ID	Partner ID	Partner	Type	Active connection partner	Subnet
1	2	SIMATIC 300(1) / CPU 317-2 DP	S7 connection	Yes	Ethernet(1) [IE]

Projektierte Verbindungen werden zum Beispiel bei folgenden Kommunikationsarten verwendet:

- S7-Kommunikation
- Offene-Kommunikation mit Send/Receive-Bausteinen

Realisierung einer Kommunikation

Um eine Kommunikation zu realisieren, sind folgende Schritte auszuführen:

- Projektierung mit STEP 7:
Verbindungsprojektierung mit NetPro
- Programmierung in STEP 7:
Aufruf Kommunikationsbausteine (zur Datenübertragung)

Bei der Verbindungsprojektierung wird pro Verbindung eine eindeutige lokale Kennung vergeben, die "Lokale ID". Diese Lokale ID wird bei der Parametrierung der Kommunikationsbausteine benötigt.

Aufbau und Abbau der Verbindung

Aufbau

Automatischer Aufbau der Verbindung beim Hochlauf der kommunikationsfähigen Baugruppen (CPU, CP). Mit dem Aufbau werden die erforderlichen Verbindungsressourcen vom Betriebssystem der Baugruppen belegt.

Abbau

Die Verbindung wird nach der Datenübertragung nicht abgebaut. D.h. die Verbindungsressourcen bleiben dauerhaft belegt.

5.6 Nicht projektierte Verbindung

Nicht projektierte Verbindungen sind Verbindungen, die nicht mit STEP 7 (NetPro) projektiert werden.

Nicht projektierte Verbindungen werden bei den folgenden Kommunikationsarten verwendet (Beispiele):

- S7-Basiskommunikation
- Offene-Kommunikation mit T-Bausteinen

Es sind zwei Fälle zu unterscheiden:

- Automatisches Einrichten einer Verbindung
(Beispiel: S7-Basiskommunikation)
- Programmieretes Einrichten einer Verbindung
(Beispiel: Offene-Kommunikation mit T-Bausteinen)

5.6.1 Automatisches Einrichten einer Verbindung

Realisierung einer Kommunikation

Um eine Kommunikation zu realisieren, sind folgende Schritte auszuführen:

- Programmierung in STEP 7:
Aufruf Kommunikationsbausteine (zur Datenübertragung)

Aufbau und Abbau der Verbindung

Aufbau

Bei erstmaligem Aufruf des Kommunikationsbausteines erfolgt ein Verbindungsaufbau durch das Betriebssystem der kommunikationsfähigen Baugruppen (CPU, CP). Mit dem Aufbau werden die erforderlichen Verbindungsressourcen vom Betriebssystem der Baugruppen belegt.

Abbau

Durch Parametrierung am Kommunikationsbaustein wird festgelegt, ob die Verbindung nach Abschluss der Datenübertragung bestehen bleibt, oder wieder abgebaut wird. Durch das Abbauen einer bestehenden Verbindung werden vorher belegte Verbindungsressourcen wieder freigegeben.

Wird die Verbindung nach der Übertragung der Daten nicht abgebaut, ist die Bearbeitungszeit des Kommunikationsbausteines bei einer erneuten Übertragung kürzer, da die Verbindung bereits steht, und nicht aufgebaut werden muss.

5.6.2 Programmieretes Einrichten einer Verbindung

Realisierung einer Kommunikation

Um eine Kommunikation zu realisieren, sind folgende Schritte auszuführen:

- Programmierung in STEP 7:
Aufruf Verbindungsbausteine (zum Aufbau von Verbindungen)
- Programmierung in STEP 7:
Aufruf Kommunikationsbausteine (zur Datenübertragung)

Aufbau und Abbau Verbindung

Aufbau

Bei Aufruf des Verbindungsbausteines erfolgt ein Verbindungsaufbau durch das Betriebssystem der kommunikationsfähigen Baugruppen (CPU, CP).

Abbau

Die Verbindungen können durch Aufruf eines Verbindungsbausteines abgebaut werden. Damit können Verbindungsressourcen wieder freigegeben werden.

6 Datenkonsistenz bei SIMATIC

In diesem Kapitel wird das Thema Datenkonsistenz aus Sicht der CPU-CPU Kommunikation mit SIMATIC Controllern betrachtet.

6.1 Definitionen

Datenbereich

Ein Datenbereich ist ein zusammenhängender Bereich von Daten im Anwenderspeicher einer SIMATIC CPU (zum Beispiel: MW100 bis MW200).

Konsistente Daten

Ein Datenbereich, der nicht gleichzeitig durch konkurrierende Prozesse (Anwenderprogramme, Betriebssystemprogramme, ...) verändert werden kann, wird als konsistenter (zusammengehöriger) Datenbereich bezeichnet.

Dieser Datenbereich enthält konsistente Daten. Die Größe dieses Datenbereiches wird im Dokument mit „Anzahl konsistenter Daten“ bezeichnet.

Inkonsistente Daten

Ein Datenbereich, der größer als der konsistente Datenbereich ist, kann verfälscht werden. Der Datenbereich kann zu einem Zeitpunkt teilweise aus neuen und teilweise aus alten konsistenten Datenbereichen bestehen.

Beispiel

Inkonsistente Daten können entstehen, wenn im STEP 7 Anwenderprogramm ein laufender Kommunikationsbaustein durch einen Prozessalarm OB mit höherer Priorität unterbrochen wird. Verändert das Anwenderprogramm in diesem OB jetzt die Daten, die teilweise bereits vom Kommunikationsbaustein verarbeitet wurden, dann können inkonsistente Daten entstehen.

Die Daten sind dann inkonsistent (nicht zusammengehörig), weil:

- Ein Teil der Daten stammt aus der Zeit vor der Prozessalarmbearbeitung („alter Datenbereich“)
- Ein Teil der Daten stammt aus der Zeit nach der Prozessalarmbearbeitung („neuer Datenbereich“)

Datenkonsistenz

Bei Datenkonsistenz werden zwei Fälle unterschieden:

- Systembedingte Datenkonsistenz
- weitergehende Datenkonsistenz

Eigenschaften der systembedingten Datenkonsistenz:

Unter der Randbedingung (a) gilt (b):

(a): Datenbereich mit „Anzahl Daten“ \leq „Anzahl konsistenter Daten“

(b): Datenkonsistenz ist ohne Zusatzmaßnahmen im STEP 7 Anwenderprogramm garantiert

Eigenschaften der weitergehenden Datenkonsistenz:

Unter der Randbedingung (a) gilt (b):

(a): Datenbereich mit „Anzahl Daten“ > „Anzahl konsistenter Daten“

(b): Datenkonsistenz ist nicht garantiert. Datenkonsistenz ist nur mit Zusatzmaßnahmen im STEP 7 Anwenderprogramm erreichbar (Beispiel: Prozessalarm OB sperren für die Zeitdauer der Datenübertragung)

6.2 Systembedingte Datenkonsistenz

Das Betriebssystem der SIMATIC Controller garantiert eine systembedingte Datenkonsistenz. Diese systembedingte Datenkonsistenz hängt ab von:

- Typ CPU (falls Kommunikation über CPU)
- Typ CP und Typ CPU (falls Kommunikation über CP)
- Typ Kommunikationsbaustein

Es folgen Aussagen zu den verschiedenen SIMATIC Familien.

6.2.1 S7-300

Bei S7-300 werden die Daten in Blöcken zu x Byte (siehe unten) vom Betriebssystem konsistent in den STEP 7 Anwenderspeicher kopiert. Das Kopieren erfolgt im Zykluskontrollpunkt des Betriebssystems. Für größere Datenbereiche wird vom System keine Datenkonsistenz garantiert.

Ist eine bestimmte Datenkonsistenz gefordert, so dürfen die Daten im Anwenderprogramm nicht größer als diese x Byte sein.

Tabelle 6-1

Fallunterscheidung	Anzahl konsistenter Daten
Kommunikation über integrierte Schnittstelle der CPU	64 Byte bis 240 Byte (*1)
Kommunikation über CP	32 Byte

(*1): Die konkreten Werte zur „Anzahl konsistenter Daten“ sind in den Handbüchern der CPUs bzw. CPs zu finden.

6.2.2 S7-400

Bei S7-400 werden, im Gegensatz zur S7-300, die Daten nicht im Zykluskontrollpunkt des Betriebssystems, sondern in festen Zeitscheiben während des Zyklusses (OB1) bearbeitet.

Es wird die Datenkonsistenz einer Variablen (Byte, Wort oder Doppelwort) garantiert. Die maximale Datenkonsistenz beträgt damit 32 Byte.

6.2.3 S7-1200

Die CPU garantiert die Datenkonsistenz für alle elementaren Datentypen (z.B. Word oder DWord) und alle systemdefinierten Strukturen (z.B. IEC_TIMERS oder DTL).

6.3 Weitergehende Datenkonsistenz

6.3.1 Zusatzmaßnahmen

Um weitergehende Datenkonsistenz zu gewährleisten, müssen Zusatzmaßnahmen im STEP 7 Anwenderprogramm des Senders und Empfängers getroffen werden:

Zusatzmaßnahmen im Sender

Zugriff auf den Sendebereich (Datenbaustein, Merker, ...) erst dann, wenn die Daten komplett übertragen wurden. Dies kann an den Kontrollparametern der Kommunikationsbausteine abgelesen werden (Beispiel: DONE = 1).

Zusatzmaßnahmen im Empfänger

Zugriff auf den Empfangsbereich (Datenbaustein, Merker, ...) erst dann, wenn die Daten komplett empfangen wurden. Dies kann an den Kontrollparametern der Kommunikationsbausteine abgelesen werden (Beispiel: NDR = 1).

Anschließend Sperren des Empfangsbereiches solange, bis die Daten bearbeitet wurden. Dies kann an den Kontrollparameter der Kommunikationsbausteine abgelesen werden (Beispiel: EN_R = 0).

6.3.2 Fallunterscheidung

Es sind zwei Fälle zu unterscheiden:

- Client Client Kommunikation
- Client Server Kommunikation

Client Client Kommunikation

Beispiele für Kommunikationsbausteine: BSEND / BRCV

Soll weitergehende Datenkonsistenz gewährleistet werden, dann dürfen während der Übertragung die Daten nicht verändert werden (siehe oben „Zusatzmaßnahmen“).

Client Server Kommunikation

Beispiele für Kommunikationsbausteine: PUT, GET

Im STEP 7 Anwenderprogramm des Servers ist kein Kommunikationsbaustein vorhanden. Deswegen kann der Zugriff auf die Daten im Anwenderprogramm nicht koordiniert werden.

Hier muss bereits bei der Programmierung bzw. Projektierung die systembedingte Größe der konsistenten Datenbereiche (systembedingte Datenkonsistenz) berücksichtigt werden.

7 SIMATIC Controller

Es folgt eine Übersicht der im Dokument betrachteten SIMATIC Familien ([/1/](#)). Die im Dokument verwendeten Kurz-Bezeichnungen für die SIMATIC Familien sind in Klammern gesetzt.

Hinweis

Die Oberbegriffe (SIMATIC Modular Controller, ...) zu den SIMATIC Familien (ET 200 CPU, S7-300, ...) entsprechen der aktuellen Navigation im Produktsupport des Service & Support Portals (Stand September 2010, [/0/](#)).

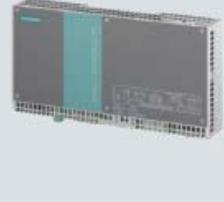
SIMATIC Modular Controller

Tabelle 7-1

SIMATIC Familie	
SIMATIC ET 200 (ET 200 CPU) Modulares, dezentrales Peripheriesystem mit Intelligenz vor Ort	
SIMATIC S7-300 (S7-300) Der modulare Controller für Systemlösungen in der Fertigungsindustrie	
SIMATIC S7-400 (S7-400) Der Power-Controller für Systemlösungen in der Fertigungs- und Prozessindustrie	
SIMATIC S7-1200 (S7-1200) Modularer, kompakter Controller für diskrete und Stand-Alone-Automatisierungslösungen	

SIMATIC Embedded Automation

Tabelle 7-2

SIMATIC Familie	
SIMATIC S7-modular Embedded Controller (S7-mEC) Embedded Controller in S7-300 Aufbautechnik mit Software Controller und Runtime-Visualisierungssoftware	
SIMATIC Embedded Box PC-Bundles (Box PC) Einschaltfertiger Hutschienen-PC mit Software Controller und Runtime-Visualisierungssoftware	
SIMATIC Embedded Panel PC-Bundles (Panel PC) Einschaltfertiger Panel PC mit Software Controller und Runtime-Visualisierungssoftware	
SIMATIC WinAC MP für Multi Panels (WinAC MP) Software Controller für Multi Panels	

SIMATIC PC-based Controller

Tabelle 7-3

SIMATIC Familie	
SIMATIC WinAC RTX (WinAC RTX) Software Controller – offen, flexibel und zuverlässig	

8 Medien zur SIMATIC Kommunikation

Um eine CPU-CPU Kommunikation zu realisieren, gibt es für die SIMATIC unterschiedliche Möglichkeiten. So kann die Übertragung der Daten über verschiedene Medien erfolgen. Die folgende Tabelle zeigt welche Medien zur Verfügung stehen.

Tabelle 8-1

Medium		Kommunikationspartner	
		SIMATIC S7 Controller	Fremder Controller
Netz	PROFINET/Industrial Ethernet (PN/IE)	x	x
	PROFIBUS (PB)	x	x
	MPI	x	---
SIMATIC Rückwandbus (*1)		x	---
Serielle Schnittstelle (PtP)		x	x

(*1): nur möglich bei SIMATIC S7-400 (Multicomputing, Kapitel 4.5)

In den folgenden Kapiteln werden die Medien kurz charakterisiert.

Details zu den Medien können in der zahlreich vorhandenen Literatur nachgelesen werden (Kapitel 16).

Übersicht der folgenden Kapitel:

Tabelle 8-2

Medien zur SIMATIC Kommunikation	Kapitel
PROFINET/Industrial Ethernet (PN/IE)	9
Vorbemerkung	9.1
Ethernet	9.2
Industrial Ethernet (IE)	9.3
PROFINET (PN)	9.4
PROFIBUS (PB)	10
MPI	11
SIMATIC Rückwandbus	12
Serielle Schnittstelle (PtP)	13
Gegenüberstellung der Medien	14

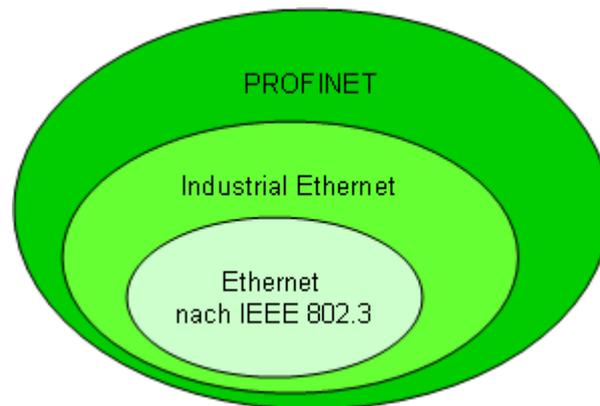
9 PROFINET/Industrial Ethernet (PN/IE)

9.1 Vorbemerkung

PROFINET, Industrial Ethernet und Ethernet sind Begriffe, die im Zusammenhang stehen:

- PROFINET basiert auf Industrial Ethernet
- Industrial Ethernet basiert auf Ethernet.

Abbildung 9-1



Im Folgenden werden Ethernet, Industrial Ethernet und PROFINET kurz charakterisiert.

9.2 Ethernet

Einführung

Ethernet ist der Standard für Netze (LAN) aus dem Bürobereich.

Merkmale

- International genormt: IEEE 802.3
- Weltweite Verbreitung
- Einfache und standardisierte Verkabelung
- Basis für überlagerte Protokolle (TCP/IP, UDP, ...)
- Ausfallsichere Netze durch Redundanz
- Einfache Anbindung an drahtlose Netze (Industrial Wireless LAN, nach IEEE 802.11)
- Skalierbare Leistung durch Switched-Ethernet (*1)

(*1): Switched-Ethernet

Switched-Ethernet unterteilt das Netz in Teilnetze, die mit Switches verbunden sind.

Damit ist folgende Funktionalität realisierbar:

- Mehrere Paare von Teilnehmern werden gleichzeitig miteinander verbunden. Jede Verbindung verfügt über den vollen Datendurchsatz.
- Lokaler Datenverkehr bleibt lokal. Nur Daten eines anderen Teilnetzes werden von Switches weitergeleitet.

Vorteil von Switched-Ethernet:

- Erhöhung Datendurchsatz durch Strukturierung des Datenverkehrs

Ein Switch kann bei SIMATIC unterschiedlich realisiert sein:

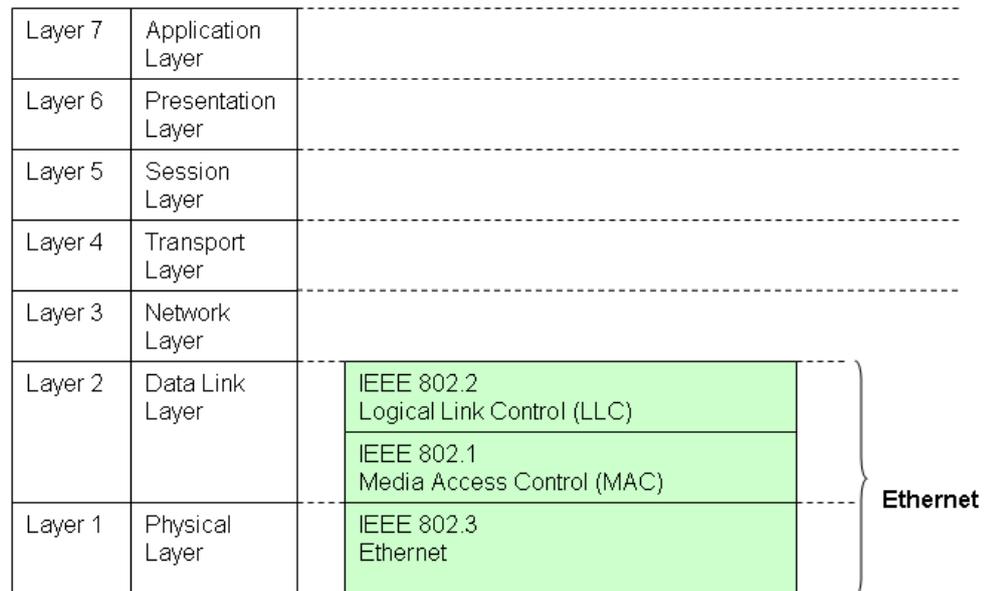
- als eigenständige Komponente (SCALANCE X)
- integriert in SIMATIC Komponenten (S7-CPU, S7-CP mit PN/IE Anschluss)

ISO/OSI-Referenzmodell

Ethernet umfasst Schicht 1 und Schicht 2 des ISO/OSI-Referenzmodells:

- Schicht 2: Zugriffskontrolle und Adressierung (MAC-Adressen)
- Schicht 1: Übertragungstechnik (Physik)

Abbildung 9-2



9.3 Industrial Ethernet (IE)

Einführung

IE ist die industrietaugliche Variante von Ethernet.

Merkmale

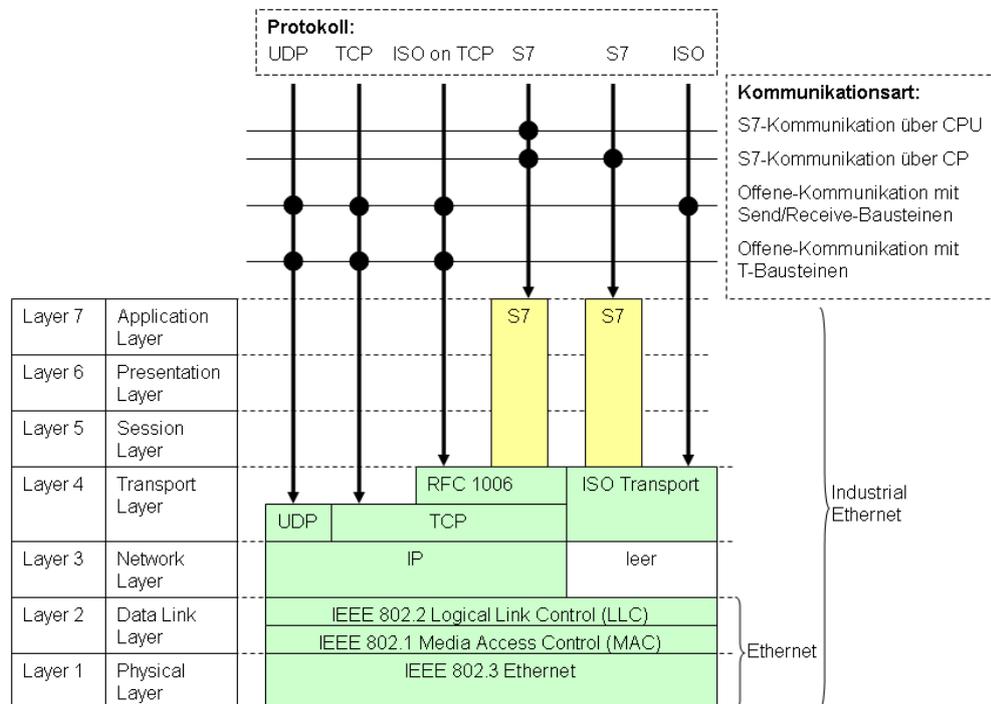
Neben den Merkmalen für Ethernet (Kapitel 9.2), gibt es für Industrial Ethernet folgende zusätzliche Merkmale:

- verbindet unterschiedliche Anwendungsbereiche: Büro und Fertigung
- nutzt die Möglichkeiten der IT-Standards (bekannt aus dem Bürobereich) in der Automatisierung (Browser, E-Mail, ...).
- Optimierte Kommunikation zwischen Automatisierungskomponenten und gleichzeitige Kommunikation gemäß TCP/IP (Offener Standard)
- Netzkomponenten für den Einsatz in rauer Industrieumgebung (Staub, Feuchtigkeit, Vibrationen, ...)
- einfache Anschlusstechnik vor Ort

ISO/OSI-Referenzmodell

Industrial Ethernet umfasst Schicht 1 bis Schicht 7 des ISO/OSI-Referenzmodells. Das folgende Bild zeigt alle Protokolle und Kommunikationsarten, die von SIMATIC Controllern unterstützt werden.

Abbildung 9-3



9.4 PROFINET (PN)

Einführung

PN ist der offene Industrial Ethernet Standard für die Automatisierung. PN setzt auf Industrial Ethernet auf.

Ausprägungen

Im Rahmen von PROFINET gibt es zwei Ausprägungen:

- PROFINET IO
- PROFINET CBA

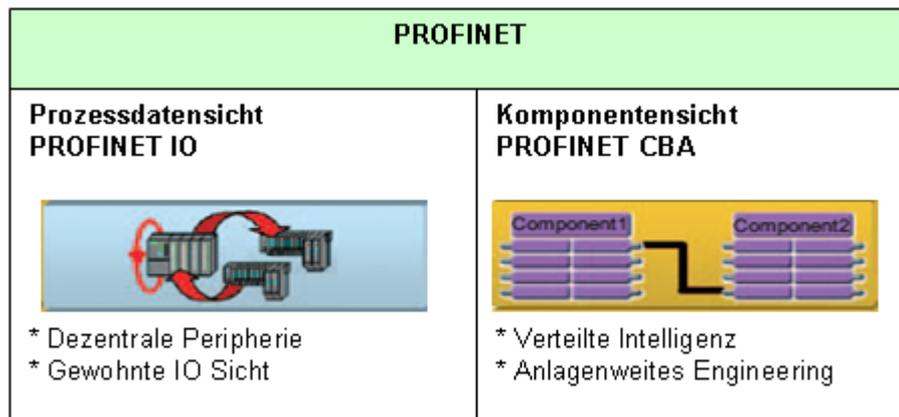
PROFINET IO ist ein Automatisierungskonzept für die Realisierung modularer Applikationen durch die Einbindung von dezentraler Peripherie mit Echtzeitkommunikation.

PROFINET CBA ist ein Komponentenmodell für Automatisierungslösungen auf Basis von verteilten Komponenten und Teilfunktionen.

Sichtweisen

PROFINET IO und PROFINET CBA sind zwei verschiedene Sichtweisen auf Automatisierungsgeräte am Industrial Ethernet.

Abbildung 9-4



PROFINET IO liefert ein Bild der Automatisierungsanlage, das der PROFIBUS DP Sichtweise sehr ähnlich ist. Die einzelnen Automatisierungsgeräte werden projektiert und programmiert.

PROFINET CBA gliedert eine komplette Automatisierungsanlage in verschiedene Funktionen auf. Diese Funktionen werden projektiert und programmiert.

Merkmale

PROFINET

- international genormt: IEC 61158, IEC 61784
- Durchgängige Kommunikation über Feldbus und Ethernet
- Integration vorhandener Feldbus-Systeme (PROFIBUS, ASi)
- Verwendung des TCP/IP Protokolls
- Kommunikation in Echtzeit
- Taktsynchrone Antriebsregelung für Motion Control Anwendungen

PROFINET IO

- Kommunikation von Feldgeräten (IO-Device) mit Controllern (IO-Controller)
- IO-Sicht, wie bei PROFIBUS DP

PROFINET CBA

- Kommunikation zwischen CBA Komponenten
- Kommunikation wird projiziert, nicht programmiert (mit dem Engineering Tool iMap)

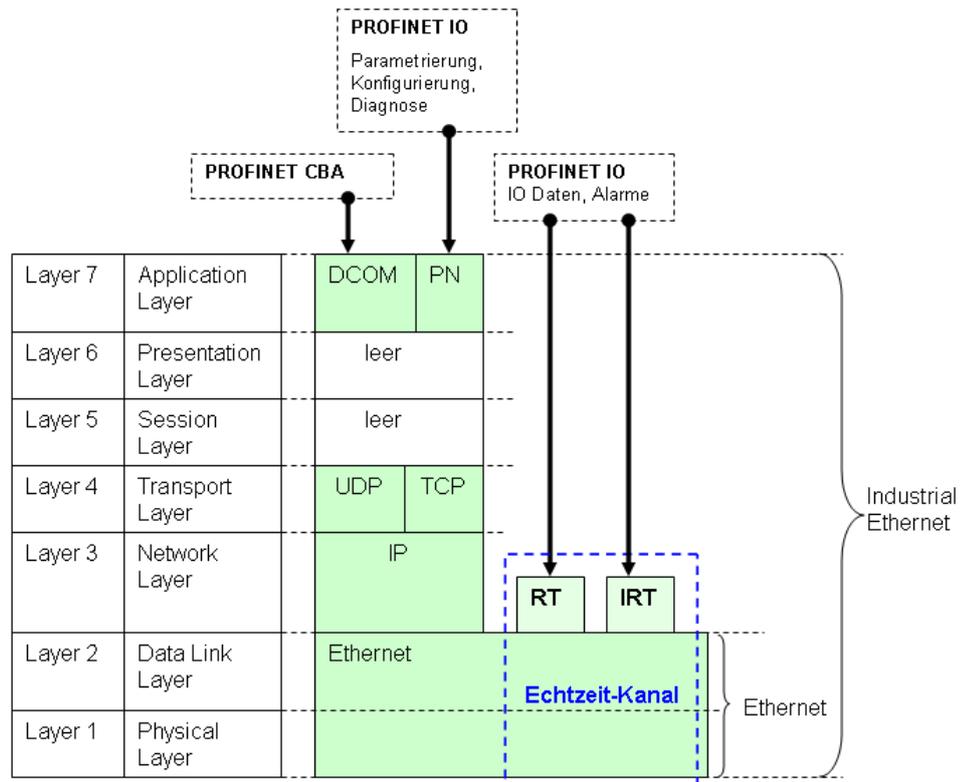
ISO/OSI-Referenzmodell

Die PROFINET Kommunikation basiert auf der Ethernet Kommunikation. Sie unterscheidet drei Kommunikationskanäle, bzw. drei Leistungsstufen:

Tabelle 9-1

Kommunikationskanal		Anwendung	Beispiele
TCP/IP		für nicht zeitkritische Kommunikation	<ul style="list-style-type: none"> • azyklisches Lesen und Schreiben von Datensätzen • Parametrierung • Konfiguration • Diagnose
Echtzeit	Real Time (RT)	für zeitkritische Kommunikation	<ul style="list-style-type: none"> • zyklisches Prozessabbild der Feldgeräte (IO Daten) • Alarmer
	Isochrones Real Time (IRT)	für hochperformante, deterministische und takt-synchrone Kommunikation	<ul style="list-style-type: none"> • Prozessdaten im Bereich Motion Control

Abbildung 9-5



10 PROFIBUS (PB)

Einführung

PROFIBUS ist ein einfaches, international standardisiertes, elektrisches Feldbus-system.

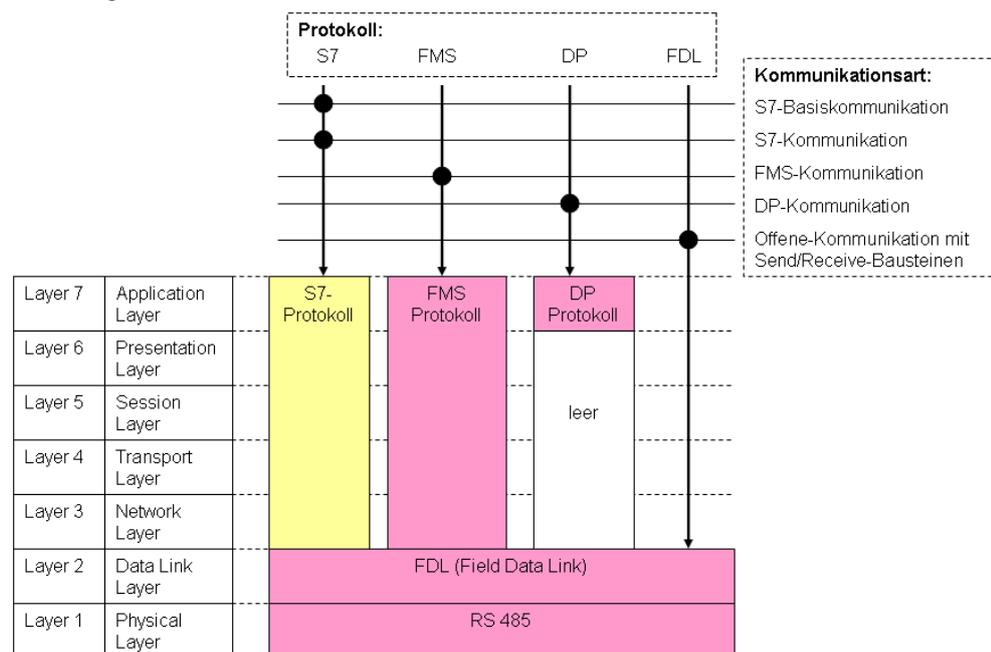
Merkmale

- international genormt: IEC61158, IEC61784
- Mechanismus Kommunikation zwischen Controllern:
Token Umlauf
- Mechanismus Kommunikation zwischen Feldgeräten und Controller:
Master Slave Prinzip

ISO/OSI-Referenzmodell

Das folgende Bild zeigt alle Protokolle und Kommunikationsarten, die von SIMATIC Controllern unterstützt werden.

Abbildung 10-1



11 MPI

Einführung

MPI ist das Netz der SIMATIC für die Kommunikation mit PG/OP und die CPU-CPU Kommunikation

Eine MPI Schnittstelle ist auf fast allen CPUs (*1) der modularen SIMATIC Controller integriert.

(*1): Ausnahme: S7-1200

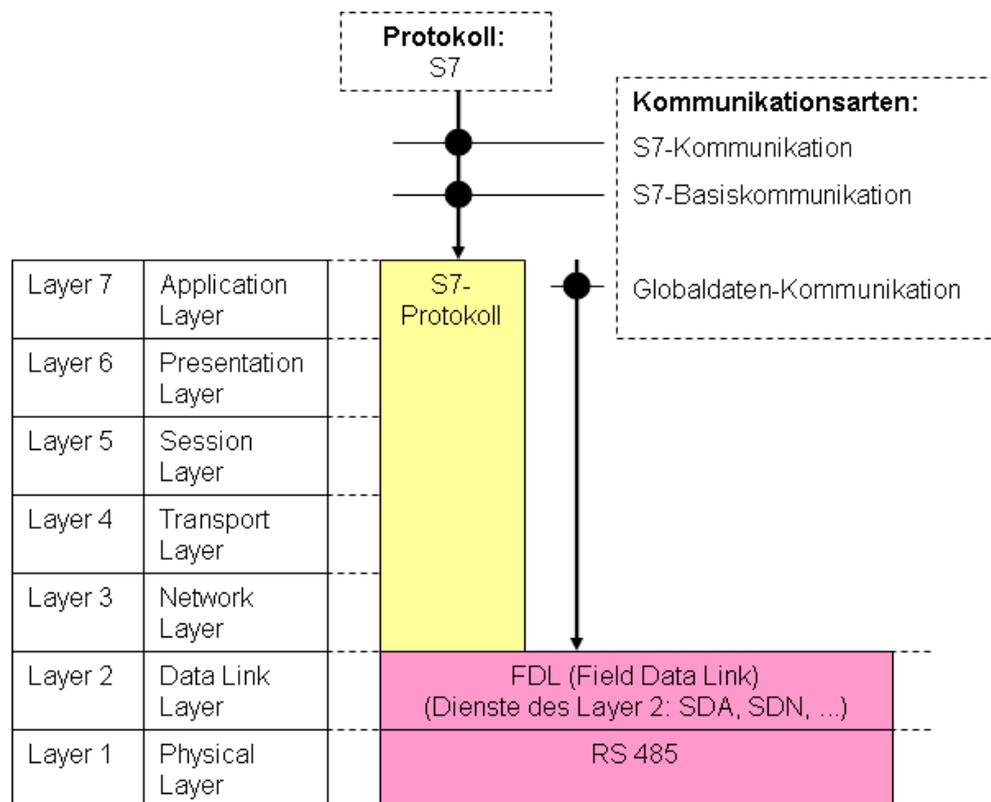
Merkmale

- Herstellerspezifische Schnittstelle (kein offener Standard)
- Netz mit geringer Ausdehnung und geringer Teilnehmerzahl
- MPI basiert auf PROFIBUS

ISO/OSI-Referenzmodell

Das folgende Bild zeigt alle Protokolle und Kommunikationsarten, die von SIMATIC Controllern unterstützt werden.

Abbildung 11-1



12 SIMATIC Rückwandbus

Einführung

Über den Rückwandbus ist CPU-CPU Kommunikation innerhalb einer SIMATIC Station möglich.

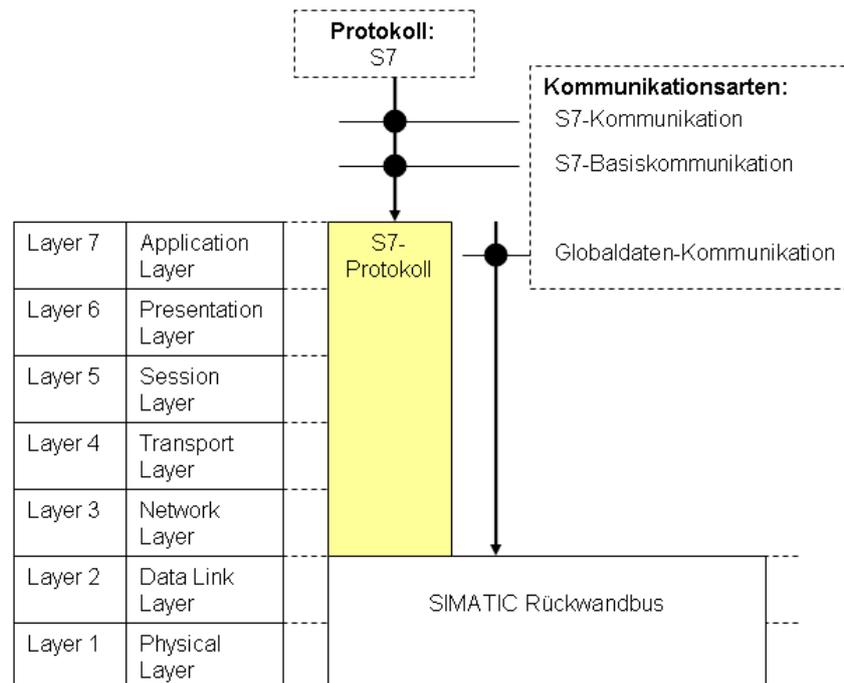
Merkmale

- Herstellerspezifische Schnittstelle
- nur möglich bei SIMATIC S7-400 (Multicomputing, Kapitel 4.5)
- für SIMATIC spezifische Kommunikationsarten:
Globaldaten-Kommunikation, S7-Kommunikation

ISO/OSI-Referenzmodell

Das folgende Bild zeigt alle Protokolle und Kommunikationsarten, die von SIMATIC Controllern unterstützt werden.

Abbildung 12-1



13 Serielle Schnittstelle (PtP)

Einführung

Die Kommunikation über eine „Serielle Schnittstelle“ bietet eine einfache Möglichkeit zwischen zwei Kommunikationspartnern Daten auszutauschen.

SIMATIC Controller können über die „Serielle Schnittstelle“ mit unterschiedlichen Partnern kommunizieren:

- einfache Geräte, wie Drucker, Barcodeleser
- Antriebe (USS Protokoll, ...)
- SIMATIC Controller, Fremde Controller

Anzahl Teilnehmer

Im Allgemeinen nehmen in Summe genau zwei Kommunikationspartner an der Kommunikation teil (Punkt zu Punkt Kopplung).

Bei RS 422/485 sind jedoch auch mehr als zwei Kommunikationspartner möglich (Mehrpunkt Kopplung).

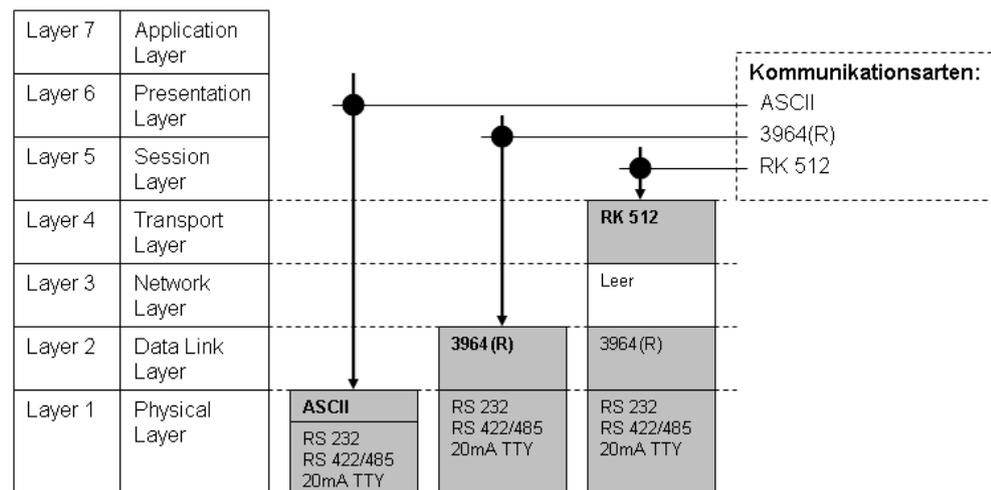
Merkmale

- Kommunikation ist meistens offen gelegt
- Kommunikation über genormte Schnittstellen (Physik):
RS 232C, RS 422/485, 20mA TTY

ISO/OSI-Referenzmodell

Das folgende Bild zeigt alle Protokolle und Kommunikationsarten, die von SIMATIC Controllern unterstützt werden.

Abbildung 13-1



14 Gegenüberstellung der Medien

Die folgende Tabelle zeigt, wie sich die Medien grundsätzlich von einander abgrenzen. Die Gegenüberstellung ist eine starke Vergrößerung. In Einzelfällen können Abweichungen von den eingetragenen Werten auftreten. In konkreten Anwendungsfällen sind deswegen die entsprechenden Handbücher zu Rate zu ziehen.

Tabelle 14-1

Medium		Übertragungsrate (von ...bis)	maximale Anzahl Partner	maximale Abstand zweier Partner	maximale Anzahl Daten pro Auftrag	Routing möglich?	Redundanter Aufbau mög- lich?
Netz	PROFINET/ Industrial Ethernet	PN: 10 bis 100 MBit/s IE: 1 GBit/s	über 1000 (*1)	elektrisch: 100 m (*2) optisch: 26 km (*2) Radio-Wellen: 100 m (*2)	64 KByte	ja (nicht PN IO)	ja
	PROFIBUS	9,6 KBit/s bis 12 MBit/s	126	elektrisch: 1 km (ohne Repeater) bzw. 10 km (mit Repeater) optisch: 1875 km (mit OLM)	64 KByte	nein	ja
	MPI	187,5 KBit/s bis 12 MBit/s	126	elektrisch: 50 m	64 KByte	nein	nein
SIMATIC Rückwandbus (nur bei S7-400)		---	4 CPUs in der SIMATIC Station	---	64 KByte	---	---
Serielle Schnittstelle (ASCII, 3964(R), RK 512)		110 Bit/s bis 115,2 KBit/s (20mA-TTY: bis 19,2 kBit/s)	Punkt zu Punkt: 2 Mehrpunkt: 32	RS232C: 15 m RS422/485: 1200 m 20mA-TTY: 1000 m	4 KByte	---	---

Erläuterungen zur Tabelle:

(*1): In einem PROFINET IO Netz ist die maximale Anzahl Partner auf 512 beschränkt.

(*2): maximale Segmentlänge

15 Schnittstellen der SIMATIC Familien

Die Tabelle gibt einen Überblick, welche SIMATIC Familien über welche Medien kommunizieren können (CPU-CPU Kommunikation).
Bei PROFIBUS und PROFINET/Industrial (PN/IE) wird zusätzlich angegeben, welche Funktionalität die Schnittstelle annehmen kann.

Tabelle 15-1

SIMATIC Controller	SIMATIC Familie	MPI	PROFIBUS		PN/IE			SIMATIC Rückwandbus	Serielle Schnittstelle		
				Funktionalität			Funktionalität				
				DP-Master	DP-Slave		PROFINET IO Controller			PROFINET IO Device	PROFINET CBA
Modulare Controller	ET 200 CPU	X	X	X	X	X	X	X	X	---	X
	S7-300	X	X	X	X	X	X	X	X	X	X
	S7-400	X	X	X	X	X	X	X	X	X	X
	S7-1200	---	---	---	---	X	---	---	---	---	X
Embedded Automation	S7-mEC (mit WinAC RTX)	---	X	X	---	X	X	---	X	---	X
	Box PC (mit WinAC RTX)	---	X	X	---	X	X	---	X	---	X
	Panel PC (mit WinAC RTX)	---	X	X	---	X	X	---	X	---	X
	WinAC MP (mit WinAC MP)	---	X	X	---	X	---	---	---	---	X
PC-based Controller	WinAC RTX (Installiert auf PC)	---	X	X	---	X	X	---	X	---	X

Bedeutung der Einträge in der Tabelle:

„x“: Schnittstelle / Funktionalität vorhanden

„---“ Schnittstelle / Funktionalität nicht vorhanden

Hinweis

Ein „x“ bedeutet nicht, dass alle Schnittstellen einer SIMATIC Familie diese Funktionalität haben.

Ein „x“ bedeutet, dass es mindestens eine Schnittstelle gibt, welche diese Funktionalität aufweist.

16 Informationen Teil 1

In der Tabelle sind Verweise auf Informationen zu Themen aus Teil 1 zu finden.

Alle Verweise /x/ sind zentral im Kapitel 47 hinterlegt. Dort sind auch die entsprechenden Links ins Internet zu finden.

Tabelle 16-1

Verweis	Titel / Inhalt	Informationen zu
---	STEP 7 Online Hilfe: „Projektieren von Verbindungen und Datenaustausch“	Verbindungen Verbindungsressourcen
/0/	Service & Support Portal: FAQs, Handbücher (SIMATIC CPU, CP)	Kommunikationsarten Datenkonsistenz
/3/	SIMATIC / Kommunikation mit SIMATIC Systemhandbuch	
/6/	SIMATIC System- und Standardfunktionen für S7-300/400, Referenzhandbuch	
/15/	SIMATIC / Hardware konfigurieren und Verbindungen projektieren mit STEP 7, Handbuch	
/1/	SIMATIC Controller / Die innovative Lösung für alle Au- tomatisierungsaufgaben. Broschüre	SIMATIC Controller
/4/	Katalog ST 70 / Produkte für Totally Integrated Automa- tion und Micro Automation	
/2/	SIMATIC NET / Industrielle Kommunikation Broschüre	Medien
/3/	SIMATIC / Kommunikation mit SIMATIC Systemhandbuch	
/5/	Katalog IK PI / Industrielle Kommunikation	

TEIL 2: Auswahlhilfe

Teil 2 ist der zentrale Teil des Dokumentes

Übersichtliche Darstellung aller Möglichkeiten
für eine CPU-CPU Kommunikation mit SIMATIC Controllern

TEIL 2: Gliederung und Inhalt

Tabelle 16-2

Kapitel	Gliederung	Inhalt
17	Vorbemerkungen	Erläuterungen zu den verwendeten Tabellen
18	Sprungverteiler	Pro Medium existiert eine Seite mit Querverweisen zu den zentralen Kapiteln der Dokumentation. Damit können gesuchte Informationen schnell ausgewählt werden.
19	Auswahlhilfe PN/IE	Pro Medium wird beschrieben: <ul style="list-style-type: none"> • Alle Schnittstellen (CPU, CP) und Kommunikationsarten pro SIMATIC Familie (Tabelle Schnittstellen). • Alle Möglichkeiten, wie SIMATIC Familien miteinander kommunizieren können (Tabelle Kombinationen) • Gegenüberstellung aller Kommunikationsarten (Tabelle Kommunikationsarten-Kompakt)
20	Auswahlhilfe PB	
21	Auswahlhilfe MPI	
22	Auswahlhilfe SIMATIC Rückwandbus	
23	Auswahlhilfe Serielle Schnittstelle	Pro SIMATIC Familie wird beschrieben: <ul style="list-style-type: none"> • Alle Schnittstellen (CPU, CP) und Kommunikationsarten (ASCII, 3964(R), ...) • Eigenschaften der Kommunikationsarten
24	Informationen	Hinweis auf weitere Informationen

Erläuterungen zur Tabelle

Medium:

- Netz: PN/IE, PB, MPI
- SIMATIC Rückwandbus

SIMATIC Familien:

- Modulare Controller: ET 200 CPU, S7-300, S7-400, S7-1200
- Embedded Automation: S7-mEC, Box PC, Panel PC, WinAC MP
- PC-based Controller: WinAC RTX

Kommunikationsarten:

- SIMATIC spezifisch
- Offener Standard

17 Vorbemerkungen

Im Folgenden wird der Aufbau der im Teil 2 verwendeten Tabellen erläutert:

- Tabelle Schnittstellen
- Tabelle Kombinationen
- Tabelle Kommunikationsarten-Kompakt

Vorab wird beschrieben, nach welcher Systematik im Dokument die zahlreichen Kombinationsmöglichkeiten der SIMATIC Familien dargestellt werden.

17.1 Übersicht aller Kombinationen

17.1.1 Begriffe

Definitionen

Ein Paar:

- SIMATIC Familie x, SIMATIC Familie y

Eine Kombination:

- Controller 1 / Controller 2

Ein Paar ergibt zwei Kombinationen:

- Controller 1 / Controller 2: SIMATIC Familie x / SIMATIC Familie y
- Controller 1 / Controller 2: SIMATIC Familie y / SIMATIC Familie x

Beispiel

Paar:

- ET 200 CPU, S7-300

Kombinationen:

- ET 200 CPU / S7-300
- S7-300 / ET 200 CPU

17.1.2 Symmetrie der Kombinationen

Die folgende Tabelle zeigt alle Kombinationen von SIMATIC Familien (unabhängig vom Medium). Jeder Eintrag in der Tabelle (Bezeichnung mit „x.y“) ergibt eine definierte Kombination von zwei SIMATIC Familien.

Tabelle 17-1

Controller 2		Controller 1								
		Modular				Embedded				PC-based
		ET 200 CPU	S7-300	S7-400	S7-1200	S7-mEC	Box PC	Panel PC	WinAC MP	WinAC RTX
Modular	ET 200 CPU	1.1	2.1	3.1	4.1	5.1	6.1	7.1	8.1	9.1
	S7-300	1.2	2.2	3.2	4.2	5.2	6.2	7.2	8.2	9.2
	S7-400	1.3	2.3	3.3	4.3	5.3	6.3	7.3	8.3	9.3
	S7-1200	1.4	2.4	3.4	4.4	5.4	6.4	7.4	8.4	9.4
Embedded	S7-mEC	1.5	2.5	3.5	4.5	5.5	6.5	7.5	8.5	9.5
	Box PC	1.6	2.6	3.6	4.6	5.6	6.6	7.6	8.6	9.6
	Panel PC	1.7	2.7	3.7	4.7	5.7	6.7	7.7	8.7	9.7
	WinAC MP	1.8	2.8	3.8	4.8	5.8	6.8	7.8	8.8	9.8
PC-based	WinAC RTX	1.9	2.9	3.9	4.9	5.9	6.9	7.9	8.9	9.9

Für jedes Paar (SIMATIC Familie x, SIMATIC Familie y) gibt es zwei Kombinationen (Einträge) in der Tabelle:

- im orange hinterlegten Bereich der Tabelle
- im grau hinterlegten Bereich der Tabelle

Beispiel

Paar:

- ET 200 CPU, S7-300

Kombination 1.2:

- ET 200 CPU / S7-300

Kombination 2.1:

- S7-300 / ET 200 CPU

Das folgende Kapitel zeigt, wie diese Symmetrie im Dokument berücksichtigt wird.

17.1.3 Umsetzung im Dokument

Für jedes Paar zweier SIMATIC Familien (x,y) gibt es im Dokument zwei Kapitel:

- ein Kapitel für die Kombination x/y (Kapitel x/y)
- ein Kapitel für die Kombination y/x (Kapitel y/x)

Die Beschreibung der möglichen Kommunikationsarten für ein Paar erfolgt natürlich nur in einem Kapitel (Kapitel x/y). In dem anderen Kapitel (Kapitel y/x) steht nur ein Verweis auf dieses Kapitel (Kapitel x/y).

Diese Vorgehensweise hat Auswirkungen auf die Gliederung des Dokumentes und auf die Darstellung der Kombinationen (Sprungverteiler).

Gliederung des Dokumentes (Inhaltsverzeichnis)

Wenn in einem Kapitel (Kapitel y/x) nur ein Verweis auf ein anderes Kapitel (Kapitel x/y) zu finden ist, dann steht die Überschrift dieses Kapitels (Kapitel y/x) in Klammern.

Beispiel

Betrachtet wird das Paar (ET 200 CPU, S7-300), für das Medium PN/IE.

Die folgende Abbildung zeigt einen Auszug aus dem Inhaltsverzeichnis.

Abbildung 17-1

19.3	PN/IE: Controller 1 = ET 200 CPU ..
19.3.1	ET 200 CPU / ET 200 CPU
19.3.2	ET 200 CPU / S7-300
19.3.3	ET 200 CPU / S7-400
19.3.4	ET 200 CPU / S7-1200
19.3.5	ET 200 CPU / S7-mEC
19.3.6	ET 200 CPU / Box PC
19.3.7	ET 200 CPU / Panel PC
19.3.8	ET 200 CPU / WinAC MP
19.3.9	ET 200 CPU / WinAC RTX
19.4	PN/IE: Controller 1 = S7-300
19.4.1	(S7-300 / ET 200 CPU)
19.4.2	S7-300 / S7-300
19.4.3	S7-300 / S7-400
19.4.4	S7-300 / S7-1200
19.4.5	S7-300 / S7-mEC
19.4.6	S7-300 / Box PC
19.4.7	S7-300 / Panel PC
19.4.8	S7-300 / WinAC MP
19.4.9	S7-300 / WinAC RTX

Das Paar (ET 200 CPU, S7-300) wird im Kapitel 19.3.2 beschrieben

Im Kapitel 19.4.1 steht ein Verweis auf das Kapitel 19.3.2

Kombinationen (Sprungverteiler)

Im Dokument werden die Kombinationen in Tabellen (Sprungverteiler) dargestellt. Für jede Kombination wird das Kapitel eingetragen, in dem diese Kombination konkret beschrieben wird.

Beispiel

Betrachtet wird das Paar (ET 200 CPU, S7-300), für das Medium PN/IE.

Die folgende Abbildung zeigt den Sprungverteiler für PN/IE.

Abbildung 17-2

Controller 2		Controller 1								
		Modular				Embedded				PC-based
		ET 200 CPU	S7-300	S7-400	S7-1200	S7-mEC	Box PC	Panel PC	WinAC MP	WinAC RTX
Modular	ET 200 CPU	19.3.1	19.3.2	19.3.3	19.3.4	19.3.5	19.3.6	19.3.7	19.3.8	19.3.9
	S7-300	19.3.2	19.4.2	19.4.3	19.4.4	19.4.5	19.4.6	19.4.7	19.4.8	19.4.9
	S7-400	19.3.3	19.4.3	19.5.3	19.5.4	19.5.5	19.5.6	19.5.7	19.5.8	19.5.9
	S7-1200	19.3.4	19.4.4	19.5.4	19.6.4	19.6.5	19.6.6	19.6.7	19.6.8	19.6.9
Embedded	S7-mEC	19.3.5	19.4.5	19.5.5	19.6.5	19.7.5	19.7.6	19.7.7	19.7.8	19.7.9
	Box PC	19.3.6	19.4.6	19.5.6	19.6.6	19.7.6	19.8.6	19.8.7	19.8.8	19.8.9
	Panel PC	19.3.7	19.4.7	19.5.7	19.6.7	19.7.7	19.8.7	19.9.7	19.9.8	19.9.9
	WinAC MP	19.3.8	19.4.8	19.5.8	19.6.8	19.7.8	19.8.8	19.9.8	19.10.8	19.10.9
PC-based	WinAC RTX	19.3.9	19.4.9	19.5.9	19.6.9	19.7.9	19.8.9	19.9.9	19.10.9	19.11.9

Für das Paar gibt es zwei Einträge (Controller 1 / Controller 2) in der Tabelle:

- ET 200 CPU / S7-300 (grün umrandet)
- S7-300 / ET 200 CPU (rot umrandet)

Beide Einträge enthalten die identische Kapitelnummer (19.3.2).

17.2 Tabelle Schnittstellen

17.2.1 Zweck der Tabelle

Mit der Tabelle Schnittstellen wird kompakt dargestellt, welche Schnittstellen die SIMATIC Familien zur Verfügung stellen. Für jedes Medium und jede SIMATIC Familie gibt es eine Tabelle. Diese Tabellen sind die Basis für die Tabellen Kombinationen (Kapitel 17.3).

Die Tabelle beantwortet folgende Fragen:

- Welche Schnittstellen (CPU, CP) stehen zur Verfügung?
- Welche Kommunikationsarten sind möglich?
- Welche Besonderheiten gibt es?

17.2.2 Aufbau der Tabelle

Im Folgenden wird der Aufbau der Tabellen beschrieben.

Medium PN/IE

Der Aufbau wird an Hand eines konkreten Beispiels erläutert:

- S7-300 an PN/IE

Das Bild zeigt die zugehörige Tabelle Schnittstellen (Tabelle 19-2).

1 Abbildung 17-3

Controller an PN/IE: S7-300		Kommunikationsart				
		SIMATIC spezifisch		Offener Standard		
		S7-Kommunikation		Offene-Kommunikation	PN-Kommunikation	
				IOc	IOD	CBA
CPU	alle mit Schnittstelle: PN (*2)	(1) (IoT) (14) (IoT)	(3) (IoT, TCP, UDP) (13) (IoT, TCP) (*1)	(2)	(2)	x
CP	343-1 Lean	PUT, GET Server (IoT)	(8) (IoT, TCP, UDP) (13) (IoT, TCP)	---	(6)	---
	343-1	(1) (IoT, ISO)	(8) (IoT, TCP, UDP, ISO) (13) (IoT, TCP, ISO)	(6)	(6)	---
	343-1 Advanced	(1) (IoT, ISO)	(8) (IoT, TCP, UDP, ISO) (13) (IoT, TCP, ISO)	(6)	(6)	x
	343-1 ERPC (*3)	(1) (IoT)	(8) (IoT, TCP, UDP) (13) (TCP)	---	---	---

2

1a

1b

1c

2a

2b

2c

3

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET	(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
(3) TSEND/TRCV, TUSEND/TURCV	(6) PNIO_SEND, PNIO_RECV
(8) AG_SEND/AG_RECV	(13) Server für Fetch/Write
(14) USEND_E / URCV_E, PUT_E, GET_E	

Die Tabelle besteht aus mehreren Bereichen, die im Folgenden erläutert werden.

Bereich 1: Schnittstellen Controller

Hier werden die Eigenschaften der SIMATIC Familie beschrieben:

Tabelle 17-2

	Bedeutung	Beispiel
1a	Bezeichnung des Mediums und der SIMATIC Familie	S7-300 an PN/IE
1b	Art der Schnittstelle: Integrierte Schnittstelle einer CPU, oder Schnittstelle eines CP	Es gibt beide Varianten: CPU und CP
1c	Bezeichnung der Baugruppen	CPU: siehe Bild CP: siehe Bild

Bereich 2: Kommunikationsarten

Hier werden die Kommunikationsmöglichkeiten beschrieben:

Tabelle 17-3

	Bedeutung	Beispiel
2a	Bezeichnung der Kommunikationsarten, unterteilt in die Klassen „SIMATIC spezifisch“ und „Offener Standard“.	S7-Kommunikation Offene-Kommunikation PN-Kommunikation
2b	Funktionalität der Schnittstelle	PROFINET IO Controller, PROFINET IO Device, PROFINET CBA
2c	Hier wird eingetragen, welche Möglichkeiten zur Kommunikation bei den Baugruppen vorhanden sind. Die möglichen Kommunikationsbausteine werden entweder direkt eingetragen, oder über einen Index repräsentiert. Der Index wird im Bereich 3 erläutert.	direkter Eintrag: • PUT, GET Server (IoT) Index: • (1), (2), ...
	Die möglichen Protokolle werden in Klammern angegeben. Kann ein Controller <u>nur</u> als Server eingesetzt werden, dann wird dies ebenfalls eingetragen.	Der CP 343-1 Lean kann über PUT und GET kommunizieren. Die Baugruppe ist dabei Server. Als Protokolle sind IoT und ISO möglich.
	Bedeutung der Einträge: --- Kommunikation nicht möglich x Kommunikation möglich	

Bereich 3: Kommunikationsbausteine zum Index (x)

Hier werden die zu einem Index möglichen K-Bausteine aufgeführt.

Medium PB

Der Aufbau wird an Hand eines konkreten Beispiels erläutert:

- S7-300 an PB

Das Bild zeigt einen Ausschnitt aus der Tabelle Schnittstellen (Tabelle 20-2).

Abbildung 17-4

Controller an PB: S7-300		Funktionalität der Schnittstelle		Kommunikationsart	
				SIMATIC spezifisch	
				S7-Basiskommunikation	S7-Kommunikation
CPU	alle mit Schnittstelle: DP, MPI/DP (*2)	DP-Master		L_PUT, L_GET, Client (S7)	PUT, GET, Server (S7)
		DP-Slave (*3)	aktiv	L_PUT, L_GET, Server (S7)	PUT, GET, Server (S7)
			passiv	L_PUT, L_GET, Server (S7)	---
CP	342-5 (*1)	kein DP-Betrieb		---	(1) (S7)
		DP-Master		---	(1) (S7)
	DP-Slave (*3)	aktiv	---	PUT, GET, Server (S7)	
		passiv	---	---	
343-5	kein DP-Betrieb		---	PUT, GET, Server (S7)	

Der Aufbau der Tabelle ist genau so wie beim Medium PN/IE (siehe vorher), mit einer Ausnahme:

- Funktionalität der Schnittstelle (Bereich 2b)

Bereich 4: Funktionalität der Schnittstelle

Hier werden die möglichen Funktionalitäten der Schnittstelle angegeben. Je nach Funktionalität ergeben sich unterschiedliche Kommunikationsarten.

Der Bereich 2b der Tabelle für PB entspricht dem Bereich 2b der Tabelle für PN/IE.

17.2.3 Abkürzungen und Indizes

Eine Übersicht der verwendeten Abkürzungen und Indizes ist im Kapitel 17.5 zu finden.

17.3 Tabelle Kombinationen

17.3.1 Zweck der Tabelle

Mit der Tabelle Kombinationen wird kompakt dargestellt, wie SIMATIC Controller miteinander kommunizieren können. Für jedes Medium und für jede Kombination zweier SIMATIC Familien gibt es im Dokument eine Tabelle.

Die Tabelle beantwortet folgende Fragen:

- Welche Schnittstellen (CPU, CP) stehen zur Verfügung?
- Welche Kommunikationsarten sind möglich?
- Welche Besonderheiten bezüglich der Kommunikation gibt es?

17.3.2 Aufbau der Tabelle

Der Aufbau der Tabelle wird an einem vereinfachten (nicht realen) Beispiel beschrieben.

Randbedingungen:

- Controller 1 ist eine SIMATIC Familie mit einer CPU Schnittstelle (CPU11) und zwei CP Schnittstellen (CP11, CP12)
- Controller 2 ist eine SIMATIC Familie mit einer CPU Schnittstelle (CPU21) und zwei CP Schnittstellen (CP21, CP22)
- Es wird angenommen, dass zwei Kommunikationsarten möglich sind: S7-Kommunikation (S7) und Offene-Kommunikation (OC)

Unter obigen Randbedingungen sieht die Tabelle Kombinationen so aus:

Tabelle 17-4

Controller 2: S7-400			Controller 1: S7-300 an PN/IE					
			CPU		CP			
			CPU11	CP11	CP12			
			IOC, IOD	IOD	IOD			
			S7	OC	S7	OC	S7	OC
CPU	CPU21	IOC	(1)	(3)	(21)	(8) / (3)	(1)	(8) / (3)
	CP	CP21	(1)	(3) / (8)	(21)	(8)	(1)	(8)
		CP22	(1)	(3) / (8)	(21)	(8)	(1)	(8)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (3) TSEND/TRCV, TUSEND/TURCV
- (8) AG_SEND/AG_RECV
- (21) Controller 1 ist Server (für PUT, GET)

Es folgt eine Erläuterung der einzelnen Bereiche der Tabelle.

Überblick

Die Tabelle zeigt alle Kombinationen der Baugruppen von Controller 1 (Bereich 1) und Controller 2 (Bereich 2):

- CPU11 mit CPU21, CP21, CP22
- CP11 mit CPU21, CP21, CP22
- CP12 mit CPU21, CP21, CP22

Für jede Kombination wird im Bereich 4 angegeben, welche Kommunikationsarten (Bereich 3) möglich sind.

Die erforderlichen Kommunikationsbausteine sind im Bereich 5 aufgeführt.

Bereich 1 (gelb)

Bereich 1 zeigt alle kommunikationsfähigen Baugruppen von Controller 1.

Eine Spalte repräsentiert eine Baugruppe. Baugruppen mit gleichem Verhalten werden zusammengefasst.

Die Baugruppen sind gruppiert nach dem Typ der Schnittstelle (Zeile 2):

- CPU Schnittstellen: CPU11 (Zeile 3)
- CP Schnittstellen: CP11, CP12 (Zeile 3)

Die Funktionalität der Schnittstelle ist in Zeile 4 eingetragen:

- CPU11: IOC, IOD (PROFINET IO Controller, PROFINET IO Device)
- CP11: IOD (PROFINET IO Device)

Bereich 2 (blau)

Bereich 2 zeigt alle kommunikationsfähigen Baugruppen von Controller 2.

Eine Zeile repräsentiert eine Baugruppe. Baugruppen mit gleichem Verhalten werden zusammengefasst.

Die Baugruppen sind gruppiert nach dem Typ der Schnittstelle (Spalte 1):

- CPU Schnittstellen: CPU21 (Spalte 2)
- CP Schnittstellen: CP21, CP22 (Spalte 2)

Die Funktionalität der Schnittstelle ist in Spalte 3 eingetragen:

- CPU21: IOC (PROFINET IO Controller)
- CP21: IOD (PROFINET IO Device)

Bereich 3 (grau)

Im Bereich 3 werden die für die beiden Controller prinzipiell möglichen Kommunikationsarten aufgeführt. Sie werden für jede Baugruppe (Spalte) von Controller 1 (Bereiches 1) wiederholt:

- S7 (S7-Kommunikation)
- OC (Offene Kommunikation)

Bereich 4 (weiß)

Jede Zelle im Bereich 4 repräsentiert eine Kombination von zwei Baugruppen. Gleichzeitig ist jede Zelle einer Kommunikationsart (Bereich 3) zugeordnet.

Bereich 4 liefert pro Kombination folgende Informationen:

- Mögliche Kommunikationsarten (Bereich 3)
- Zugehörige Kommunikationsbausteine, repräsentiert über einen Index ((x))

Die Einträge in den Zellen haben folgende Bedeutung:

Tabelle 17-5

Eintrag in Zelle	Kommunikationsart entsprechend Bereich 3 möglich?	Welche Kommunikationsbausteine können eingesetzt werden?
(1)	ja	Siehe Bereich 5: Index (1)
(1)+(2)	ja	Siehe Bereich 5: (1) und (2)
(1) / (2)	ja	Controller 1 mit (1) / Controller 2 mit (2)
(1) / (2)+(3)	ja	Zwei Fälle sind möglich: <ul style="list-style-type: none"> • Controller 1 mit (1) / Controller 2 mit (2) • Controller 1 mit (1) / Controller 2 mit (3)
x	ja	Gilt nur für CBA (dort sind keine Kommunikationsbausteine erforderlich).
---	nein	entfällt

Bereich 5 (Kommunikationsbausteine)

Im Bereich 5 werden die Indizes (x) aus Bereich 4 erläutert. Zu jedem Index wird angegeben, welche Kommunikationsbausteine möglich sind, und welche Besonderheiten zu beachten sind.

Zusammenfassung

Jede Zelle im Bereich 4 liefert folgende Informationen:

- Baugruppe x (Bereich 1) kann mit Baugruppe y (Bereich 2) über die Kommunikationsart z (Bereich 3) kommunizieren.
- Die möglichen Kommunikationsbausteine sind im Bereich 5 beschrieben.

Beispiel

Die rot umrandete Zelle in Tabelle 17-4 bedeutet:

Kombination:

- CPU11 / CP21

Kommunikationsart:

- Offene-Kommunikation

Kommunikationsbausteine:

- Für Controller 1: TSEND/TRCV, TUSEND/TURCV (T-Bausteine)
- Für Controller 2: AG_SEND/AG_RECV (Send/Receive-Bausteine)

17.3.3 Reales Beispiel

Der Umgang mit der Tabelle wird jetzt an einem realen Beispiel erläutert:

- Kommunikation über PN/IE
- Controller 1: ET 200 CPU
- Controller 2: S7-300

Das Bild zeigt die zugehörige Tabelle Kombinationen (Tabelle 19-14).

Abbildung 17-5

Controller 2: S7-300			Controller 1: ET 200 CPU an PN/IE			
			CPU			
			IM 151-8(F) PN/DP, IM 154-8(F) PN/DP CPU			
			IOC, IOD			
			S7	OC	PN (*1)	
					PNIO	CBA
CPU	alle mit Schnittstelle: PN	IOC, IOD	(1)+(14)	(3)	(2)	x
CP	343-1 Lean	IOD	(22)	(3) / (8)	(2) / (6)	---
	343-1	IOC, IOD	(1)	(3) / (8)	(2) / (6)	---
	343-1 Advanced	IOC, IOD	(1)	(3) / (8)	(2) / (6)	x
	343-1 ERPC	---	(1)	(3) / (8)	---	---

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (3) TSEND/TRCV, TUSEND/TURCV
- (6) PNIO_SEND, PNIO_RECV
- (8) AG_SEND/AG_RECV
- (14) USEND_E/URCV_E, PUT_E, GET_E
- (22) Controller 2 ist Server (für PUT, GET)



Beispiele (in obiger Tabelle rot umrandet):

Tabelle 17-6

	Controller 1		Controller 2		Kommunikationsart	Kommunikationsbausteine (Index)
	Schnittstelle	Funktionalität	Schnittstelle	Funktionalität		
1	IM151-8, IM154-8	IOC, IOD	343- 1	IOC, IOD	S7	Beide Controller: (1)
2	IM151-8, IM154-8	IOC, IOD	343-1 ERPC	---	OC	Controller 1: (3) Controller 2: (8)
3	IM151-8, IM154-8	IOC, IOD	343-1 Lean	IOD	PN: PNIO	Controller 1: (2) Controller 2: (6)
4	IM151-8, IM154-8	IOC, IOD	S7-300 CPU PN	IOC, IOD	PN: CBA	---

17.3.4 Abkürzungen und Indizes

Eine Übersicht der Abkürzungen und Indizes ist im Kapitel 17.5 zu finden.

17.4 Tabelle Kommunikationsarten-Kompakt

17.4.1 Zweck der Tabelle

Die Tabelle Kommunikationsarten-Kompakt beantwortet für jedes Medium (PN/IE, PB, MPI) die folgenden Fragen:

- Welche Kommunikationsarten gibt es pro Medium?
- Wie unterscheiden sie sich im Wesentlichen?

Die Tabelle Kommunikationsarten-Kompakt ist eine Zusammenfassung (Verdichtung) der Tabellen Kommunikationsarten-Detail. Nicht jede SIMATIC Familie bzw. Baugruppe erfüllt alle in der Tabelle Kommunikationsarten-Kompakt angegebenen Werte. Die Werte sind als Obermengen zu betrachten.

17.4.2 Aufbau der Tabelle

Beispiel: Ausschnitt aus der Tabelle für PN/IE (Tabelle 19-58).

Abbildung 17-6

	SIMATIC spezifisch		Offener Standard	
	S7-Kommunikation		Offene-Kommunikation	
			Send/Receive-Bausteine	T-Bausteine
Protokolle	ISO, IoT		ISO, IoT, TCP, UDP	
Schnittstellen	ISO: CP IoT: CPU, CP		CP CPU, CP	
Kommunikationsbausteine (max. Daten)	BSEND (≤ 64 KByte) USEND, USEND_E (≥ 160 Byte) PUT, PUT_E, GET, GET_E (≥ 160 Byte)		AG_xSEND (ISO, IoT, TCP ≤ 8 KByte) (UDP ≤ 2 KByte) ----- Server für FETCH, WRITE (nicht bei UDP)	
remote Quittierung	BSEND: Applikation USEND: Transport PUT, GET: Applikation		ISO, IoT, TCP: Transport UDP: keine	
Routingfähig?	ISO: nein IoT: ja		ISO: nein sonst: ja	
Verbindungen?	ja		UDP: nein sonst: ja	

Die Tabelle besteht aus zwei Bereichen, die im Folgenden erläutert werden.

Bereich 1: Kommunikationsarten

Bezeichnung der Kommunikationsarten, unterteilt in die Klassen „SIMATIC spezifisch“ und „Offener Standard“.

Bereich 2: Kriterien

Die Kriterien werden im Kapitel 25.2.3 beschrieben.

Hinweis

Die Kriterien der Tabelle Kommunikationsarten-Kompakt sind eine Untermenge der Tabelle Kommunikationsarten-Detail.

17.5 Übersicht der Abkürzungen und Indizes

17.5.1 Abkürzungen

In der Tabelle Schnittstellen und der Tabelle Kombinationen werden folgende Abkürzungen verwendet:

Tabelle 17-7

Abkürzung	Bedeutung
CBA	PROFINET CBA (Component Based Automation)
Cx	Controller x
Cx Server	Controller x ist Server
DP	DP-Kommunikation
GD	Globaldaten-Kommunikation
IOC	IO Controller
IOD	IO Device
Master	DP-Master
OC	Offene-Kommunikation „Offene-Kommunikation“ steht für die beiden Kommunikationsarten (*1): <ul style="list-style-type: none"> • Offene-Kommunikation mit T-Bausteinen • Offene-Kommunikation mit Send/Receive-Bausteinen
PN	PN-Kommunikation
PNIO	PROFINET IO
S/R	Send/Receive
S7	S7-Kommunikation
S7 Basis S7 B	S7-Basiskommunikation
Slave	DP-Slave

(*1): Gegenüberstellung

Tabelle 17-8

	Offene Kommunikation	
	Offene-Kommunikation mit T-Bausteinen	Offene-Kommunikation mit Send/Receive-Bausteinen
Medium	PN/IE	PN/IE, PB
Kommunikationsbausteine	TSEND / TRCV TUSEND / TURCV	AG_SEND / AG_RECV AG_SSEND / AG_SRECV AG_LSEND / AG_LRECV
Schnittstelle	CPU, CP	CP
Kommunikation mit Controllern fremder Hersteller	ja	ja

17.5.2 Übersicht der Indizes

In der Tabelle Schnittstellen und der Tabelle Kombinationen werden folgende Indizes verwendet:

Tabelle 17-9

(x)	Zur Verfügung stehende Kommunikationsbausteine
(1)	USEND/URCV, BSEND/BRCV, PUT, GET
(2)	Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
(3)	TSEND/TRCV, TUSEND/TURCV
(4)	TSEND/TRCV
(5)	TSEND_C/TRCV_C
(6)	PNIO_SEND, PNIO_RECV
(7)	DP_SEND, DP_RECV
(8)	AG_SEND/AG_RECV
(9)	AG_SEND/AG_RECV, AG_LSEND/AG_LRECV, AG_SSEND/AG_SRECV
(10)	READ, WRITE, REPORT
(11)	X_SEND/X_RCV, X_PUT, X_GET
(12)	GD_SND/GD_RCV
(13)	Server für FETCH/WRITE
(14)	USEND_E/URCV_E, PUT_E, GET_E

Folgende Indizes werden nur in der Tabelle Kombinationen verwendet:

Tabelle 17-10

(x)	Zur Verfügung stehende Kommunikationsbausteine
S7-Kommunikation	
(21)	Controller 1 ist Server (für PUT, GET)
(22)	Controller 2 ist Server (für PUT, GET)
(31)	Controller 1 ist Server (für PUT_E, GET_E)
(32)	Controller 2 ist Server (für PUT_E, GET_E)
S7-Basiskommunikation	
(61)	Controller 1 ist Server (für I_PUT, I_GET)
(62)	Controller 2 ist Server (für I_PUT, I_GET)

18 ***** Sprungverteiler *****

18.1 Netz PN/IE

Übersicht Schnittstellen und Kommunikationsarten

Tabelle 18-1

Familie		Kapitel
Modulare Controller	ET 200 CPU	19.2.1
	S7-300	19.2.2
	S7-400	19.2.3
	S7-1200	19.2.4
Embedded Controller	S7-mEC	19.2.5
	Box PC	19.2.6
	Panel PC	19.2.7
	WinAC MP	19.2.8
PC-based Controller	WinAC RTX	19.2.9

Kombinationen Controller 1 / Controller 2

Tabelle 18-2

Controller 2		Controller 1								
		Modular				Embedded				PC-based
		ET 200 CPU	S7-300	S7-400	S7-1200	S7-mEC	Box PC	Panel PC	WinAC MP	WinAC RTX
Modular	ET 200 CPU	19.3.1	19.3.2	19.3.3	19.3.4	19.3.5	19.3.6	19.3.7	19.3.8	19.3.9
	S7-300	19.3.2	19.4.2	19.4.3	19.4.4	19.4.5	19.4.6	19.4.7	19.4.8	19.4.9
	S7-400	19.3.3	19.4.3	19.5.3	19.5.4	19.5.5	19.5.6	19.5.7	19.5.8	19.5.9
	S7-1200	19.3.4	19.4.4	19.5.4	19.6.4	19.6.5	19.6.6	19.6.7	19.6.8	19.6.9
Embedded	S7-mEC	19.3.5	19.4.5	19.5.5	19.6.5	19.7.5	19.7.6	19.7.7	19.7.8	19.7.9
	Box PC	19.3.6	19.4.6	19.5.6	19.6.6	19.7.6	19.8.6	19.8.7	19.8.8	19.8.9
	Panel PC	19.3.7	19.4.7	19.5.7	19.6.7	19.7.7	19.8.7	19.9.7	19.9.8	19.9.9
	WinAC MP	19.3.8	19.4.8	19.5.8	19.6.8	19.7.8	19.8.8	19.9.8	19.10.8	19.10.9
PC-based	WinAC RTX	19.3.9	19.4.9	19.5.9	19.6.9	19.7.9	19.8.9	19.9.9	19.10.9	19.11.9

Kommunikationsarten

Tabelle 18-3

Kommunikationsart		Kapitel
Alle Kommunikationsarten (Tabelle Kompakt)		19.12
Tabelle mit Details	S7-Kommunikation	29.2
	Offene-Kommunikation mit Send/Receive-Bausteinen	31.2
	Offene-Kommunikation mit T-Bausteinen	32.2
	PNIO	34.2
Modbus/TCP (SIMATIC / fremder Controller)		44

18.2 Netz PB

Übersicht Schnittstellen und Kommunikationsarten

Tabelle 18-4

Familie		Kapitel
Modulare Controller	ET 200 CPU	20.2.1
	S7-300	20.2.2
	S7-400	20.2.3
	S7-1200	20.2.4
Embedded Controller	S7-mEC	20.2.5
	Box PC	20.2.6
	Panel PC	20.2.7
	WinAC MP	20.2.8
PC-based Controller	WinAC RTX	20.2.9

Kombinationen Controller 1 / Controller 2

Tabelle 18-5

Controller 2		Controller 1								
		Modular				Embedded				PC-based
		ET 200 CPU	S7-300	S7-400	S7-1200	S7-mEC	Box PC	Panel PC	WinAC MP	WinAC RTX
Modular	ET 200 CPU	20.3.1	20.3.2	20.3.3	---	20.3.4	20.3.5	20.3.6	20.3.7	20.3.8
	S7-300	20.3.2	20.4.2	20.4.3	---	20.4.4	20.4.5	20.4.6	20.4.7	20.4.8
	S7-400	20.3.3	20.4.3	20.5.3	---	20.5.4	20.5.5	20.5.6	20.5.7	20.5.8
	S7-1200	---	---	---	---	---	---	---	---	---
Embedded	S7-mEC	20.3.4	20.4.4	20.5.4	---	20.6.4	20.6.5	20.6.6	20.6.7	20.6.8
	Box PC	20.3.5	20.4.5	20.5.5	---	20.6.5	20.7.5	20.7.6	20.7.7	20.7.8
	Panel PC	20.3.6	20.4.6	20.5.6	---	20.6.6	20.7.6	20.8.6	20.8.7	20.8.8
	WinAC MP	20.3.7	20.4.7	20.5.7	---	20.6.7	20.7.7	20.8.7	20.9.7	20.9.8
PC-based	WinAC RTX	20.3.8	20.4.8	20.5.8	---	20.6.8	20.7.8	20.8.8	20.9.8	20.10.8

Kommunikationsarten

Tabelle 18-6

Kommunikationsart		Kapitel
Alle Kommunikationsarten (Tabelle Kompakt)		20.11
Tabelle mit Details	S7-Basiskommunikation	28.2
	S7-Kommunikation	29.2
	Offene-Kommunikation mit Send/Receive-Bausteinen	36.2
	FMS-Kommunikation	37.2
	DP-Kommunikation	38.2

18.3 Netz MPI

Übersicht Schnittstellen und Kommunikationsarten

Tabelle 18-7

Familie		Kapitel
Modulare Controller	ET 200 CPU	21.2.1
	S7-300	21.2.2
	S7-400	21.2.3

Kombinationen Controller 1 / Controller 2

Tabelle 18-8

Controller 2		Controller 1								
		Modular				Embedded				PC-based
		ET 200 CPU	S7-300	S7-400	S7-1200	S7-mEC	Box PC	Panel PC	WinAC MP	WinAC RTX
Modular	ET 200 CPU	21.3.1	21.3.2	21.3.3	---	---	---	---	---	---
	S7-300	21.3.2	21.4.2	21.4.3	---	---	---	---	---	---
	S7-400	21.3.3	21.4.3	21.5.3	---	---	---	---	---	---
	S7-1200	---	---	---	---	---	---	---	---	---
Embedded	S7-mEC	---	---	---	---	---	---	---	---	---
	Box PC	---	---	---	---	---	---	---	---	---
	Panel PC	---	---	---	---	---	---	---	---	---
	WinAC MP	---	---	---	---	---	---	---	---	---
PC-based	WinAC RTX	---	---	---	---	---	---	---	---	---

Copyright © Siemens AG 2011 All rights reserved
20982954_SIMATIC_Comm_DOKU_v201_d.doc

Kommunikationsarten

Tabelle 18-9

Kommunikationsart		Kapitel
Alle Kommunikationsarten (Tabelle Kompakt)		21.6
Tabelle mit Details	Globaldaten	27.2
	S7-Basiskommunikation	28.2
	S7-Kommunikation	29.2

18.4 SIMATIC Rückwandbus

Übersicht Schnittstellen und Kommunikationsarten

Tabelle 18-10

Familie		Kapitel
Modulare Controller	S7-400	22.2

Kombinationen Controller 1 / Controller 2

Tabelle 18-11

Controller	Controller		
	Modular	Embedded	PC-based
Modular	22.3	---	---
Embedded	---	---	---
PC-based	---	---	---

Kommunikationsarten

Tabelle 18-12

Kommunikationsart		Kapitel
Alle Kommunikationsarten (Tabelle Kompakt)		22.4
Tabelle mit Details	Globaldaten	27.2
	S7-Basiskommunikation	28.2
	S7-Kommunikation	29.2

18.5 Serielle Schnittstellen

Übersicht Schnittstellen und Kommunikationsarten

Tabelle 18-13

Familie		Kapitel
Modulare Controller	ET 200 CPU	23.2
	S7-300	23.3
	S7-400	23.4
	S7-1200	23.5
Embedded Controller	S7-mEC	23.6
	Box PC	23.7
	Panel PC	23.8
	WinAC MP	23.9
PC-based Controller	WinAC RTX	23.10

Kommunikationsarten

Tabelle 18-14

Kommunikationsart	Kapitel
3964(R), ASCII, RK 512	40
Anwenderdefiniertes Protokoll	41
Modbus Seriell (RTU Format)	45

19 Auswahlhilfe: PROFINET/Industrial Ethernet (PN/IE)

19.1 PN/IE: Inhalt des Kapitels

Für das Medium PN/IE wird beschrieben:

- Welche Schnittstellen (Baugruppen) und Kommunikationsarten stehen zur Verfügung?
(-> Tabelle Schnittstellen)
- Welche Partner können über welche Kommunikationsarten miteinander kommunizieren?
(-> Tabelle Kombinationen)
- Übersicht aller zur Verfügung stehenden Kommunikationsarten
(-> Tabelle Kommunikationsarten-Kompakt)

19.2 PN/IE: Schnittstellen und Kommunikationsarten

19.2.1 ET 200 CPU an PN/IE

Tabelle 19-1

Controller an PN/IE: ET 200 CPU			Kommunikationsarten				
			SIMATIC spezifisch		Offener Standard		
			S7-Kommunikation	Offene-Kommunikation	PN-Kommunikation		
					IOC	IOD	CBA
ET 200S	CPU	IM 151-8(F) PN/DP CPU	(1) (IoT) (14) (IoT)	(3) (IoT, TCP, UDP)	(2)	(2)	x
ET 200Pro	CPU	IM 154-8(F) PN/DP CPU	(1) (IoT) (14) (IoT)	(3) (IoT, TCP, UDP)	(2)	(2)	x

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (3) TSEND/TRCV, TUSEND/TURCV
- (14) USEND_E/URCV_E, PUT_E, GET_E

Copyright © Siemens AG 2011 All rights reserved
20982954_SIMATIC_Comm_DOKU_v201_d.doc

19.2.2 S7-300 an PN/IE

Tabelle 19-2

Controller an PN/IE: S7-300		Kommunikationsart				
		SIMATIC spezifisch		Offener Standard		
		S7-Kommunikation	Offene-Kommunikation	PN-Kommunikation		
				IOC	IOD	CBA
CPU	alle mit Schnittstelle: PN (*2)	(1) (IoT) (14) (IoT)	(3) (IoT, TCP, UDP) (13) (IoT, TCP) (*1)	(2)	(2) (*4)	x
CP	343-1 Lean	PUT, GET Server (IoT)	(8) (IoT, TCP, UDP) (13) (IoT, TCP)	---	(6)	---
	343-1	(1) (IoT, ISO)	(8) (IoT, TCP, UDP, ISO) (13) (IoT, TCP, ISO)	(6)	(6)	---
	343-1 Advanced	(1) (IoT, ISO)	(8) (IoT, TCP, UDP, ISO) (13) (IoT, TCP, ISO)	(6)	(6)	x
	343-1 ERPC (*3)	(1) (IoT)	(8) (IoT, TCP, UDP) (13) (TCP)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
 (3) TSEND/TRCV, TUSEND/TURCV (6) PNIO_SEND, PNIO_RECV
 (8) AG_SEND/AG_RECV (13) Server für Fetch/Write (14) USEND_E / URCV_E, PUT_E, GET_E

Erläuterungen zur Tabelle

(*1): Kommunikationsbausteine: FW_TCP bzw. FW_IOT (nutzen intern die T-Bausteine). In STEP 7 enthalten (Standard Library).

(*2): CPU 315(F)-2 PN/DP, CPU 317(F)-2 PN/DP, CPU 319(F)-3 PN/DP

(*3): Der CP unterstützt die ERPC-Kommunikation. Da dies nicht Gegenstand der Dokumentation ist, erfolgt in der Tabelle kein Eintrag.

(*4) IO Controller als I-Device

19.2.3 S7-400 an PN/IE

Tabelle 19-3

Controller an PN/IE: S7-400		Kommunikationsart				
		SIMATIC spezifisch		Offener Standard		
		S7-Kommunikation	Offene-Kommunikation	PN-Kommunikation		
				IOC	IOD	CBA
CPU	alle mit Schnittstelle: PN (*2)	(1) (IoT)	(3) (IoT, TCP, UDP) (13) (IoT, TCP) (*1)	(2)	(2)	x
CP	443-1	(1) (IoT, ISO)	(4) (IoT) (9) (ISO, IoT, TCP, UDP) (13) (ISO, IoT, TCP)	(2)	(2) (*3)	---
	443-1 Adv.	(1) (IoT, ISO)	(4) (IoT) (9) (ISO, IoT, TCP, UDP) (13) (ISO, IoT, TCP)	(2)	(2) (*3)	x

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(3) TSEND/TRCV, TUSEND/TURCV

(9) AG_SEND/AG_RECV, AG_LSEND/AG_LRECV, AG_SSEND/AG_SRECV

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(4) TSEND/TRCV

(13) Server für Fetch/Write

Erläuterungen zur Tabelle

(*1): Kommunikationsbausteine: FW_TCP bzw. FW_IOT (nutzen intern die T-Bausteine)

(*2): CPU 412-2 PN, CPU 414(F)-3 PN/DP, CPU 416(F)-3 PN/DP

(*3) IO Controller als I-Device

19.2.4 S7-1200 an PN/IE

Tabelle 19-4

Controller an PN/IE: S7-1200		Kommunikationsart					
		SIMATIC spezifisch		Offener Standard			
		S7-Kommunikation		Offene-Kommunikation	PN-Kommunikation		
					IOC	IOD	CBA
CPU	1211C 1212C 1214C	PUT, GET Server (IoT)	(4) (IoT, TCP) (5) (IoT, TCP)	---	---	---	

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (4) TSEND/TRCV
- (5) TSEND_C/TRCV_C

19.2.5 S7-mEC an PN/IE

Tabelle 19-5

Controller an PN/IE: S7-mEC			Kommunikationsart				
			SIMATIC spezifisch		Offener Standard		
			S7-Kommunikation	Offene-Kommunikation	PN-Kommunikation		
					IOC	IOD	CBA
CPU	EC31 (*1)	unter Kontrolle von RTX (Submodule) (*2): X1	(1) (IoT) (14) (IoT)	(3) (TCP, UDP)	(2)	---	x
		unter Kontrolle von Windows (*3): X2	(1) (IoT) (14) (IoT)	---	---	---	---
CP	EM PC	unter Kontrolle von Windows (*3): X1	(1) (IoT)	---	---	---	---
	EM PCI-104 (*4)	---	---	---	---	---	---
	CPs aus S7-300	---	---	---	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(3) TSEND/TRCV, TUSEND/TURCV

(14) USEND_E / URCV_E, PUT_E, GET_E

Erläuterungen zur Tabelle

(*1): mit vorinstalliertem WinAC RTX (F) 2009

(*2): Schnittstelle, unter der Kontrolle von WinAC RTX. (F). Die Schnittstelle muss als „Submodul“ konfiguriert werden.

(*3): Schnittstellen unter der Kontrolle von Windows. Die beiden Schnittstellen sind nicht gleichzeitig betreibbar (EC31 EXOR EM PC). Die Schnittstelle am EM PC ist eine „Gigabit Ethernet Schnittstelle“.

(*4): steckbar sind maximal 3 Karten des PCI-104 Standards

19.2.6 Box PC an PN/IE

Tabelle 19-6

Controller an PN/IE: Box PC (*1) IPC427C Bundles mit RTX		Kommunikationsart				
		SIMATIC spezifisch		Offener Standard		
		S7-Kommunikation	Offene-Kommunikation	PN-Kommunikation		
				IOC	IOD	CBA
CPU / CP (*2)	unter Kontrolle von RTX (Submodule) (*3): • „CP1616-CP1604“	(1) (IoT)	(3) (TCP, UDP)	(2)	---	x
	unter Kontrolle von Windows (*4): • „IE Allgemein“	(1) (IoT)	---	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (3) TSEND/TRCV, TUSEND/TURCV

Erläuterungen zur Tabelle

(*1): mit vorinstalliertem WinAC RTX (F) 2009 (optional)

(*2): Bedeutung der Schnittstellen CPU bzw. CP: CPU = Onboard Schnittstelle des PC, CP = steckbare PC-Karte

Die Bezeichnungen der Schnittstellen in der Tabelle sind Sammelbezeichnungen. Die folgende Tabelle erläutert dies.

Tabelle 19-7

Sammelbezeichnung	PC Karte (entspricht CP Schnittstelle)	Onboard Schnittstelle des PC (entspricht CPU Schnittstelle)
„CP1616-CP1604“	CP 1616, CP1604	Onboard PN-Schnittstelle der SIMATIC PC: CP 1616 integriert
„IE Allgemein“	Intel PRO/1000: 82574L, 82571EB, 82573L, 82541PI (non shared IRQ erforderlich)	Onboard Ethernet-Schnittstelle der SIMATIC PC: 4x7B, 6x7B, 8x7B, 4x7C

(*3): Schnittstellen, unter der Kontrolle von WinAC RTX. Die Schnittstellen müssen als „Submodule“ konfiguriert werden.

(*4): Schnittstellen unter der Kontrolle von Windows.

19.2.7 Panel PC an PN/IE

Tabelle 19-8

Controller an PN/IE: Box PC (*1) HMI IPC477C Bundles mit RTX		Kommunikationsart				
		SIMATIC spezifisch	Offener Standard			
		S7-Kommunikation	Offene-Kommunikation	PN-Kommunikation		
				IOC	IOD	CBA
CPU / CP (*2)	unter Kontrolle von RTX (Submodule) (*3): • „CP1616-CP1604“	(1) (IoT)	(3) (TCP, UDP)	(2)	---	x
	unter Kontrolle von Windows (*4): • „IE Allgemein“	(1) (IoT)	---	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (3) TSEND/TRCV, TUSEND/TURCV

Erläuterungen zur Tabelle

(*1): mit vorinstalliertem WinAC RTX (F) 2009 (optional)

(*2): Bedeutung der Schnittstellen CPU bzw. CP: CPU = Onboard Schnittstelle des PC, CP = steckbare PC-Karte

Die Bezeichnungen der Schnittstellen in der Tabelle sind Sammelbezeichnungen. Die folgende Tabelle erläutert dies.

Tabelle 19-9

Sammelbezeichnung	PC Karte (entspricht CP Schnittstelle)	Onboard Schnittstelle des PC (entspricht CPU Schnittstelle)
„CP1616-CP1604“	CP 1616, CP1604	Onboard PN-Schnittstelle der SIMATIC PC: CP 1616 integriert
„IE Allgemein“	Intel PRO/1000: 82574L, 82571EB, 82573L, 82541PI (non shared IRQ erforderlich)	Onboard Ethernet-Schnittstelle der SIMATIC PC: 4x7B, 6x7B, 8x7B, 4x7C

(*3): Schnittstellen, unter der Kontrolle von WinAC RTX. Die Schnittstellen müssen als „Submodule“ konfiguriert werden.

(*4): Schnittstellen unter der Kontrolle von Windows.

19.2.8 WinAC MP an PN/IE

Tabelle 19-10

Controller an PN/IE: WinAC MP für Multipanel (*1) MP177, MP277, MP377		Kommunikationsart				
		SIMATIC spezifisch		Offener Standard		
		S7-Kommunikation		PN-Kommunikation		
				IOC	IOD	CBA
CPU	unter Kontrolle von WinAC MP: integrierte IE-Schnittstelle	(1) (IoT)	---	---	---	

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

Erläuterungen zur Tabelle

(*1): mit WinAC MP 2008

19.2.9 WinAC RTX an PN/IE

Tabelle 19-11

Controller an PN/IE: WinAC RTX (*1)		Kommunikationsart				
		SIMATIC spezifisch	Offener Standard			
		S7-Kommunikation	Offene-Kommunikation	PN-Kommunikation		
				IOC	IOD	CBA
CPU / CP (*2)	unter Kontrolle von RTX (Submodule) (*3): <ul style="list-style-type: none"> „CP1616-CP1604“ „IE Allgemein“ 	(1) (IoT)	(3) (TCP, UDP)	(2)	---	x
	unter Kontrolle von Windows (*4): <ul style="list-style-type: none"> „CP1616-CP1604“ „IE Allgemein“ 	(1) (IoT)	---	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (3) TSEND/TRCV, TUSEND/TURCV

Copyright © Siemens AG 2011 All rights reserved
20982954_SIMATIC_Comm_DOKU_v201_d.doc

Erläuterungen zur Tabelle

(*1): WinAC RTX (F) 2009 auf PC

(*2): Bedeutung der Schnittstellen CPU bzw. CP: CPU = Onboard Schnittstelle des PC, CP = steckbare PC-Karte

Die Bezeichnungen der Schnittstellen in der Tabelle sind Sammelbezeichnungen. Die folgende Tabelle erläutert dies.

Tabelle 19-12

Sammelbezeichnung	PC Karte (entspricht CP Schnittstelle)	Onboard Schnittstelle des PC (entspricht CPU Schnittstelle)
„CP1616-CP1604“	CP 1616, CP1604	Onboard PN-Schnittstelle der SIMATIC PC: CP 1616 integriert
„IE Allgemein“	Intel PRO/1000: 82574L, 82571EB, 82573L, 82541PI (non shared IRQ erforderlich)	Onboard Ethernet-Schnittstelle der SIMATIC PC: 4x7B, 6x7B, 8x7B, 4x7C

(*3): Schnittstellen, unter der Kontrolle von WinAC RTX. Die Schnittstellen müssen als „Submodule“ konfiguriert werden.

(*4): Schnittstellen unter der Kontrolle von Windows.

19.3 PN/IE: Controller 1 = ET 200 CPU

19.3.1 ET 200 CPU / ET 200 CPU

Tabelle 19-13

Controller 2: ET 200 CPU			Controller 1: ET 200 CPU an PN/IE			
			CPU			
			IM 151-8(F) PN/DP CPU, IM 154-8(F) PN/DP CPU			
			IOC, IOD			
			S7	OC	PN (*1)	
					PNIO	CBA
CPU	IM 151-8(F) PN/DP CPU IM 154-8(F) PN/DP CPU	IOC, IOD	(1)+(14)	(3)	(2)	x

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (3) TSEND/TRCV, TUSEND/TURCV
- (14) USEND_E/URCV_E, PUT_E, GET_E

Erläuterungen zur Tabelle

(*1): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

19.3.2 ET 200 CPU / S7-300

Tabelle 19-14

Controller 2: S7-300			Controller 1: ET 200 CPU an PN/IE			
			CPU			
			IM 151-8(F) PN/DP, IM 154-8(F) PN/DP CPU			
			IOC, IOD			
			S7	OC	PN (*1)	
PNIO	CBA					
CPU	alle mit Schnittstelle: PN	IOC, IOD	(1)+(14)	(3)	(2)	x
CP	343-1 Lean	IOD	(22)	(3) / (8)	(2) / (6)	---
	343-1	IOC, IOD	(1)	(3) / (8)	(2) / (6)	---
	343-1 Advanced	IOC, IOD	(1)	(3) / (8)	(2) / (6)	x
	343-1 ERPC	---	(1)	(3) / (8)	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (3) TSEND/TRCV, TUSEND/TURCV
- (6) PNIO_SEND, PNIO_RECV
- (8) AG_SEND/AG_RECV
- (14) USEND_E/URCV_E, PUT_E, GET_E
- (22) Controller 2 ist Server (für PUT, GET)

Erläuterungen zur Tabelle

(*1): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

19.3.3 ET 200 CPU / S7-400

Tabelle 19-15

Controller 2: S7-400			Controller 1: ET 200 CPU an PN/IE			
			CPU			
			IM 151-8(F) PN/DP CPU, IM 154-8(F) PN/DP CPU			
			IOC, IOD			
			S7	OC	PN (*1)	
PNIO	CBA					
CPU	alle mit Schnittstelle: PN	IOC, IOD	(1)+(14)	(3)	(2)	x
CP	443-1	IOC, IOD	(1)	(3) / (4)+(9)	(2)	---
	443-1 Advanced	IOC, IOD	(1)	(3) / (4)+(9)	(2)	x

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (3) TSEND/TRCV, TUSEND/TURCV
- (4) TSEND/TRCV
- (9) AG_SEND/AG_RECV, AG_LSEND/AG_LRECV, AG_SSEND/AG_SRECV
- (14) USEND_E/URCV_E, PUT_E, GET_E

Erläuterungen zur Tabelle

(*1): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

19.3.4 ET 200 CPU / S7-1200

Tabelle 19-16

Controller 2: S7-1200			Controller 1: ET 200 CPU an PN/IE			
			CPU			
			IM 151-8(F) PN/DP CPU, IM 154-8(F) PN/DP CPU			
			IOC, IOD			
			S7	OC	PN	
		PNIO	CBA			
CPU	1211C 1212C 1214C	---	(22)+(32)	(4) / (4)+(5)	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (4) TSEND/TRCV
- (5) TSEND_C/TRCV_C
- (22) Controller 2 ist Server (für PUT, GET)
- (32) Controller 2 ist Server (für PUT_E, GET_E)

19.3.5 ET 200 CPU / S7-mEC

Tabelle 19-17

Controller 2: S7-mEC				Controller 1: ET 200 CPU an PN/IE			
				CPU			
				IM 151-8(F) PN/DP CPU, IM 154-8(F) PN/DP CPU			
				IOC, IOD			
				S7	OC	PN (*2)	
						PNIO	CBA
CPU	EC31	Kontrolle von RTX (Submodule): X1	IOC	(1)+(14)	(3)	(2)	x
		Kontrolle von Windows (*1): X2	---	(1)+(14)	---	---	---
CP	EM PC	Kontrolle von Windows (*1): X1	---	(1)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (3) TSEND/TRCV, TUSEND/TURCV
- (14) USEND_E/URCV_E, PUT_E, GET_E

Erläuterungen zur Tabelle

(*1): Die beiden Schnittstellen sind nicht gleichzeitig betreibbar (EC31 EXOR EM PC).

(*2): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

19.3.6 ET 200 CPU / Box PC

Tabelle 19-18

Controller 2: Box PC IPC427C Bundles mit RTX			Controller 1: ET 200 CPU an PN/IE			
			CPU			
			IM 151-8(F) PN/DP CPU, IM 154-8(F) PN/DP CPU			
			IOC, IOD			
			S7	OC	PN (*1)	
					PNIO	CBA
CPU/CP	Kontrolle von RTX (Submodule): „CP1616-CP1604“	IOC	(1)+(14)	(3)	(2)	x
	Kontrolle von Windows: „IE- Allgemein“	---	(1)+(14)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (3) TSEND/TRCV, TUSEND/TURCV
- (14) USEND_E/URCV_E, PUT_E, GET_E

Erläuterungen zur Tabelle

(*1): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

19.3.7 ET 200 CPU / Panel PC

Tabelle 19-19

Controller 2: Panel PC HMI IPC477C Bundles mit RTX			Controller 1: ET 200 CPU an PN/IE			
			CPU			
			IM 151-8(F) PN/DP CPU, IM 154-8(F) PN/DP CPU			
			IOC, IOD			
			S7	OC	PN (*1)	
		PNIO	CBA			
CPU/CP	Kontrolle von RTX (Submodule): „CP1616-CP1604“	IOC	(1)+(14)	(3)	(2)	x
	Kontrolle von Windows: „IE-Allgemein“	---	(1)+(14)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (3) TSEND/TRCV, TUSEND/TURCV
- (14) USEND_E/URCV_E, PUT_E, GET_E

Erläuterungen zur Tabelle

(*1): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

19.3.8 ET 200 CPU / WinAC MP

Tabelle 19-20

Controller 2: WinAC MP auf Multipanel MP177, MP277, MP377			Controller 1: ET 200 CPU an PN/IE			
			CPU			
			IM 151-8(F) PN/DP CPU, IM 154-8(F) PN/DP CPU			
			IOC, IOD			
			S7	OC	PN	
					PNIO	CBA
CPU	Kontrolle von WinAC MP: integrierte IE-Schnittstelle	---	(1)+(14)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(14) USEND_E/URCV_E, PUT_E, GET_E

19.3.9 ET 200 CPU / WinAC RTX

Tabelle 19-21

Controller 2: WinAC RTX auf PC WinAC RTX 2009			Controller 1: ET 200 CPU an PN/IE			
			CPU			
			IM 151-8(F) PN/DP CPU, IM 154-8(F) PN/DP CPU			
			IOC, IOD			
			S7	OC	PN (*1)	
					PNIO	CBA
CPU/CP	Kontrolle von RTX (Submodule): • „CP1616-CP1604“ • „IE-Allgemein“	IOC	(1)+(14)	(3)	(2)	x
	Kontrolle von Windows: • „CP1616-CP1604“ • „IE-Allgemein“	---	(1)+(14)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (3) TSEND/TRCV, TUSEND/TURCV
- (14) USEND_E/URCV_E, PUT_E, GET_E

Erläuterungen zur Tabelle

(*1): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

19.4 PN/IE: Controller 1 = S7-300

19.4.1 (S7-300 / ET 200 CPU)

Siehe ET 200 CPU / S7-300: 19.3.2

19.4.2 S7-300 / S7-300

Tabelle 19-22

Controller 2: S7-300			Controller 1: S7-300 an PN/IE																			
			CPU			343-1 Lean							343-1				343-1 Advanced				343-1 ERPC	
			alle mit Schnittstelle: PN			IOD							IOC, IOD				IOC, IOD				---	
			S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC		
					PNIO	CBA			PNIO	CBA			PNIO	CBA			PNIO	CBA				
CPU	alle mit Schnittstelle: PN	IOC, IOD	(1)+(14)	(3)	(2)	x	(21)	(8)/(3)	(6)/(2)	---	(1)	(8)/(3)	(6)/(2)	---	(1)	(8)/(3)	(6)/(2)	x	(1)	(8)/(3)		
CP	343-1 Lean	IOD	(22)	(3)/(8)	(2)/(6)	---	---	(8)	---	---	(22)	(8)	(6)	---	(22)	(8)	(6)	---	(22)	(8)		
	343-1	IOC, IOD	(1)	(3)/(8)	(2)/(6)	---	(21)	(8)	(6)	---	(1)	(8)	(6)	---	(1)	(8)	(6)	---	(1)	(8)		
	343-1 Advanced	IOC, IOD	(1)	(3)/(8)	(2)/(6)	x	(21)	(8)	(6)	---	(1)	(8)	(6)	---	(1)	(8)	(6)	x	(1)	(8)		
	343-1 ERPC	---	(1)	(3)/(8)	---	---	(21)	(8)	---	---	(1)	(8)	---	---	(1)	(8)	---	---	(1)	(8)		

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT (3) TSEND/TRCV, TUSEND/TURCV
 (6) PNIO_SEND, PNIO_RECV (8) AG_SEND/AG_RECV (14) USEND_E/URCV_E, PUT_E, GET_E
 (21) Controller 1 ist Server (für PUT, GET) (22) Controller 2 ist Server (für PUT, GET)

Erläuterungen zur Tabelle

(*1): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

19.4.3 S7-300 / S7-400

Tabelle 19-23

Controller 2: S7-400			Controller 1: S7-300 an PN/IE																	
			CPU			CP														
			alle mit Schnittstelle: PN			343-1 Lean				343-1				343-1 Advanced				343-1 ERPC		
			IOC, IOD			IOD				IOC, IOD				IOC, IOD				---		
			S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC
PNIO	CBA	PNIO			CBA	PNIO			CBA	PNIO			CBA							
CPU	alle mit Schnittstelle: PN	IOC, IOD	(1)	(3)	(2)	x	(21)	(8)/(3)	(6)/(2)	---	(1)	(8)/(3)	(6)/(2)	---	(1)	(8)/(3)	(6)/(2)	x	(1)	(8)/(3)
CP	443-1	IOC, IOD	(1)	(3)/(4)+(9)	(2)	---	(21)	(8)/(4)+(9)	(6)/(2)	---	(1)	(8)/(4)+(9)	(6)/(2)	---	(1)	(8)/(4)+(9)	(6)/(2)	---	(1)	(8)/(4)+(9)
	443-1 Advanced	IOC, IOD	(1)	(3)/(4)+(9)	(2)	x	(21)	(8)/(4)+(9)	(6)/(2)	---	(1)	(8)/(4)+(9)	(6)/(2)	---	(1)	(8)/(4)+(9)	(6)/(2)	x	(1)	(8)/(4)+(9)

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (3) TSEND/TRCV, TUSEND/TURCV (4) TSEND/TRCV
- (6) PNIO_SEND, PNIO_RECV (21) Controller 1 ist Server (für PUT, GET)
- (8) AG_SEND/AG_RECV (9) AG_SEND/AG_RECV, AG_LSEND/AG_LRECV, AG_SSEND/AG_SRECV

Erläuterungen zur Tabelle

(*1): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

19.4.4 S7-300 / S7-1200

Tabelle 19-24

Controller 2: S7-1200			Controller 1: S7-300 an PN/IE																						
			CPU			CP																			
			alle mit Schnittstelle: PN			343-1 Lean					343-1					343-1 Advanced					343-1 ERPC				
			IOC, IOD			IOD					IOC, IOD					IOC, IOD					---				
			S7	OC	PN	S7	OC	PN	S7	OC	PN	S7	OC	PN	S7	OC	PN	S7	OC	PN	S7	OC			
PNIO	CBA	PNIO																					CBA	PNIO	CBA
CPU	1211C	---	(22)	(3)/(4)+(5)	---	---	---	(8)/(4)+(5)	---	---	(22)	(8)/(4)+(5)	---	---	(22)	(8)/(4)+(5)	---	---	(22)	(8)/(4)+(5)					
	1212C																								
	1214C																								

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (3) TSEND/TRCV, TUSEND/TURCV
- (4) TSEND/TRCV
- (5) TSEND_C/TRCV_C
- (8) AG_SEND/AG_RECV
- (22) Controller 2 ist Server (für PUT, GET)

19.4.5 S7-300 / S7-mEC

Tabelle 19-25

Controller 2: S7-mEC				Controller 1: S7-300 an PN/IE																			
				CPU				CP															
				alle mit Schnittstelle: PN				343-1 Lean				343-1				343-1 Advanced				343-1 ERPC			
				IOC, IOD				IOD				IOC, IOD				IOC, IOD				---			
				S7	OC	PN (*2)		S7	OC	PN (*2)		S7	OC	PN (*2)		S7	OC	PN		S7	OC		
PNIO	CBA	PNIO	CBA			PNIO	CBA			PNIO	CBA												
CPU	EC31	Kontrolle von RTX (Submodule): X1		IOC	(1)+(14)/(1)	(3)	(2)	x	(21)	(8)/(3)	(6)/(2)	---	(1)	(8)/(3)	(6)/(2)	---	(1)	(8)/(3)	(6)/(2)	x	(1)	(8)/(3)	
		Kontrolle von Windows (*1): X2			---	(1)+(14)/(1)	---	---	---	(21)	---	---	---	(1)	---	---	---	(1)	---	---	---	(1)	---
CP	EM PC	Kontrolle von Windows (*1): X1		---	(1)	---	---	---	(21)	---	---	---	(1)	---	---	---	(1)	---	---	---	(1)	---	

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (3) TSEND/TRCV, TUSEND/TURCV
- (8) AG_SEND/AG_RECV
- (21) Controller 1 ist Server (für PUT, GET)

- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (6) PNIO_SEND, PNIO_RECV
- (14) USEND_E/URCV_E, PUT_E, GET_E

Erläuterungen zur Tabelle

- (*1): Die beiden Schnittstellen sind nicht gleichzeitig betreibbar (EC31 EXOR EM PC).
- (*2): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

Copyright © Siemens AG 2011 All rights reserved
20982954_SIMATIC_Comm_DOKU_v201_d.doc

19.4.6 S7-300 / Box PC

Tabelle 19-26

Controller 2: Box PC IPC427C Bundles mit RTX			Controller 1: S7-300 an PN/IE																	
			CPU				CP													
			alle mit Schnittstelle: PN				343-1 Lean				343-1				343-1 Advanced				343-1 ERPC	
			IOC, IOD				IOD				IOC, IOD				IOC, IOD				---	
			S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC
PNIO	CBA	PNIO			CBA	PNIO			CBA	PNIO			CBA							
CPU/CP	Kontrolle von RTX (Submodule): „CP1616-CP1604“	IOC	(1)	(3)	(2)	x	(21)	(8)/(3)	(6)/(2)	---	(1)	(8)/(3)	(6)/(2)	---	(1)	(8)/(3)	(6)/(2)	x	(1)	(8)/(3)
	Kontrolle von Windows: „IE-Allgemein“	---	(1)	---	---	---	(21)	---	---	---	(1)	---	---	---	(1)	---	---	---	(1)	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (3) TSEND/TRCV, TUSEND/TURCV
- (8) AG_SEND/AG_RECV

- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (6) PNIO_SEND, PNIO_RECV
- (21) Controller 1 ist Server (für PUT, GET)

Erläuterungen zur Tabelle

(*1): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

19.4.7 S7-300 / Panel PC

Tabelle 19-27

Controller 2: Panel PC HMI IPC477C Bundles mit RTX			Controller 1: S7-300 an PN/IE																													
			CPU						CP																							
			alle mit Schnittstelle: PN						343-1 Lean						343-1						343-1 Advanced						343-1 ERPC					
			IOC, IOD						IOD						IOC, IOD						IOC, IOD						---					
			S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC								
PNIO	CBA	PNIO			CBA	PNIO			CBA	PNIO			CBA	PNIO			CBA															
CPU/CP	Kontrolle von RTX (Submodule): „CP1616-CP1604“	IOC	(1)	(3)	(2)	x	(21)	(8)/(3)	(6)/(2)	---	(1)	(8)/(3)	(6)/(2)	---	(1)	(8)/(3)	(6)/(2)	x	(1)	(8)/(3)												
	Kontrolle von Windows: „IE-Allgemein“	---	(1)	---	---	---	(21)	---	---	---	(1)	---	---	---	(1)	---	---	---	(1)	---												

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (3) TSEND/TRCV, TUSEND/TURCV
- (8) AG_SEND/AG_RECV

- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (6) PNIO_SEND, PNIO_RECV
- (21) Controller 1 ist Server (für PUT, GET)

Erläuterungen zur Tabelle

(*1): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

19.4.8 S7-300 / WinAC MP

Tabelle 19-28

Controller 2: WinAC MP MP177, MP277, MP377			Controller 1: S7-300 an PN/IE																					
			CPU						CP															
			alle mit Schnittstelle: PN						343-1 Lean				343-1				343-1 Advanced				343-1 ERPC			
			IOC, IOD						IOD				IOC, IOD				IOC, IOD				---			
			S7	OC	PN		S7	OC	PN		S7	OC	PN		S7	OC	PN		S7	OC	PN		S7	OC
PNIO	CBA	PNIO			CBA	PNIO			CBA	PNIO			CBA	PNIO			CBA							
CPU	Kontrolle von WinAC MP: integrierte IE-Schnittstelle	---	(1)	---	---	---	(21)	---	---	---	(1)	---	---	---	(1)	---	---	---	(1)	---				

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(21) Controller 1 ist Server (für PUT, GET)

19.4.9 S7-300 / WinAC RTX

Tabelle 19-29

Controller 2: WinAC RTX auf PC WinAC RTX 2009			Controller 1: S7-300 an PN/IE																	
			CPU			CP														
			alle mit Schnittstelle: PN			343-1 Lean				343-1				343-1 Advanced				343-1 ERPC		
			IOC, IOD			IOD				IOC, IOD				IOC, IOD				---		
			S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC
PNIO	CBA	PNIO			CBA	PNIO			CBA	PNIO			CBA							
CPU/CP	Kontrolle von RTX (Submodule) (IOC): • „CP1616-CP1604“ • „IE-Allgemein“	IOC	(1)	(3)	(2)	x	(21)	(8)/(3)	(6)/(2)	---	(1)	(8)/(3)	(6)/(2)	---	(1)	(8)/(3)	(6)/(2)	x	(1)	(8)/(3)
	Kontrolle von Windows (---): • „CP1616-CP1604“ • „IE-Allgemein“	---	(1)	---	---	---	(21)	---	---	---	(1)	---	---	---	(1)	---	---	---	(1)	

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(3) TSEND/TRCV, TUSEND/TURCV

(8) AG_SEND/AG_RECV

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(6) PNIO_SEND, PNIO_RECV

(21) Controller 1 ist Server (für PUT, GET)

Erläuterungen zur Tabelle

(*1): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

19.5 PN/IE: Controller 1 = S7-400

19.5.1 (S7-400 / ET 200 CPU)

Siehe ET 200 CPU / S7-400: 19.3.3

19.5.2 (S7-400 / S7-300)

Siehe S7-300 / S7-400: 19.4.3

19.5.3 S7-400 / S7-400

Tabelle 19-30

Controller 2: S7-400			Controller 1: S7-400 an PN/IE											
			CPU						CP					
			alle mit Schnittstelle: PN						443-1			443-1 Advanced		
			IOC, IOD						IOC, IOD					
			S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC	PN (*1)	
PNIO	CBA	PNIO			CBA	PNIO			CBA					
CPU	alle mit Schnittstelle: PN	IOC, IOD	(1)	(3)	(2)	x	(1)	(4)+(9)/(3)	(2)	---	(1)	(4)+(9)/(3)	(2)	x
CP	443-1	IOC, IOD	(1)	(3)/(4)+(9)	(2)	---	(1)	(4)+(9)	(2)	---	(1)	(4)+(9)	(2)	---
	443-1 Advanced	IOC, IOD	(1)	(3)/(4)+(9)	(2)	x	(1)	(4)+(9)	(2)	---	(1)	(4)+(9)	(2)	x

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (3) TSEND/TRCV, TUSEND/TURCV (4) TSEND/TRCV
- (9) AG_SEND/AG_RECV, AG_LSEND/AG_LRECV, AG_SSEND/AG_SRECV

Erläuterungen zur Tabelle

(*1): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

19.5.4 S7-400 / S7-1200

Tabelle 19-31

Controller 2: S7-1200			Controller 1: S7-400 an PN/IE												
			CPU				CP								
			alle mit Schnittstelle: PN				443-1				443-1 Advanced				
			IOC, IOD				IOC, IOD				IOC, IOD				
			S7	OC	PN		S7	OC	PN		S7	OC	PN		
PNIO	CBA	PNIO			CBA	PNIO			CBA						
CPU	1211C 1212C 1214C	---		(22)	(3)/(4)+(5)	---	---	(22)	(4)+(9)/(4)+(5)	---	---	(22)	(4)+(9)/(4)+(5)	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (3) TSEND/TRCV, TUSEND/TURCV
- (4) TSEND/TRCV
- (5) TSEND_C/TRCV_C
- (9) AG_SEND/AG_RECV, AG_LSEND/AG_LRECV, AG_SSEND/AG_SRECV
- (22) Controller 2 ist Server (für PUT, GET)

19.5.5 S7-400 / S7-mEC

Tabelle 19-32

Controller 2: S7-mEC				Controller 1: S7-400 an PN/IE												
				CPU				CP								
				alle mit Schnittstelle: PN				443-1				443-1 Advanced				
				IOC, IOD				IOC, IOD				IOC, IOD				
				S7	OC	PN (*2)		S7	OC	PN (*2)		S7	OC	PN (*2)		
PNIO	CBA	PNIO	CBA			PNIO	CBA									
CPU	EC31	Kontrolle von RTX (Submodule): X1		IOC	(1)	(3)	(2)	x	(1)	(4)+(9)/(3)	(2)	---	(1)	(4)+(9)/(3)	(2)	x
		Kontrolle Windows (*1): X2		---	(1)	---	---	---	(1)	---	---	---	(1)	---	---	---
CP	EM PC	Kontrolle von Windows (*1): X1		---	(1)	---	---	---	(1)	---	---	---	(1)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (3) TSEND/TRCV, TUSEND/TURCV (4) TSEND/TRCV
- (9) AG_SEND/AG_RECV, AG_LSEND/AG_LRECV, AG_SSEND/AG_SRECV

Erläuterungen zur Tabelle

- (*1): Die beiden Schnittstellen sind nicht gleichzeitig betreibbar (EC31 EXOR EM PC).
- (*2): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

19.5.6 S7-400 / Box PC

Tabelle 19-33

Controller 2: Box PC IPC427C Bundles mit RTX			Controller 1: S7-400 an PN/IE											
			CPU				CP							
			alle mit Schnittstelle: PN				443-1				443-1 Advanced			
			IOC, IOD				IOC, IOD				IOC, IOD			
			S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC	PN (*1)	
PNIO	CBA	PNIO			CBA	PNIO			CBA					
CPU/CP	Kontrolle von RTX (Submodule): „CP1616-CP1604“	IOC	(1)	(3)	(2)	x	(1)	(4)+(9)/(3)	(2)	---	(1)	(4)+(9)/(3)	(2)	x
	Kontrolle von Windows: „IE-Allgemein“	---	(1)	---	---	---	(1)	---	---	---	(1)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
 (3) TSEND/TRCV, TUSEND/TURCV (4) TSEND/TRCV
 (9) AG_SEND/AG_RECV, AG_LSEND/AG_LRECV, AG_SSEND/AG_SRECV

Erläuterungen zur Tabelle

(*1): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

19.5.7 S7-400 / Panel PC

Tabelle 19-34

Controller 2: Panel PC HMI IPC477C Bundles mit RTX			Controller 1: S7-400 an PN/IE											
			CPU						CP					
			alle mit Schnittstelle: PN						443-1			443-1 Advanced		
			IOC, IOD						IOC, IOD			IOC, IOD		
			S7	OC	PN (*1)		S7	OC	PN (*1)		S7	OC	PN (*1)	
PNIO	CBA	PNIO			CBA	PNIO			CBA					
CPU/CP	Kontrolle von RTX (Submodule): „CP1616-CP1604“	IOC	(1)	(3)	(2)	x	(1)	(4)+(9)/(3)	(2)	---	(1)	(4)+(9)/(3)	(2)	x
	Kontrolle von Windows: „IE-Allgemein“	---	(1)	---	---	---	(1)	---	---	---	(1)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (3) TSEND/TRCV, TUSEND/TURCV (4) TSEND/TRCV
- (9) AG_SEND/AG_RECV, AG_LSEND/AG_LRECV, AG_SSEND/AG_SRECV

Erläuterungen zur Tabelle

(*1): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

19.5.8 S7-400 / WinAC MP

Tabelle 19-35

Controller 2: WinAC MP MP177, MP277, MP377			Controller 1: S7-400 an PN/IE													
			CPU						CP							
			alle mit Schnittstelle: PN						443-1				443-1 Advanced			
			IOC, IOD						IOC, IOD				IOC, IOD			
			S7	OC	PN		S7	OC	PN		S7	OC	PN			
PNIO	CBA	PNIO			CBA	PNIO			CBA							
CPU	Kontrolle von WinAC MP: integrierte IE-Schnittstelle	---	(1)	---	---	---	(1)	---	---	---	(1)	---	---	---		

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

19.5.9 S7-400 / WinAC RTX

Tabelle 19-36

Controller 2: WinAC RTX auf PC WinAC RTX 2009			Controller 1: S7-400 an PN/IE											
			CPU			CP								
			alle mit Schnittstelle: PN			443-1				443-1 Advanced				
			IOC, IOD			IOC, IOD								
			S7	OC	PN (*1)	S7	OC	PN (*1)	S7	OC	PN (*1)			
		PNIO	CBA			PNIO	CBA		PNIO	CBA				
CPU/CP	Kontrolle von RTX (Submodule): • „CP1616-CP1604“ • „IE-Allgemein“	IOC	(1)	(3)	(2)	x	(1)	(4)+(9)/(3)	(2)	---	(1)	(4)+(9)/(3)	(2)	x
	Kontrolle von Windows: • „CP1616-CP1604“ • „IE-Allgemein“	---	(1)	---	---	---	(1)	---	---	---	(1)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (3) TSEND/TRCV, TUSEND/TURCV (4) TSEND/TRCV
- (9) AG_SEND/AG_RECV, AG_LSEND/AG_LRECV, AG_SSEND/AG_SRECV

Erläuterungen zur Tabelle

(*1): Kommunikation über PNIO zwischen PROFINET IO Controller und PROFINET IO Device.

19.6 PN/IE: Controller 1 = S7-1200

19.6.1 (S7-1200 / ET 200 CPU)

Siehe ET 200 CPU / S7-1200: 19.3.4

19.6.2 (S7-1200 / S7-300)

Siehe S7-300 / S7-1200: 19.4.4

19.6.3 (S7-1200 / S7-400)

Siehe S7-400 / S7-1200: 19.5.4

19.6.4 S7-1200 / S7-1200

Tabelle 19-37

Controller 2: S7-1200			Controller 1: S7-1200 an PN/IE			
			CPU			
			1211C, 1212C, 1214C			

			S7	OC	PN	
					PNIO	CBA
CPU	1211C 1212C 1214C	---	---	(4)+(5)	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (4) TSEND/TRCV
- (5) TSEND_C/TRCV_C

Copyright © Siemens AG 2011 All rights reserved
20982954_SIMATIC_Comm_DOKU_v201_d.doc

19.6.5 S7-1200 / S7-mEC

Tabelle 19-38

Controller 2: S7-mEC				Controller 1: S7-1200 an PN/IE			
				CPU			
				1211C, 1212C, 1214C			

		S7	OC	PN			
				PNIO	CBA		
CPU	EC31	Kontrolle von RTX (Submodule): X1	IOC	(21)	(4)+(5)/(3)	---	---
		Kontrolle von Windows (*1): X2	---	(21)	---	---	---
CP	EM PC	Kontrolle von Windows (*1): X1	---	(21)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(3) TSEND/TRCV, TUSEND/TURCV

(4) TSEND/TRCV

(5) TSEND_C/TRCV_C

(21) Controller 1 ist Server (für PUT, GET)

Erläuterungen zur Tabelle

(*1): Die beiden Schnittstellen sind nicht gleichzeitig betreibbar (EC31 EXOR EM PC).

19.6.6 S7-1200 / Box PC

Tabelle 19-39

Controller 2: Box PC IPC427C Bundles mit RTX			Controller 1: S7-1200 an PN/IE			
			CPU			
			1211C, 1212C, 1214C			

			S7	OC	PN	
					PNIO	CBA
CPU/CP	Kontrolle von RTX (Submodule): „CP1616-CP1604“	IOC	(21)	(4)+(5)/(3)	---	---
	Kontrolle von Windows: „IE-Allgemein“	---	(21)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(3) TSEND/TRCV, TUSEND/TURCV

(4) TSEND/TRCV

(5) TSEND_C/TRCV_C (nur bei S7-1200)

(21) Controller 1 ist Server (für PUT, GET)

19.6.7 S7-1200 / Panel PC

Tabelle 19-40

Controller 2: Panel PC HMI IPC477C Bundles mit RTX			Controller 1: S7-1200 an PN/IE			
			CPU			
			1211C, 1212C, 1214C			

			S7	OC	PN	
					PNIO	CBA
CPU/CP	Kontrolle von RTX (Submodule): „CP1616-CP1604“	IOC	(21)	(4)+(5)/(3)	---	---
	Kontrolle von Windows: „IE-Allgemein“	---	(21)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(3) TSEND/TRCV, TUSEND/TURCV

(4) TSEND/TRCV

(5) TSEND_C/TRCV_C

(21) Controller 1 ist Server (für PUT, GET)

19.6.8 S7-1200 / WinAC MP

Tabelle 19-41

Controller 2: WinAC MP MP177, MP277, MP377			Controller 1: S7-1200 an PN/IE			
			CPU			
			1211C, 1212C, 1214C			

			S7	OC	PN	
					PNIO	CBA
CPU	Kontrolle von WinAC MP: integrierte IE-Schnittstelle	---	(21)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(21) Controller 1 ist Server (für PUT, GET)

19.6.9 S7-1200 / WinAC RTX

Tabelle 19-42

Controller 2: WinAC RTX auf PC WinAC RTX 2009			Controller 1: S7-1200 an PN/IE			
			CPU			
			1211C, 1212C, 1214C			

			S7	OC	PN	
					PNIO	CBA
CPU/CP	Kontrolle von RTX (Submodule): • „CP1616-CP1604“ • „IE-Allgemein“	IOC	(21)	(4)+(5)/(3)	---	---
	Kontrolle von Windows: • „CP1616-CP1604“ • „IE-Allgemein“	---	(21)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

- (3) TSEND/TRCV, TUSEND/TURCV
- (4) TSEND/TRCV
- (5) TSEND_C/TRCV_C (nur bei S7-1200)
- (21) PUT, GET, Controller 1 ist Server

19.7 PN/IE: Controller 1 = S7-mEC

19.7.1 (S7-mEC / ET 200 CPU)

Siehe ET 200 CPU / S7-mEC: 19.3.5

19.7.2 (S7-mEC / S7-300)

Siehe S7-300 / S7-mEC: 19.4.5

19.7.3 (S7-mEC / S7-400)

Siehe S7-400 / S7-mEC: 19.5.5

19.7.4 (S7-mEC / S7-1200)

Siehe S7-1200 / S7-mEC: 19.6.5

19.7.5 S7-mEC / S7-mEC

Tabelle 19-43

Controller 2: S7-mEC				Controller 1: S7-mEC an PN/IE					
				CPU				CP	
				EC31				EM PC	
				Submodul: X1			Windows (*1): X2	Windows (*1): X1	
				IOC				---	
				S7	OC	PN		S7	S7
		PNIO	CBA						
CPU	EC31	Kontrolle von RTX (Submodule): X1	IOC	(1)	(3)	---	x	(1)	(1)
		Kontrolle von Windows (*1): X2	---	(1)	---	---	---	(1)	(1)
CP	EM PC	Kontrolle von Windows (*1): X1	---	(1)	---	---	---	(1)	(1)

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(3) TSEND/TRCV, TUSEND/TURCV

Erläuterungen zur Tabelle

(*1): Die beiden Schnittstellen sind nicht gleichzeitig betreibbar (EC31 EXOR EM PC).

19.7.6 S7-mEC / Box PC

Tabelle 19-44

Controller 2: Box PC IPC427C Bundles mit RTX			Controller 1: S7-mEC an PN/IE					
			CPU				CP	
			EC31				EM PC	
			Submodul: X1			Windows (*1): X2		Windows (*1): X1
			IOC			---		---
			S7	OC	PN		S7	S7
		PNIO	CBA					
CPU/CP	Kontrolle von RTX (Submodule): „CP1616-CP1604“	IOC	(1)	(3)	---	x	(1)	(1)
	Kontrolle von Windows: „IE-Allgemein“	---	(1)	---	---	---	(1)	(1)

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(3) TSEND/TRCV, TUSEND/TURCV

Erläuterungen zur Tabelle

(*1): Die beiden Schnittstellen sind nicht gleichzeitig betreibbar (EC31 EXOR EM PC).

19.7.7 S7-mEC / Panel PC

Tabelle 19-45

Controller 2: Panel PC IPC477C Bundles mit RTX			Controller 1: S7-mEC an PN/IE						
			CPU				CP		
			EC31				EM PC		
			Submodul: X1		Windows (*1): X2		Windows (*1): X1		
			IOC			---		---	
			S7	OC	PN		S7	S7	
		PNIO	CBA						
CPU/CP	Kontrolle von RTX (Submodule): „CP1616-CP1604“	IOC	(1)	(3)	---	x	(1)	(1)	
	Kontrolle von Windows: „IE-Allgemein“	---	(1)	---	---	---	(1)	(1)	

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(3) TSEND/TRCV, TUSEND/TURCV

Erläuterungen zur Tabelle

(*1): Die beiden Schnittstellen sind nicht gleichzeitig betreibbar (EC31 EXOR EM PC).

19.7.8 S7-mEC / WinAC MP

Tabelle 19-46

Controller 2: WinAC MP MP177, MP277, MP377			Controller 1: S7-mEC an PN/IE					
			CPU				CP	
			EC31				EM PC	
			Submodul: X1			Windows (*1): X2		Windows (*1): X1
			IOC			---		---
			S7	OC	PN		S7	S7
		PNIO	CBA					
CPU	Kontrolle von WinAC MP: integrierte IE-Schnittstelle	---	(1)	---	---	(1)	(1)	

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

Erläuterungen zur Tabelle

(*1): Die beiden Schnittstellen sind nicht gleichzeitig betreibbar (EC31 EXOR EM PC).

19.7.9 S7-mEC / WinAC RTX

Tabelle 19-47

Controller 2: WinAC RTX auf PC WinAC RTX 2009			Controller 1: S7-mEC an PN/IE						
			CPU			CP			
			EC31			EM PC			
			Submodul: X1			Windows (*1): X2		Windows (*1): X1	
			IOC			---			
			S7	OC	PN	S7	S7		
			PNIO	CBA					
CPU/CP	Kontrolle von RTX (Submodule): • „CP1616-CP1604“ • „IE-Allgemein“	IOC	(1)	(3)	---	x	(1)	(1)	
	Kontrolle von Windows: • „CP1616-CP1604“ • „IE-Allgemein“	---	(1)		---	---	(1)	(1)	

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(3) TSEND/TRCV, TUSEND/TURCV

Erläuterungen zur Tabelle

(*1): Die beiden Schnittstellen sind nicht gleichzeitig betreibbar (EC31 EXOR EM PC).

19.8 PN/IE: Controller 1 = Box PC

19.8.1 (Box PC / ET 200 CPU)

Siehe ET 200 CPU / Box PC: 19.3.6

19.8.2 (Box PC / S7-300)

Siehe S7-300 / Box PC: 19.4.6

19.8.3 (Box PC / S7-400)

Siehe S7-400 / Box PC: 19.5.6

19.8.4 (Box PC / S7-1200)

Siehe S7-1200 / Box PC: 19.6.6

19.8.5 (Box PC / S7-mEC)

Siehe S7-mEC / Box PC: 19.7.6

19.8.6 Box PC / Box PC

Tabelle 19-48

Controller 2: Box PC IPC427C Bundles mit RTX			Controller 1: Box PC an PN/IE				
			IPC427C Bundles mit RTX				
			CPU/CP				
			Kontrolle von RTX (Submodule): „CP1616-CP1604“			Kontrolle von Windows: „IE-Allgemein“	
			IOC			---	
			S7	OC	PN	S7	
		PNIO	CBA				
CPU/CP	Kontrolle von RTX (Submodule): „CP1616-CP1604“	IOC	(1)	(3)	---	x	(1)
	Kontrolle von Windows: „IE-Allgemein“	---	(1)	---	---	---	(1)

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(3) TSEND/TRCV, TUSEND/TURCV

19.8.7 Box PC / Panel PC

Tabelle 19-49

Controller 2: Panel PC IPC427C Bundles mit RTX			Controller 1: Box PC an PN/IE				
			IPC427C Bundles mit RTX				
			CPU/CP				
			Kontrolle von RTX (Submodule): „CP1616-CP1604“			Kontrolle von Windows: „IE-Allgemein“	
			IOC			---	
			S7	OC	PN	S7	
		PNIO	CBA				
CPU/CP	Kontrolle von RTX (Submodule): „CP1616-CP1604“	IOC	(1)	(3)	---	x	(1)
	Kontrolle von Windows: „IE-Allgemein“	---	(1)	---	---	---	(1)

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(3) TSEND/TRCV, TUSEND/TURCV

19.8.8 Box PC / WinAC MP

Tabelle 19-50

Controller 2: WinAC MP MP177, MP277, MP377			Controller 1: Box PC an PN/IE				
			IPC427C Bundles mit RTX				
			CPU/CP				
			Kontrolle von RTX (Submodule): „CP1616-CP1604“			Kontrolle von Windows: „IE-Allgemein“	
			IOC			---	
			S7	OC	PN		S7
		PNIO	CBA				
CPU	Kontrolle von WinAC MP: integrierte IE-Schnittstelle	---	(1)	---	---	(1)	

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

19.8.9 Box PC / WinAC RTX

Tabelle 19-51

Controller 2: WinAC RTX auf PC WinAC RTX 2009			Controller 1: Box PC an PN/IE				
			IPC427C Bundles mit RTX				
			CPU/CP				
			Kontrolle von RTX (Submodule): „CP1616-CP1604“			Kontrolle von Windows: „IE-Allgemein“	
			IOC			---	
			S7	OC	PN	S7	
		PNIO	CBA				
CPU/CP	Kontrolle von RTX (Submodule): • „CP1616-CP1604“ • „IE-Allgemein“	IOC	(1)	(3)	---	x	(1)
	Kontrolle von Windows: • „CP1616-CP1604“ • „IE-Allgemein“	---	(1)	---	---	---	(1)

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(3) TSEND/TRCV, TUSEND/TURCV

19.9 PN/IE: Controller 1 = Panel PC

19.9.1 (Panel PC / ET 200 CPU)

Siehe ET 200 CPU / Panel PC: 19.3.7

19.9.2 (Panel PC / S7-300)

Siehe S7-300 / Panel PC: 19.4.7

19.9.3 (Panel PC / S7-400)

Siehe S7-400 / Panel PC: 19.5.7

19.9.4 (Panel PC / S7-1200)

Siehe S7-1200 / Panel PC: 19.6.7

19.9.5 (Panel PC / S7-mEC)

Siehe S7-mEC / Panel PC: 19.7.7

19.9.6 (Panel PC / Box PC)

Siehe Box PC / Panel PC: 19.8.7

19.9.7 Panel PC / Panel PC

Tabelle 19-52

Controller 2: Panel PC IPC4277C Bundles mit RTX			Controller 1: Panel PC an PN/IE				
			IPC477C Bundles mit RTX				
			CPU/CP				
			Kontrolle von RTX (Submodule): „CP1616-CP1604“			Kontrolle von Windows: „IE-Allgemein“	
			IOC				---
			S7	OC	PN		S7
			PNIO			CBA	
CPU/CP	Kontrolle von RTX (Submodule): „CP1616-CP1604“	IOC	(1)	(3)	---	x	(1)
	Kontrolle von Windows: „IE-Allgemein“	---	(1)	---	---	---	(1)

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(3) TSEND/TRCV, TUSEND/TURCV

19.9.8 Panel PC / WinAC MP

Tabelle 19-53

Controller 2: WinAC MP MP177, MP277, MP377			Controller 1: Panel PC an PN/IE				
			IPC477C Bundles mit RTX				
			CPU/CP				
			Kontrolle von RTX (Submodule): „CP1616-CP1604“			Kontrolle von Windows: „IE-Allgemein“	
			IOC			---	
			S7	OC	PN		S7
		PNIO	CBA				
CPU	Kontrolle von WinAC MP: integrierte IE-Schnittstelle	---	(1)	---	---	(1)	

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

19.9.9 Panel PC / WinAC RTX

Tabelle 19-54

Controller 2: WinAC RTX auf PC WinAC RTX 2009			Controller 1: Panel PC an PN/IE				
			IPC477C Bundles mit RTX				
			CPU/CP				
			Kontrolle von RTX (Submodule): „CP1616-CP1604“			Kontrolle von Windows: „IE-Allgemein“	
			IOC			---	
			S7	OC	PN	S7	
		PNIO	CBA				
CPU/CP	Kontrolle von RTX (Submodule): • „CP1616-CP1604“ • „IE-Allgemein“	IOC	(1)	(3)	---	x	(1)
	Kontrolle von Windows: • „CP1616-CP1604“ • „IE-Allgemein“	---	(1)	---	---	---	(1)

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(3) TSEND/TRCV, TUSEND/TURCV

19.10 PN/IE: Controller 1 = WinAC MP

19.10.1 (WinAC MP / ET 200 CPU)

Siehe ET 200 CPU / WinAC MP: 19.3.8

19.10.2 (WinAC MP / S7-300)

Siehe S7-300 / WinAC MP: 19.4.8

19.10.3 (WinAC MP / S7-400)

Siehe S7-400 / WinAC MP: 19.5.8

19.10.4 (WinAC MP / S7-1200)

Siehe S7-1200 / WinAC MP: 19.6.8

19.10.5 (WinAC MP / S7-mEC)

Siehe S7-mEC / WinAC MP: 19.7.8

19.10.6 (WinAC MP / Box PC)

Siehe Box PC / WinAC MP: 19.8.8

19.10.7 (WinAC MP / Panel PC)

Siehe Panel PC / WinAC MP: 19.9.8

19.10.8 WinAC MP / WinAC MP

Tabelle 19-55

Controller 2: WinAC MP MP177, MP277, MP377			Controller 1: WinAC MP an PN/IE			
			MP177, MP277, MP377			
			CPU			
			Kontrolle von WinAC MP: integrierte IE-Schnittstelle			

			S7	OC	PN	
		PNIO	CBA			
CPU	Kontrolle von WinAC MP: integrierte IE-Schnittstelle	---	(1)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

19.10.9 WinAC MP / WinAC RTX

Tabelle 19-56

Controller 2: WinAC RTX auf PC WinAC RTX 2009			Controller 1: WinAC MP an PN/IE			
			MP177, MP277, MP377			
			CPU			
			Kontrolle von WinAC MP: integrierte IE-Schnittstelle			

			S7	OC	PN	
					PNIO	CBA
CPU/CP	Kontrolle von RTX (Submodule):	IOC	(1)	---	---	---
	<ul style="list-style-type: none"> • „CP1616-CP1604“ • „IE-Allgemein“ 					
	Kontrolle von Windows:	---	(1)	---	---	---
	<ul style="list-style-type: none"> • „CP1616-CP1604“ • „IE-Allgemein“ 					

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

19.11 PN/IE: Controller 1 = WinAC RTX

19.11.1 (WinAC RTX / ET 200 CPU)

Siehe ET 200 CPU / WinAC RTX: 19.3.9

19.11.2 (WinAC RTX / S7-300)

Siehe S7-300 / WinAC RTX: 19.4.9

19.11.3 (WinAC RTX / S7-400)

Siehe S7-400 / WinAC RTX: 19.5.9

19.11.4 (WinAC RTX / S7-1200)

Siehe S7-1200 / WinAC RTX: 19.6.9

19.11.5 (WinAC RTX / S7-mEC)

Siehe S7-mEC / WinAC RTX: 19.7.9

19.11.6 (WinAC RTX / Box PC)

Siehe Box PC / WinAC RTX: 19.8.9

19.11.7 (WinAC RTX / Panel PC)

Siehe Panel PC / WinAC RTX: 19.9.9

19.11.8 (WinAC RTX / WinAC MP)

Siehe WinAC MP / WinAC RTX: 19.10.9

19.11.9 WinAC RTX / WinAC RTX

Tabelle 19-57

Controller 2: WinAC RTX auf PC WinAC RTX 2009			Controller 1: WinAC RTX an PN/IE							
			WinAC RTX 2009							
			CPU/CP							
			Kontrolle von RTX (Submodule):				Kontrolle von Windows:			
			<ul style="list-style-type: none"> „CP1616-CP1604“ „IE-Allgemein“ 				<ul style="list-style-type: none"> „CP1616-CP1604“ „IE-Allgemein“ 			
IOC			---							
CPU/CP	Kontrolle von RTX (Submodule):	IOC	S7	OC	PN		S7	OC	PN	
					PNIO	CBA			PNIO	CBA
	<ul style="list-style-type: none"> „CP1616-CP1604“ „IE-Allgemein“ 		(1)	(3)	---	x	(1)	---	---	---
	<ul style="list-style-type: none"> „CP1616-CP1604“ „IE-Allgemein“ 	---	(1)	---	---	---	(1)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(3) TSEND/TRCV, TUSEND/TURCV

19.12 PN/IE: Übersicht Kommunikationsarten

Gegenüberstellung aller über PN/IE verfügbaren Kommunikationsarten.

Die Daten sind den Tabellen Kommunikationsarten-Detail entnommen:

- S7-Kommunikation (Tabelle 29-1)
- Offene Kommunikation mit Send/Receive-Bausteinen (Tabelle 31-3)
- Offene Kommunikation mit T-Bausteinen (Tabelle 32-3)
- PN-Kommunikation (Tabelle 34-1)

Tabelle Kommunikationsarten-Kompakt:

Tabelle 19-58

	SIMATIC spezifisch	Offener Standard			
	S7-Kommunikation	Offene-Kommunikation			PN-Kommunikation
		Send/Receive-Bausteine	T-Bausteine	T-Compact-Bausteine	PNIO
Protokolle	ISO, IoT	ISO, IoT, TCP, UDP	IoT, TCP, UDP	IoT, TCP	PN
Schnittstellen	ISO: CP IoT: CPU, CP	CP	CPU, CP	CPU	CPU, CP
Kommunikationsbausteine (max. Daten)	BSEND (≤ 64 KByte) USEND, USEND_E (≥ 160 Byte) PUT, PUT_E, GET, GET_E (≥ 160 Byte)	AG_xSEND (ISO, IoT, TCP ≤ 8 KByte) (UDP ≤ 2 KByte) ----- Server für FETCH, WRITE (nicht bei UDP)	TSEND, TUSEND, ... (IoT ≤ 32 KByte) (TCP ≤ 64 KByte) (UDP = 1472 Byte)	TSEND_C (≤ 8192 Byte)	Ladebefehle/Transferbefehle (1, 2, 4 Byte) ----- DPR_DAT, DPWR_DAT PNIO_SEND, PNIO_RECV (IOC ≤ 8192 Byte) (IOD ≤ 1440 Byte)
remote Quittierung	BSEND: Applikation USEND: Transport PUT, GET: Applikation	ISO, IoT, TCP: Transport UDP: keine	IoT, TCP: Transport UDP: keine	Transport	Applikation
Routingfähig?	ISO: nein IoT: ja	ISO: nein sonst: ja	ja	ja	nein
Verbindungen?	ja	UDP: nein sonst: ja	UDP: nein sonst: ja	ja	nein

[Zurück zum Sprungverteiler PN/IE](#)

Copyright © Siemens AG 2011 All rights reserved
20982954_SIMATIC_Comm_DOKU_v201_d.doc

20 Auswahlhilfe: PROFIBUS (PB)

20.1 PB: Inhalt des Kapitels

Für das Medium PB wird beschrieben:

- Welche Schnittstellen (Baugruppen) und Kommunikationsarten stehen zur Verfügung?
(-> Tabelle Schnittstellen)
- Welche Partner können über welche Kommunikationsarten miteinander kommunizieren?
(-> Tabelle Kombinationen)
- Übersicht aller zur Verfügung stehenden Kommunikationsarten
(-> Tabelle Kommunikationsarten-Kompakt)

20.2 PB: Schnittstellen und Kommunikationsarten

20.2.1 ET 200 CPU an PB

Tabelle 20-1

Controller an PB: ET 200 CPU			Funktionalität der Schnittstelle		Kommunikationsart		
					SIMATIC spezifisch		Offener Standard
					S7-Basiskommunikation	S7-Kommunikation	DP-Kommunikation
ET 200 S	CPU	IM151-7 CPU	DP-Slave (*1)	aktiv	I_PUT, I_GET, Server (S7)	PUT, GET, Server (S7)	(2)
				passiv	I_PUT, I_GET, Server (S7)	---	(2)
	CP	DP-Mastermodul	DP-Master	I_PUT, I_GET, Client (S7)	PUT, GET, Server (S7)	(2)	
ET 200 Pro	CPU	IM154-8(F) PN/DP CPU	DP-Master		I_PUT, I_GET, Client (S7)	PUT, GET, Server (S7)	(2)
				DP-Slave (*1)	aktiv	I_PUT, I_GET, Server (S7)	PUT, GET, Server (S7)
		passiv	I_PUT, I_GET, Server (S7)		---	(2)	
	CP	DP-Mastermodul	DP-Master	I_PUT, I_GET, Client (S7)	PUT, GET, Server (S7)	(2)	

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

Erläuterungen zur Tabelle

(*1): Ein DP-Slave kann aktiv oder passiv sein. Ein aktiver DP-Slave erhält den Token, und ist damit auch aktiver Teilnehmer am PROFIBUS. Ein passiver DP-Slave erhält keinen Token. Die Einstellung erfolgt durch Parametrierung der Betriebsart der DP-Schnittstelle in STEP 7 HW-Konfiguration.

20.2.2 S7-300 an PB

Tabelle 20-2

Controller an PB: S7-300		Funktionalität der Schnittstelle		Kommunikationsart				
				SIMATIC spezifisch		Offener Standard		
				S7-Basiskommunikation	S7-Kommunikation	Offene-Kommunikation	FMS-Kommunikation	DP-Kommunikation
CPU	alle mit Schnittstelle: DP, MPI/DP (*2)	DP-Master		I_PUT, I_GET, Client (S7)	PUT, GET, Server (S7)	---	---	(2)
		DP-Slave (*3)	aktiv	I_PUT, I_GET, Server (S7)	PUT, GET, Server (S7)	---	---	(2)
			passiv	I_PUT, I_GET, Server (S7)	---	---	---	(2)
CP	342-5 (*1)	kein DP-Betrieb		---	(1) (S7)	(8) (FDL)	---	---
		DP-Master		---	(1) (S7)	(8) (FDL)	---	(7)
		DP-Slave (*3)	aktiv	---	PUT, GET, Server (S7)	(8) (FDL)	---	(7)
			passiv	---	---	---	---	(7)
	343-5	kein DP-Betrieb		---	PUT, GET, Server (S7)	(8) (FDL)	(10) (FMS)	---

[Zurück zum Sprungverteiler PB](#)Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(7) DP_SEND, DP_RECV

(8) AG_SEND/AG_RECV

(10) READ, WRITE, REPORT

Erläuterungen zur Tabelle

(*1): 342-5 steht für die Varianten: CP 342-5, CP 342-5 FO

(*2): MPI/DP Schnittstelle in der Betriebsart DP

(*3): Ein DP-Slave kann aktiv oder passiv sein. Ein aktiver DP-Slave erhält den Token, und ist damit auch aktiver Teilnehmer am PROFIBUS. Ein passiver DP-Slave erhält keinen Token. Die Einstellung erfolgt durch Parametrierung der Betriebsart der DP-Schnittstelle in STEP 7 HW-Konfiguration.

20.2.3 S7-400 an PB

Tabelle 20-3

Controller an PB: S7-400		Funktionalität der Schnittstelle		Kommunikationsart				
				SIMATIC spezifisch		Offener Standard		
				S7-Basiskommunikation	S7-Kommunikation	Offene-Kommunikation	FMS-Kommunikation	DP-Kommunikation
CPU	alle mit Schnittstelle: DP (*3), MPI/DP (*2)	DP-Master		I_PUT, I_GET, Client (S7)	(1) (S7)	---	---	(2)
		DP-Slave (*4)	aktiv	I_PUT, I_GET, Server (S7)	PUT, GET, Server, (S7)	---	---	(2)
			passiv	I_PUT, I_GET, Server (S7)	---	---	---	(2)
CP	443-5 Basic	kein DP-Betrieb		---	(1) (S7)	(8) (FDL)	(10) (FMS)	---
	443-5 Extended	kein DP-Betrieb		---	(1) (S7)	(8) (FDL)	---	---
		DP-Master		---	(1) (S7)	(8) (FDL)	---	(2)

[Zurück zum Sprungverteiler PB](#)Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
(8) AG_SEND/AG_RECV (10) READ, WRITE, REPORT

Erläuterungen zur Tabelle

(*2): MPI/DP Schnittstelle in der Betriebsart DP

(*3): integrierte Schnittstelle (X1, ...) oder steckbares Schnittstellenmodul (IF1, ...)

(*4): Ein DP-Slave kann aktiv oder passiv sein. Ein aktiver DP-Slave erhält den Token, und ist damit auch aktiver Teilnehmer am PROFIBUS. Ein passiver DP-Slave erhält keinen Token. Die Einstellung erfolgt durch Parametrierung der Betriebsart der DP-Schnittstelle in STEP 7 HW-Konfiguration.

20.2.4 S7-1200 an PB

S7-1200 hat keine Schnittstelle zu PROFIBUS!

[Zurück zum Sprungverteiler PB](#)

20.2.5 S7-mEC an PB

Tabelle 20-4

Controller an PB: S7-mEC (*1)			Funktionalität der Schnittstelle	Kommunikationsart	
				SIMATIC spezifisch	Offener Standard
S7-Kommunikation	DP-Kommunikation				
CP	EM PCI-104 (*3)	unter Kontrolle von RTX (Submodule) (*2): CP 5603	DP-Master	(1) (S7)	(2)

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

Erläuterungen zur Tabelle

(*1): mit vorinstalliertem WinAC RTX (F) 2009

(*2): Schnittstellen, unter der Kontrolle von WinAC RTX. (F). Die Schnittstellen müssen als „Submodule“ konfiguriert werden.

(*3) : steckbar sind maximal 3 Karten des PCI-104 Standards

20.2.6 Box PC an PB

Tabelle 20-5

Controller an PB: Box PC (*1) IPC427C Bundles mit RTX		Funktionalität der Schnittstelle	Kommunikationsart	
			SIMATIC spezifisch	Offener Standard
			S7-Kommunikation	DP-Kommunikation
CPU/CP (*2)	unter Kontrolle von RTX (Submodule) (*3): <ul style="list-style-type: none"> • „CP5611-CP5621“ • „CP5613-CP5603“ 	DP-Master	(1) (S7)	(2)

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine und Protokolle

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

Erläuterungen zur Tabelle

(*1): mit WinAC RTX (F) 2009

(*2): Bedeutung der Schnittstellen CPU bzw. CP: CPU = Onboard Schnittstelle des PC, CP = steckbare PC-Karte

Die Bezeichnungen der Schnittstellen in der Tabelle sind Sammelbezeichnungen. Die folgende Tabelle erläutert dies.

Tabelle 20-6

Schnittstelle	CP (steckbar in PC)	CPU (Onboard Schnittstelle des PC)
„CP5611-CP5621“	CP 5611-A2, CP5621	Onboard PB-Schnittstelle der SIMATIC PC: CP 5611 integriert
„CP5613-CP5603“	CP 5613, CP 5613-A2, CP 5603	---

(*3): Schnittstellen, unter der Kontrolle von WinAC RTX. Die Schnittstellen müssen als „Submodule“ konfiguriert werden.

20.2.7 Panel PC an PB

Tabelle 20-7

Controller an PB: Box PC (*1) HMI IPC477C Bundles mit RTX		Funktionalität der Schnittstelle	Kommunikationsart	
			SIMATIC spezifisch	Offener Standard
			S7-Kommunikation	DP-Kommunikation
CPU/CP (*2)	unter Kontrolle von RTX (Submodule) (*3): <ul style="list-style-type: none"> • „CP5611-CP5621“ • „CP5613-CP5603“ 	DP-Master	(1) (S7)	(2)

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine und Protokolle

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

Erläuterungen zur Tabelle

(*1): mit WinAC RTX (F) 2009

(*2): Bedeutung der Schnittstellen CPU bzw. CP: CPU = Onboard Schnittstelle des PC, CP = steckbare PC-Karte

Die Bezeichnungen der Schnittstellen in der Tabelle sind Sammelbezeichnungen. Die folgende Tabelle erläutert dies.

Tabelle 20-8

Schnittstelle	CP (steckbar in PC)	CPU (Onboard Schnittstelle des PC)
„CP5611-CP5621“	CP 5611-A2, CP5621	Onboard PB-Schnittstelle der SIMATIC PC: CP 5611 integriert
„CP5613-CP5603“	CP 5613, CP 5613-A2, CP 5603	---

(*3): Schnittstellen, unter der Kontrolle von WinAC RTX. Die Schnittstellen müssen als „Submodule“ konfiguriert werden.

20.2.8 WinAC MP an PB

Tabelle 20-9

Controller an PB: WinAC MP für Multipanel (*1) MP177, MP277, MP377		Funktionalität der Schnittstelle	Kommunikationsart	
			SIMATIC spezifisch	Offener Standard
			S7-Kommunikation	DP-Kommunikation
CPU	Unter Kontrolle von WinAC MP: Integrierte PB-Schnittstelle	DP-Master	(1) (S7)	(2)

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine und Protokolle

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

Erläuterungen zur Tabelle

(*1): mit WinAC MP 2008

20.2.9 WinAC RTX an PB

Tabelle 20-10

Controller an PB: WinAC RTX (*1)		Funktionalität der Schnittstelle	Kommunikationsart	
			SIMATIC spezifisch	Offener Standard
			S7-Kommunikation	DP-Kommunikation
CPU/CP (*2)	unter Kontrolle von RTX (Submodule) (*3): <ul style="list-style-type: none"> „CP5611-CP5621“ „CP5613-CP5603“ 	DP-Master	(1) (S7)	(2)

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine und Protokolle

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

Erläuterungen zur Tabelle

(*1): WinAC RTX (F) 2009 auf PC

(*2): Bedeutung der Schnittstellen CPU bzw. CP: CPU = Onboard Schnittstelle des PC, CP = steckbare PC-Karte

Die Bezeichnungen der Schnittstellen in der Tabelle sind Sammelbezeichnungen. Die folgende Tabelle erläutert dies.

Tabelle 20-11

Schnittstelle	CP (steckbar in PC)	CPU (Onboard Schnittstelle des PC)
„CP5611-CP5621“	CP 5611-A2, CP5621	Onboard PB-Schnittstelle der SIMATIC PC: CP 5611 integriert
„CP5613-CP5603“	CP 5613, CP 5613-A2, CP 5603	---

(*3): Schnittstellen, unter der Kontrolle von WinAC RTX. Die Schnittstellen müssen als „Submodule“ konfiguriert werden.

20.3 PB: Controller 1 = ET 200 CPU

20.3.1 ET 200 CPU / ET 200 CPU

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-12

Controller 2: ET 200 CPU				Controller 1: ET 200 CPU an PB														
				ET200 S						ET 200 Pro								
				CPU		CP		CPU						CP				
				IM151-7 CPU		DP-Mastermodul		IM154-8 (F) PN/DP CPU						DP-Mastermodul				
				Slave passiv		Master		Master			Slave passiv			Master				
				S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7-B	S7	DP	S7 B	S7	DP
ET200 S	CPU	IM151-7 CPU	Slave passiv	---	---	---	(62)	---	(2)	(62)	---	(2)	---	---	---	(62)	---	(2)
	CP	DP-Mastermodul	Master	(61)	---	(2)	---	---	---	---	---	(61)	---	(2)	---	---	---	
ET200 Pro	CPU	IM154-8(F) PN/DP CPU	Master	(61)	---	(2)	---	---	---	---	---	(61)	---	(2)	---	---	---	
			Slave passiv	---	---	---	(62)	---	(2)	(62)	---	(2)	---	---	---	(62)	---	(2)
	CP	DP-Mastermodul	Master	(61)	---	(2)	---	---	---	---	---	(61)	---	(2)	---	---	---	

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(61) Controller 1 ist Server (für I_PUT, I_GET)

(62) Controller 2 ist Server (für I_PUT, I_GET)

20.3.2 ET 200 CPU / S7-300

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-13

Controller 2: S7-300			Controller 1: ET 200 CPU an PB														
			ET200 S						ET 200 Pro								
			CPU			CP			CPU						CP		
			IM151-7 CPU			DP-Mastermodul (*1)			IM154-8 PN/DP CPU						DP-Mastermodul (*1)		
			Slave passiv			Master			Master			Slave passiv			Master		
			S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7-B	S7	DP	S7-B	S7	DP
CPU	alle mit Schnittstelle: DP, MPI/DP	Master	(61)	---	(2)	---	---	---	---	---	---	(61)	---	(2)	---	---	---
		Slave passiv	---	---	---	(62)	---	(2)	(62)	---	(2)	---	---	---	(62)	---	(2)
CP	342-5	kein DP	---	---	---	---	(21)	---	---	(21)	---	---	---	---	---	(21)	---
		Master	---	---	(2) / (7)	---	(21)	---	---	(21)	---	---	---	(2) / (7)	---	(21)	---
		Slave passiv	---	---	---	---	---	(2) / (7)	---	---	(2) / (7)	---	---	---	---	---	(2) / (7)
	343-5	kein DP	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(7) DP_SEND, DP_RECV

(21) Controller 1 ist Server (für PUT, GET)

(61) Controller 1 ist Server (für I_PUT, I_GET)

(62) Controller 2 ist Server (für I_PUT, I_GET)

Erläuterungen zur Tabelle

(*1): Das DP-Mastermodul verhält sich wie eine CPU Schnittstelle: Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

20.3.3 ET 200 CPU / S7-400

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-14

Controller 2: S7-400			Controller 1: ET 200 CPU an PB														
			ET200 S						ET 200 Pro								
			CPU			CP			CPU						CP		
			IM151-7 CPU			DP-Mastermodul			IM154-8 PN/DP CPU						DP-Mastermodul		
			Slave passiv			Master			Master			Slave passiv			Master		
			S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7-B	S7	DP
CPU	alle mit Schnittstelle: DP, MPI/DP	Master	(61)	---	(2)	---	(21)	---	---	(21)	---	(61)	---	(2)	---	(21)	---
		Slave passiv	---	---	---	(62)	---	(2)	(62)	---	(2)	---	---	---	(62)	---	(2)
CP	443-5 Basic	kein DP	---	---	---	---	(21)	---	---	(21)	---	---	---	---	---	(21)	---
		443-5 Ext.	kein DP	---	---	---	(21)	---	---	(21)	---	---	---	---	---	(21)	---
		Master	---	---	(2)	---	(21)	---	---	(21)	---	---	---	(2)	---	(21)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(21) Controller 1 ist Server (für PUT, GET)

(61) Controller 1 ist Server (für I_PUT, I_GET)

(62) Controller 2 ist Server (für I_PUT, I_GET)

20.3.4 ET 200 CPU / S7-mEC

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-15

Controller 2: S7-mEC				Controller 1: ET 200 CPU an PB														
				ET200 S						ET 200 Pro								
				CPU			CP			CPU						CP		
				IM151-7 CPU			DP-Mastermodul			IM154-8 PN/DP CPU						DP-Mastermodul		
				Slave passiv			Master			Master			Slave passiv			Master		
S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	
CP	EM PCI-104	Kontrolle von RTX (Submodule): CP5603	Master	---	---	(2)	---	(21)	---	---	(21)	---	---	(2)	---	(21)	---	

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(21) Controller 1 ist Server (für PUT, GET)

20.3.5 ET 200 CPU / Box PC

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-16

Controller 2: Box PC IPC427C Bundles mit RTX			Controller 1: ET 200 CPU an PB																	
			ET200 S									ET 200 Pro								
			CPU			CP			CPU						CP					
			IM151-7 CPU			DP-Mastermodul			IM154-8 PN/DP CPU						DP-Mastermodul					
			Slave passiv			Master			Master			Slave passiv			Master					
			S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP
CPU/CP	Kontrolle von RTX (Submodule): • „CP5611-CP5621“ • „CP5613-CP5603“		Master			---	---	(2)	---	(21)	---	---	(21)	---	---	---	(2)	---	(21)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(21) Controller 1 ist Server (für PUT, GET)

20.3.6 ET 200 CPU / Panel PC

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-17

Controller 2: Panel PC HMI IPC477C Bundles mit RTX			Controller 1: ET 200 CPU an PB																
			ET200 S						ET 200 Pro										
			CPU			CP			CPU						CP				
			IM151-7 CPU			DP-Mastermodul			IM154-8 PN/DP CPU						DP-Mastermodul				
			Slave passiv			Master			Master			Slave passiv			Master				
			S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7 B	S7
CPU/CP	Kontrolle von RTX (Submodule): • „CP5611-CP5621“ • „CP5613-CP5603“		Master		---	---	(2)	---	(21)	---	---	(21)	---	---	---	(2)	---	(21)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(21) Controller 1 ist Server (für PUT, GET)

20.3.7 ET 200 CPU / WinAC MP

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-18

Controller 2: WinAC MP für Multipanel MP177, MP277, MP377			Controller 1: ET 200 CPU an PB														
			ET200 S						ET 200 Pro								
			CPU			CP			CPU						DP		
			IM151-7 CPU			DP-Mastermodul			IM154-8 PN/DP CPU						DP-Mastermodul		
			Slave passiv			Master			Master			Slave passiv			Master		
			S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP
CPU	Kontrolle von WinAC MP: Integrierte PB-Schnittstelle	Master	---	---	(2)	---	(21)	---	---	(21)	---	---	---	(2)	---	(21)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(21) Controller 1 ist Server (für PUT, GET)

20.3.8 ET 200 CPU / WinAC RTX

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-19

Controller 2: WinAC RTX auf PC WinAC RTX 2009			Controller 1: ET 200 CPU an PB														
			ET200 S						ET 200 Pro								
			CPU			CP			CPU				CP				
			IM151-7 CPU			DP-Mastermodul			IM154-8 PN/DP CPU				DP-Mastermodul				
			Slave passiv			Master			Master				Slave passiv				Master
			S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP
CPU/CP	Kontrolle von RTX (Submodule): • „CP5611-CP5621“ • „CP5613-CP5603“	Master	---	---	(2)	---	(21)	---	---	(21)	---	---	---	(2)	---	(21)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(21) Controller 1 ist Server (für PUT, GET)

20.4 PB: Controller 1 = S7-300

20.4.1 (S7-300 / ET 200 CPU)

Siehe ET 200 CPU / S7-300: 20.3.2

20.4.2 S7-300 / S7-300

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-20

Controller 2: S7-300			Controller 1: S7-300 an PB																	
			CPU						CP											
			alle mit Schnittstelle: DP, MPI/DP						342-5						343-5					
			Master			Slave passiv			kein DP			Master			Slave passiv			kein DP		
			S7 B	S7	DP	S7 B	S7	DP	S7	OC	S7	OC	DP	S7	OC	DP	S7	OC	FMS	
CPU	alle mit Schnittstelle: DP, MPI/DP	Master	---	---	---	(61)	---	(2)	(22)	---	(22)	---	---	---	---	(7) / (2)	---	---	---	
		Slave passiv	(62)	---	(2)	---	---	---	---	---	---	---	---	(7) / (2)	---	---	---	---	---	---
CP	342-5	kein DP	---	(21)	---	---	---	---	(1)	(8)	(1)	(8)	---	---	---	---	(21)	(8)	---	
		Master	---	(21)	---	---	---	(2) / (7)	(1)	(8)	(1)	(8)	---	---	---	(7)	(21)	(8)	---	
		Slave passiv	---	---	(2) / (7)	---	---	---	---	---	---	---	(7)	---	---	---	---	---	---	
	343-5	kein DP	---	---	---	---	---	---	(22)	(8)	(22)	(8)	---	---	---	---	(8)	(10)		

[Zurück zum Sprungverteiler PB](#)Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(7) DP_SEND, DP_RECV

(10) READ, WRITE, REPORT

(21) Controller 1 ist Server (für PUT, GET)

(61) Controller 1 ist Server (für I_PUT, I_GET)

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(8) AG_SEND/AG_RECV

(22) Controller 2 ist Server (für PUT, GET)

(62) Controller 2 ist Server (für I_PUT, I_GET)

20.4.3 S7-300 / S7-400

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-21

Controller 2: S7-400			Controller 1: S7-300 an PB																
			CPU						CP										
			alle mit Schnittstelle: DP, MPI/DP						342-5						343-5				
			Master			Slave passiv			kein DP		Master			Slave passiv			kein DP		
			S7 B	S7	DP	S7 B	S7	DP	S7	OC	S7	OC	DP	S7	OC	DP	S7	OC	FMS
CPU	alle mit Schnittstelle: DP, MPI/DP	Master	---	(21)	---	(61)	---	(2)	(1)	---	(1)	---	---	---	---	(7)/(2)	(21)	---	---
		Slave passiv	(62)	---	(2)	---	---	---	---	---	---	---	---	(7)/(2)	---	---	---	---	---
CP	443-5 Basic	kein DP	---	(21)	---	---	---	---	(1)	(8)	(1)	(8)	---	---	---	---	(21)	(8)	(10)
		kein DP	---	(21)	---	---	---	---	(1)	(8)	(1)	(8)	---	---	---	---	(21)	(8)	---
		Master	---	(21)	---	---	---	(2)	(1)	(8)	(1)	(8)	---	---	---	(7)/(2)	(21)	(8)	---
CP	443-5 Ext.	kein DP	---	(21)	---	---	---	---	(1)	(8)	(1)	(8)	---	---	---	---	(21)	(8)	---
		kein DP	---	(21)	---	---	---	---	(1)	(8)	(1)	(8)	---	---	---	---	(21)	(8)	---
		Master	---	(21)	---	---	---	(2)	(1)	(8)	(1)	(8)	---	---	---	(7)/(2)	(21)	(8)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(7) DP_SEND, DP_RECV

(10) READ, WRITE, REPORT

(21) Controller 1 ist Server (für PUT, GET)

(61) Controller 1 ist Server (für I_PUT, I_GET)

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(8) AG_SEND/AG_RECV

(62) Controller 2 ist Server (für I_PUT, I_GET)

20.4.4 S7-300 / S7-mEC

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-22

Controller 2: S7-400				Controller 1: S7-300 an PB																	
				CPU						CP											
				alle mit Schnittstelle: DP, MPI/DP						342-5										343-5	
				Master			Slave passiv			kein DP		Master			Slave passiv			kein DP			
				S7 B	S7	DP	S7 B	S7	DP	S7	OC	S7	OC	DP	S7	OC	DP	S7	OC	FMS	
CP	EM PCI-104	Kontrolle von RTX (Submodule): CP5603	Master	---	(21)	---	---	---	(2)	(1)	---	(1)	---	---	---	---	(2)	(21)	---	---	

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (21) Controller 1 ist Server (für PUT, GET)

20.4.5 S7-300 / Box PC

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-23

Controller 2: S7-400 IPC427C Bundles mit RTX			Controller 1: S7-300 an PB																	
			CPU						CP											
			alle mit Schnittstelle: DP, MPI/DP						342-5						343-5					
			Master			Slave passiv			kein DP			Master			Slave passiv			kein DP		
			S7 B	S7	DP	S7 B	S7	DP	S7	OC	S7	OC	DP	S7	OC	DP	S7	OC	FMS	
CPU/CP	Kontrolle von RTX (Submodule):	Master	---	(21)	---	---	---	(2)	(1)	---	(1)	---	---	---	---	(2)	(21)	---	---	
	<ul style="list-style-type: none"> „CP5611-CP5621“ „CP5613-CP5603“ 																			

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (21) Controller 1 ist Server (für PUT, GET)

20.4.6 S7-300 / Panel PC

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-24

Controller 2: S7-400 HMI IPC477C Bundles mit RTX			Controller 1: S7-300 an PB																	
			CPU						CP											
			alle mit Schnittstelle: DP, MPI/DP						342-5						343-5					
			Master			Slave passiv			kein DP			Master			Slave passiv			kein DP		
			S7 B	S7	DP	S7 B	S7	DP	S7	OC	S7	OC	DP	S7	OC	DP	S7	OC	FMS	
CPU/CP	Kontrolle von RTX (Submodule): • „CP5611-CP5621“ • „CP5613-CP5603“		Master	---	(21)	---	---	---	(2)	(1)	---	(1)	---	---	---	(2)	(21)	---	---	

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (21) Controller 1 ist Server (für PUT, GET)

20.4.7 S7-300 / WinAC MP

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-25

Controller 2: S7-400 MP177, MP277, MP377			Controller 1: S7-300 an PB																	
			CPU						CP											
			alle mit Schnittstelle: DP, MPI/DP						342-5						343-5					
			Master			Slave passiv			kein DP			Master			Slave passiv			kein DP		
			S7 B	S7	DP	S7 B	S7	DP	S7	OC	S7	OC	DP	S7	OC	DP	S7	OC	FMS	
CPU	Kontrolle von WinAC MP: Integrierte PB-Schnittstelle	Master	---	(21)	---	---	---	(2)	(1)	---	(1)	---	---	---	---	(2)	(21)	---	---	

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (21) Controller 1 ist Server (für PUT, GET)

20.4.8 S7-300 / WinAC RTX

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-26

Controller 2: S7-400 WinAC RTX 2009			Controller 1: S7-300 an PB																	
			CPU						CP											
			alle mit Schnittstelle: DP, MPI/DP						342-5						343-5					
			Master			Slave passiv			kein DP			Master			Slave passiv			kein DP		
			S7 B	S7	DP	S7 B	S7	DP	S7	OC	S7	OC	DP	S7	OC	DP	S7	OC	FMS	
CPU/CP	Kontrolle von RTX (Submodule): • „CP5611-CP5621“ • „CP5613-CP5603“		Master	---	(21)	---	---	---	(2)	(1)	---	(1)	---	---	---	---	(2)	(21)	---	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
- (21) Controller 1 ist Server (für PUT, GET)

20.5 PB: Controller 1 = S7-400

20.5.1 (S7-400 / ET 200 CPU)

Siehe ET 200 CPU / S7-400: 20.3.3

20.5.2 (S7-400 / S7-300)

Siehe S7-300 / S7-400: 20.4.3

20.5.3 S7-400 / S7-400

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-27

Controller 2: S7-400			Controller 1: S7-400 an PB													
			CPU							CP						
			alle mit Schnittstelle: DP, MPI/DP							443-5 Basic			443-5 Ext.			
			Master			Slave passiv				kein DP			kein DP		Master	
			S7 B	S7	DP	S7 B	S7	DP	S7	OC	FMS	S7	OC	S7	OC	DP
CPU	alle mit Schnittstelle: DP, MPI/DP	Master	---	(1)	---	(21)	---	(2)	(1)	---	---	(1)	---	(1)	---	---
		Slave passiv	(22)	---	(2)	---	---	---	---	---	---	---	---	---	---	---
CP	443-5 Basic	kein DP	---	(1)	---	---	---	---	(1)	(8)	(10)	(1)	(8)	(1)	(8)	---
		kein DP	---	(1)	---	---	---	---	(1)	(8)	---	(1)	(8)	(1)	(8)	---
		Master	---	(1)	---	---	---	(2)	(1)	(8)	---	(1)	(8)	(1)	(8)	---
CP	443-5 Ext.	kein DP	---	(1)	---	---	---	---	(1)	(8)	---	(1)	(8)	(1)	(8)	---
		kein DP	---	(1)	---	---	---	---	(1)	(8)	---	(1)	(8)	(1)	(8)	---
		Master	---	(1)	---	---	---	(2)	(1)	(8)	---	(1)	(8)	(1)	(8)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(7) DP_SEND, DP_RECV

(8) AG_SEND/AG_RECV

(21) Controller 1 ist Server (für PUT, GET)

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(10) READ, WRITE, REPORT

(22) Controller 2 ist Server (für PUT, GET)

20.5.4 S7-400 / S7-mEC

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-28

Controller 2: S7-400				Controller 1: S7-400 an PB													
				CPU						CP							
				alle mit Schnittstelle: DP, MPI/DP						443-5 Basic			443-5 Ext.				
				Master			Slave passiv			kein DP			kein DP		Master		
				S7 B	S7	DP	S7 B	S7	DP	S7	OC	FMS	S7	OC	S7	OC	DP
CP	EM PCI-104	Kontrolle von RTX (Submodule): CP5603	Master	---	(1)	---	---	---	(2)	(1)	---	---	(1)	---	(1)	---	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

20.5.5 S7-400 / Box PC

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-29

Controller 2: S7-400 IPC427C Bundles mit RTX			Controller 1: S7-400 an PB													
			CPU						CP							
			alle mit Schnittstelle: DP, MPI/DP						443-5 Basic			443-5 Ext.				
			Master			Slave passiv			kein DP			kein DP		Master		
			S7 B	S7	DP	S7 B	S7	DP	S7	OC	FMS	S7	OC	S7	OC	DP
CPU/CP	Kontrolle von RTX (Submodule): • „CP5611-CP5621“ • „CP5613-CP5603“	Master	---	(1)	---	---	---	(2)	(1)	---	---	(1)	---	(1)	---	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

20.5.6 S7-400 / Panel PC

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-30

Controller 2: S7-400 HMI IPC477C Bundles mit RTX			Controller 1: S7-400 an PB													
			CPU						CP							
			alle mit Schnittstelle: DP, MPI/DP						443-5 Basic			443-5 Ext.				
			Master			Slave passiv			kein DP			kein DP		Master		
			S7 B	S7	DP	S7 B	S7	DP	S7	OC	FMS	S7	OC	S7	OC	DP
CPU/CP	Kontrolle von RTX (Submodule): • „CP5611-CP5621“ • „CP5613-CP5603“	Master	---	(1)	---	---	---	(2)	(1)	---	---	(1)	---	(1)	---	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

20.5.7 S7-400 / WinAC MP

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-31

Controller 2: S7-400 MP177, MP277, MP377			Controller 1: S7-400 an PB														
			CPU							CP							
			alle mit Schnittstelle: DP, MPI/DP							443-5 Basic			443-5 Ext.				
			Master			Slave passiv				kein DP			kein DP		Master		
			S7 B	S7	DP	S7 B	S7	DP	S7	OC	FMS	S7	OC	S7	OC	DP	
CPU	Kontrolle von WinAC MP: Integrierte PB-Schnittstelle	Master	---	(1)	---	---	---	(2)	(1)	---	---	(1)	---	(1)	---	---	

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

- (1) USEND/URCV, BSEND/BRCV, PUT, GET
- (2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

20.5.8 S7-400 / WinAC RTX

In der folgenden Tabelle wird angenommen, dass die DP-Slave Schnittstelle „passiv“ ist. Wird die DP-Slave Schnittstelle als „aktiv“ betrieben, dann ergeben sich zusätzliche Kommunikationsmöglichkeiten (Siehe Kapitel 20.2).

Tabelle 20-32

Controller 2: S7-400 WinAC RTX 2009			Controller 1: S7-400 an PB													
			CPU						CP							
			alle mit Schnittstelle: DP, MPI/DP						443-5 Basic			443-5 Ext.				
			Master			Slave passiv			kein DP			kein DP		Master		
			S7 B	S7	DP	S7 B	S7	DP	S7	OC	FMS	S7	OC	S7	OC	DP
CPU/CP	Kontrolle von RTX (Submodule): • „CP5611-CP5621“ • „CP5613-CP5603“	Master	---	(1)	---	---	---	(2)	(1)	---	---	(1)	---	(1)	---	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

20.6 PB: Controller 1 = S7-mEC

20.6.1 (S7-mEC / ET 200 CPU)

Siehe ET 200 CPU / S7-mEC: 20.3.4

20.6.2 (S7-mEC / S7-300)

Siehe S7-300 / S7-mEC: 20.4.4

20.6.3 (S7-mEC / S7-400)

Siehe S7-400 / S7-mEC: 20.5.4

20.6.4 S7-mEC / S7-mEC

Tabelle 20-33

Controller 2: S7-mEC				Controller 1: S7-mEC an PB	
				CP	
				EM PCI-104	
				Kontrolle von RTX (Submodule): CP5603	
				Master	
				S7	DP
CP	EM PCI-104	Kontrolle von RTX (Submodule): CP5603	Master	(1)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

20.6.5 S7-mEC / Box PC

Tabelle 20-34

Controller 2: Box PC IPC427C Bundles mit RTX			Controller 1: S7-mEC an PB	
			CP	
			EM PCI-104	
			Kontrolle von RTX (Submodule): CP5603	
			Master	
CPU/CP	Kontrolle von RTX (Submodule):	Master	S7	DP
	<ul style="list-style-type: none"> • „CP5611-CP5621“ • „CP5613-CP5603“ 		(1)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

20.6.6 S7-mEC / Panel PC

Tabelle 20-35

Controller 2: Panel PC HMI IPC477C Bundles mit RTX			Controller 1: S7-mEC an PB	
			CP	
			EM PCI-104	
			Kontrolle von RTX (Submodule): CP5603	
			Master	
			S7	DP
CPU/CP	Kontrolle von RTX (Submodule): <ul style="list-style-type: none"> • „CP5611-CP5621“ • „CP5613-CP5603“ 	Master	(1)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

20.6.7 S7-mEC / WinAC MP

Tabelle 20-36

Controller 2: WinAC MP für Multipanel MP177, MP277, MP377			Controller 1: S7-mEC an PB	
			CP	
			EM PCI-104	
			Kontrolle von RTX (Submodule): CP5603	
			Master	
			S7	DP
CPU	Kontrolle von WinAC MP: Integrierte PB-Schnittstelle	Master	(1)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

20.6.8 S7-mEC / WinAC RTX

Tabelle 20-37

Controller 2: WinAC RTX WinAC RTX 2009			Controller 1: S7-mEC an PB	
			EM PCI-104	
			Kontrolle von RTX (Submodule): CP5603	
			EM PCI-104	
			Master	
			S7	DP
CPU/CP	Kontrolle von RTX (Submodule): • „CP5611-CP5621“ • „CP5613-CP5603“	Master	(1)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

20.7 PB: Controller 1 = Box PC

20.7.1 (Box PC / ET 200 CPU)

Siehe ET 200 CPU / Box PC: 20.3.5

20.7.2 (Box PC / S7-300)

Siehe S7-300 / Box PC: 20.4.5

20.7.3 (Box PC / S7-400)

Siehe S7-400 / Box PC: 20.5.5

20.7.4 (Box PC / S7-mEC)

Siehe S7-mEC / Box PC: 20.6.5

20.7.5 Box PC / Box PC

Tabelle 20-38

Controller 2: Box PC IPC427C Bundles mit RTX			Controller 1: Box PC an PB	
			IPC427C Bundles mit RTX	
			CPU/CP	
			Kontrolle von RTX (Submodule): <ul style="list-style-type: none"> • „CP5611-CP5621“ • „CP5613-CP5603“ 	
			Master	
			S7	DP
CPU/CP	Kontrolle von RTX (Submodule): <ul style="list-style-type: none"> • „CP5611-CP5621“ • CP5613-CP5603“ 	Master	(1)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

20.7.6 Box PC / Panel PC

Tabelle 20-39

Controller 2: Panel PC HMI IPC477C Bundles mit RTX			Controller 1: Box PC an PB	
			IPC427C Bundles mit RTX	
			CPU/CP	
			Kontrolle von RTX (Submodule): <ul style="list-style-type: none"> • „CP5611-CP5621“ • „CP5613-CP5603“ 	
			Master	
			S7	DP
CPU/CP	Kontrolle von RTX (Submodule): <ul style="list-style-type: none"> • „CP5611-CP5621“ • „CP5613-CP5603“ 	Master	(1)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

20.7.7 Box PC / WinAC MP

Tabelle 20-40

Controller 2: WinAC MP MP177, MP277, MP377			Controller 1: Box PC an PB	
			IPC427C Bundles mit RTX	
			CPU/CP	
			Kontrolle von RTX (Submodule):	
			<ul style="list-style-type: none"> • „CP5611-CP5621“ • „CP5613-CP5603“ 	
Master			S7	DP
CPU	Kontrolle von WinAC MP: Integrierte PB-Schnittstelle	Master	(1)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

20.7.8 Box PC / WinAC RTX

Tabelle 20-41

Controller 2: WinAC RTX WinAC RTX 2009			Controller 1: Box PC an PB	
			IPC427C Bundles mit RTX	
			CPU/CP	
			Kontrolle von RTX (Submodule):	
			<ul style="list-style-type: none"> • „CP5611-CP5621“ • „CP5613-CP5603“ 	
			Master	
			S7	DP
CPU/CP	Kontrolle von RTX (Submodule):	Master	(1)	---
<ul style="list-style-type: none"> • „CP5611-CP5621“ • „CP5613-CP5603“ 				

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

20.8 PB: Controller 1 = Panel PC

20.8.1 (Panel PC / ET 200 CPU)

Siehe ET 200 CPU / Panel PC: 20.3.6

20.8.2 (Panel PC / S7-300)

Siehe S7-300 / Panel PC: 20.4.6

20.8.3 (Panel PC / S7-400)

Siehe S7-400 / Panel PC: 20.5.6

20.8.4 (Panel PC / S7-mEC)

Siehe S7-mEC / Panel PC: 20.6.6

20.8.5 (Panel PC / Box PC)

Siehe Box PC / Panel PC: 20.7.6

20.8.7 Panel PC / WinAC MP

Tabelle 20-43

Controller 2: WinAC MP MP177, MP277, MP377			Controller 1: Panel PC an PB	
			HMI IPC477C Bundles mit RTX	
			CPU/CP	
			Kontrolle von RTX (Submodule):	
			<ul style="list-style-type: none"> • „CP5611-CP5621“ • „CP5613-CP5603“ 	
			Master	
			S7	DP
CPU	Kontrolle von WinAC MP: Integrierte PB-Schnittstelle	Master	(1)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

20.8.8 Panel PC / WinAC RTX

Tabelle 20-44

Controller 2: WinAC RTX WinAC RTX 2009			Controller 1: Panel PC an PB	
			HMI IPC477C Bundles mit RTX	
			CPU/CP	
			Kontrolle von RTX (Submodule): <ul style="list-style-type: none"> • „CP5611-CP5621“ • „CP5613-CP5603“ 	
			Master	
			S7	DP
CPU/CP	Kontrolle von RTX (Submodule): <ul style="list-style-type: none"> • „CP5611-CP5621“ • „CP5613-CP5603“ 	Master	(1)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

20.9 PB: Controller 1 = WinAC MP

20.9.1 (WinAC MP / ET 200 CPU)

Siehe ET 200 CPU / WinAC MP: 20.3.7

20.9.2 (WinAC MP / S7-300)

Siehe S7-300 / WinAC MP: 20.4.7

20.9.3 (WinAC MP / S7-400)

Siehe S7-400 / WinAC MP: 20.5.7

20.9.4 (WinAC MP / S7-mEC)

Siehe S7-mEC / WinAC MP: 20.6.7

20.9.5 (WinAC MP / Box PC)

Siehe Box PC / WinAC MP: 20.7.7

20.9.6 (WinAC MP / Panel PC)

Siehe Panel PC / WinAC MP: 20.8.7

20.9.7 WinAC MP / WinAC MP

Tabelle 20-45

Controller 2: WinAC MP MP177, MP277, MP377			Controller 1: WinAC MP an PB	
			MP177, MP277, MP377	
			CPU	
			Kontrolle von WinAC MP: Integrierte PB-Schnittstelle	
			Master	
			S7	DP
CPU	Kontrolle von WinAC MP: Integrierte PB-Schnittstelle	Master	(1)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

20.9.8 WinAC MP / WinAC RTX

Tabelle 20-46

Controller 2: WinAC RTX WinAC RTX 2009			Controller 1: WinAC MP an PB	
			MP177, MP277, MP377	
			CPU	
			Kontrolle von WinAC MP: Integrierte PB-Schnittstelle	
			Master	
			S7	DP
CPU/CP	Kontrolle von RTX (Submodule): • „CP5611-CP5621“ • „CP5613-CP5603“	Master	(1)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

20.10 PB: Controller 1 = WinAC RTX

20.10.1 (WinAC RTX / ET 200 CPU)

Siehe ET 200 CPU / WinAC RTX: 20.3.8

20.10.2 (WinAC RTX / S7-300)

Siehe S7-300 / WinAC RTX: 20.4.8

20.10.3 (WinAC RTX / S7-400)

Siehe S7-400 / WinAC RTX: 20.5.8

20.10.4 (WinAC RTX / S7-mEC)

Siehe S7-mEC / WinAC RTX: 20.6.8

20.10.5 (WinAC RTX / Box PC)

Siehe Box PC / WinAC RTX: 20.7.8

20.10.6 (WinAC RTX / Panel PC)

Siehe Panel PC / WinAC RTX: 20.8.8

20.10.7 (WinAC RTX / WinAC MP)

Siehe WinAC MP / WinAC RTX: 20.9.8

20.10.8 WinAC RTX / WinAC RTX

Tabelle 20-47

Controller 2: WinAC RTX WinAC RTX 2009			Controller 1: WinAC RTX an PB	
			WinAC RTX 2009	
			CPU/CP	
			Kontrolle von RTX (Submodule): <ul style="list-style-type: none"> • „CP5611-CP5621“ • „CP5613-CP5603“ 	
			Master	
CPU/CP	Kontrolle von RTX (Submodule): <ul style="list-style-type: none"> • „CP5611-CP5621“ • „CP5613-CP5603“ 	Master	S7	DP
			(1)	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

20.11 PB: Übersicht Kommunikationsarten

Gegenüberstellung aller über PB verfügbaren Kommunikationsarten.

Die Daten sind den Tabellen Kommunikationsarten-Detail entnommen:

- S7-Basiskommunikation (Tabelle 28-1)
- S7-Kommunikation (Tabelle 29-1)
- Offene Kommunikation mit Send/Receive-Bausteinen (Tabelle 36-1)
- FMS-Kommunikation (Tabelle 37-1)
- DP-Kommunikation Tabelle 38-1)

Tabelle Kommunikationsarten-Kompakt:

Tabelle 20-48

	SIMATIC spezifisch		Offener Standard		DP-Kommunikation
	S7-Basiskommunikation	S7-Kommunikation	Offene-Kommunikation	FMS-Kommunikation	
Protokolle	S7 (DP)	S7 (FDL)	FDL	FMS	DP
Schnittstellen	CPU	CPU, CP	CP	CP	CPU, CP
Kommunikationsbausteine (max. Daten)	I_PUT (= 84 Bytes) I_GET (= 94 Bytes)	BSEND (<= 64 KByte) USEND (>= 160 Byte) PUT, GET (>= 160 Byte)	AG_SEND (=240 Byte) AG_LSEND (=240 Byte)	READ (<= 237 Byte) WRITE (<= 233 Byte) REPORT (<= 233 Byte)	Ladebefehle / Transferbefehle (1, 2, 4 Byte) DPR_DAT, DPWR_DAT (<= 64 Worte) DP_SEND, DP_RECV (<= 244 Byte)
remote Quittierung	Applikation	BSEND: Applikation USEND: Transport PUT, GET: Applikation	Transport	READ: Applikation WRITE: Applikation REPORT: nein	Applikation
Routingfähig?	nein	nein	nein	nein	nein
Verbindungen?	ja	ja	ja	ja	nein

[Zurück zum Sprungverteiler PB](#)

21 Auswahlhilfe: MPI (MPI)

21.1 MPI: Inhalt des Kapitels

Für das Medium MPI wird beschrieben:

- Welche Schnittstellen (Baugruppen) und Kommunikationsarten stehen zur Verfügung?
(-> Tabelle Schnittstellen)
- Welche Partner können über welche Kommunikationsarten miteinander kommunizieren?
(-> Tabelle Kombinationen)
- Übersicht aller zur Verfügung stehenden Kommunikationsarten
(-> Tabelle Kommunikationsarten-Kompakt)

21.2 MPI: Schnittstellen und Kommunikationsarten

21.2.1 ET 200 CPU an MPI

Tabelle 21-1

Controller an MPI: ET 200 CPU			Kommunikationsart		
			SIMATIC spezifisch		
			Globaldaten	S7-Basiskommunikation	S7-Kommunikation
ET 200 S	CPU	IM151-7 CPU	zyklisch	(11)	PUT, GET, Server, (S7)
ET 200 Pro	CPU	IM154-8(F) PN/DP CPU	zyklisch	(11)	PUT, GET, Server, (S7)

[Zurück zum Sprungverteiler MPI](#)

Kommunikationsbausteine

(11) X_SEND/X_RCV, X_PUT, X_GET

21.2.2 S7-300 an MPI

Tabelle 21-2

Controller an MPI: S7-300		Kommunikationsart		
		SIMATIC spezifisch		
		Globaldaten	S7-Basiskommunikation	S7-Kommunikation
CPU	alle mit Schnittstelle: MPI, MPI/DP (*2)	zyklisch	(11)	PUT, GET, Server, (S7)

[Zurück zum Sprungverteiler MPI](#)

Kommunikationsbausteine

(11) X_SEND/X_RCV, X_PUT, X_GET

Erläuterungen zur Tabelle

(*2): MPI/DP Schnittstelle in der Betriebsart MPI

21.2.3 S7-400 an MPI

Tabelle 21-3

Controller an MPI: S7-400		Kommunikationsart		
		SIMATIC spezifisch		
		Globaldaten	S7-Basiskommunikation	S7-Kommunikation
CPU	alle mit Schnittstelle: MPI/DP (*2)	zyklisch azyklisch (12)	(11)	(1)

[Zurück zum Sprungverteiler MPI](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(11) X_SEND/X_RCV, X_PUT, X_GET

(12) GD_SND/GD_RCV

Erläuterungen zur Tabelle

(*2): MPI/DP Schnittstelle in der Betriebsart MPI

21.3 MPI: Controller 1 = ET 200 CPU

21.3.1 ET 200 CPU / ET 200 CPU

Tabelle 21-4

Controller 2: ET 200 CPU		Controller 1: ET 200 CPU an MPI		
		CPU		
		IM151-7 CPU, IM154-8(F) PN/DP CPU		
		GD	S7 Basis	S7
CPU	151-7, 154-8	zyklisch	(11)	---

[Zurück zum Sprungverteiler MPI](#)

Kommunikationsbausteine

(11) X_SEND/X_RCV, X_PUT, X_GET

21.3.2 ET 200 CPU / S7-300

Tabelle 21-5

Controller 2: S7-300		Controller 1: ET 200 CPU an MPI		
		CPU		
		IM151-7 CPU, IM154-8(F) PN/DP CPU		
		GD	S7 Basis	S7
CPU	alle mit Schnittstelle: MPI, MPI/DP	zyklisch	(11)	---

[Zurück zum Sprungverteiler MPI](#)

Kommunikationsbausteine

(11) X_SEND/X_RCV, X_PUT, X_GET

21.3.3 ET 200 CPU / S7-400

Tabelle 21-6

Controller 2: S7-400		Controller 1: ET 200 CPU an MPI		
		CPU		
		IM151-7 CPU, IM154-8(F) PN/DP CPU		
		GD	S7 Basis	S7
CPU	alle mit Schnittstelle: MPI/DP	zyklisch + C2 azyklisch (12)	(11)	(21)

[Zurück zum Sprungverteiler MPI](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(11) X_SEND/X_RCV, X_PUT, X_GET

(12) GD_SND/GD_RCV

(21) Controller 1 Server (für PUT, GET)

21.4 MPI: Controller 1 = S7-300

21.4.1 (S7-300 / ET 200 CPU)

Siehe ET 200 CPU / S7-300: 21.3.2

21.4.2 S7-300 / S7-300

Tabelle 21-7

Controller 2: S7-300		Controller 1: S7-300 an MPI		
		CPU		
alle mit Schnittstelle: MPI, MPI/DP		alle mit Schnittstelle: MPI, MPI/DP		
		GD	S7 Basis	S7
CPU	alle mit Schnittstelle: MPI, MPI/DP	zyklisch	(11)	---

[Zurück zum Sprungverteiler MPI](#)

Kommunikationsbausteine

(11) X_SEND/X_RCV, X_PUT, X_GET

21.4.3 S7-300 / S7-400

Tabelle 21-8

Controller 2: S7-400		Controller 1: S7-300 an MPI		
		CPU		
alle mit Schnittstelle: MPI/DP		alle mit Schnittstelle: MPI, MPI/DP		
		GD	S7 Basis	S7
CPU	alle mit Schnittstelle: MPI/DP	zyklisch + C2 azyklisch (12)	(11)	(21)

[Zurück zum Sprungverteiler MPI](#)

Kommunikationsbausteine

(11) X_SEND/X_RCV, X_PUT, X_GET

(12) GD_SND/GD_RCV

(21) Controller 1 Server (für PUT, GET)

21.5 MPI: Controller 1 = S7-400

21.5.1 (S7-400 / ET 200 CPU)

Siehe ET 200 CPU / S7-400: 21.3.3

21.5.2 (S7-400 / S7-300)

Siehe S7-300 / S7-400: 21.4.3

21.5.3 S7-400 / S7-400

Tabelle 21-9

Controller 2: S7-400		Controller 1: S7-400 an MPI		
		CPU		
CPU		alle mit Schnittstelle: MPI/DP		
		GD	S7 Basis	S7
	alle mit Schnittstelle: MPI/DP	zyklisch + azyklisch (12)	(11)	(1)

[Zurück zum Sprungverteiler MPI](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(11) X_SEND/X_RCV, X_PUT, X_GET

(12) GD_SND/GD_RCV

21.6 MPI: Übersicht Kommunikationsarten

Gegenüberstellung aller über MPI verfügbaren Kommunikationsarten.

Die Daten sind den Tabellen Kommunikationsarten-Detail entnommen:

- Globaldaten-Kommunikation (Tabelle 27-1)
- S7-Basiskommunikation (Tabelle 28-1)
- S7-Kommunikation (Tabelle 29-1)

Tabelle Kommunikationsarten-Kompakt:

Tabelle 21-10

	SIMATIC spezifisch		
	Globaldaten-Kommunikation	S7-Basiskommunikation	S7-Kommunikation
Protokolle	S7	S7	S7 (FDL)
Schnittstellen	CPU	CPU	CPU, CP
Kommunikationsbausteine (max. Daten)	zyklisch: keine Kommunikationsbausteine ----- azyklisch: GD_SND, GD_RCV (22 Byte pro GD-Paket)	X_PUT (= 76 Bytes) X_GET (= 76 Bytes)	BSEND (<= 64 KByte) USEND (>= 160 Byte) PUT, GET (>= 160 Byte)
remote Quittierung	nein	Applikation	BSEND: Applikation USEND: Transport PUT, GET: Applikation
Routingfähig?	nein	nein	nein
Verbindungen?	nein	ja	ja

[Zurück zum Sprungverteiler MPI](#)

22 Auswahlhilfe: SIMATIC Rückwandbus

22.1 Inhalt des Kapitels

Für das Medium SIMATIC Rückwandbus wird beschrieben::

- Welche Schnittstellen (Baugruppen) stehen zur Verfügung?
- Welche Partner können über welche Kommunikationsarten miteinander kommunizieren? (*1)
- Übersicht aller zur Verfügung stehenden Kommunikationsarten

(*1) Betrachtete Kombinationen Controller 1 / Controller 2:

Bei der S7-400 können bis zu 4 CPUs in einem einzigen Controller betrieben werden (Multicomputing, Kapitel 4.5). D.h. in den folgenden Tabellen gibt es keine Unterscheidung zwischen Controller 1 und Controller 2. Die CPUs stecken im gleichen Controller!

22.2 Schnittstellen und Kommunikationsarten

Tabelle 22-1

CPU am Rückwandbus: S7-400		Kommunikationsart	
		SIMATIC	
		Globaldaten-Kommunikation	S7-Kommunikation
CPU	alle	+ azyklisch	(1)

[Zurück zum Sprungverteiler Rückwandbus](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

22.3 Controller 1 = S7-400 / Controller 2 = S7-400

Tabelle 22-2

Controller: S7-400		Controller: S7-400	
		CPU	
		alle	
		GD	S7
CPU	alle	zyklisch, azyklisch (12)	(1)

[Zurück zum Sprungverteiler Rückwandbus](#)

Kommunikationsbausteine

(1) USEND/URCV, BSEND/BRCV, PUT, GET

(12) GD_SND/GD_RCV

22.4 Übersicht Kommunikationsarten

Gegenüberstellung aller über den SIMATIC Rückwandbus verfügbaren Kommunikationsarten.

Die Daten sind den Tabellen Kommunikationsarten-Detail entnommen:

- Globaldaten-Kommunikation (Tabelle 27-1)
- S7-Basiskommunikation (Tabelle 28-1)

Tabelle Kommunikationsarten-Kompakt:

Tabelle 22-3

Kommunikationsart	SIMATIC	
	Globaldaten-Kommunikation	S7-Kommunikation
Protokolle	S7-Protokoll	S7-Protokoll
Schnittstellen	CPU	CPU
Kommunikationsbausteine (max. Daten)	zyklische Übertragung: keine Kommunikationsbausteine erforderlich azyklische Übertragung: GD_SND, GD_RCV (54Byte pro GD-Paket)	BSEND (<= 64 KByte) USEND (>= 160 Byte) PUT, GET (>= 160 Byte)
remote Quittierung	keine	BSEND: Applikation USEND: Transport PUT, GET: Applikation
Routingfähig?	---	---
Verbindungen ?	nein	ja

[Zurück zum Sprungverteiler Rückwandbus](#)

23 Auswahlhilfe: Serielle Schnittstelle (PtP)

23.1 Inhalt des Kapitels

Im vorliegenden Kapitel wird beschrieben:

- Welche Möglichkeiten bieten die SIMATIC Familien zur Kommunikation über eine „Serielle Schnittstelle“?
- Eine Betrachtung der Kombinationen von SIMATIC Familien (wie bei den Netzen PN/IE, PB und MPI) wird nicht vorgenommen.

Betrachtete Schnittstellen:

- Modulare Controller: ET 200 CPU, S7-300, S7-400, S7-1200
- Embedded Controller: S7-mEC
- Dezentrale Station

23.2 ET 200 CPU an PtP

Tabelle 23-1

Controller an PtP: ET 200 CPU			Protokoll		maximale Anzahl Daten	Physik (max. Ab- stand Kommunikati- onspartner)	Übertragungsrate
			Typ	auf CP integriert / ladbar			
ET 200S	CP	Modbus/USS-Modul	Modbus Seriell (RTU Format), Modbus Master und Modbus Slave	integriert	224 Byte	RS 232C (15 m) RS 422/485 (1200 m)	110 Bit/s bis 115,2 KBit/s
			USS Protokoll (*1)				
		1SI-Modul	3964(R) ASCII	integriert			

Zusätzliche Möglichkeit: In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden. Siehe 23.11.

[Zurück zum Sprungverteiler SERIELL](#)

Erläuterungen zur Tabelle

(*1): Wird im Dokument nicht weiter betrachtet, da damit keine CPU-CPU Kommunikation realisiert werden kann.

23.3 S7-300 an PtP

Tabelle 23-2

Controller an PtP: S7-300		Protokoll		maximale Anzahl Daten	Physik (max. Abstand Kommunikati- onspartner)	Übertragungsrate
		Typ	auf CP integriert / ladbar			
CPU	313C-2 PtP	3964(R), ASCII	integriert	1024 Byte	RS 422/485 (1200 m)	300 Bits/s bis 38,4 kBit/s
	314C-2 PtP	3964(R), ASCII, RK512				
CP	340 (*2)	3964(R), ASCII Drucker (*3)	integriert	1024 Byte	RS 422/485 (1200 m) RS 232C (15 m) 20mA TTY (aktiv: 100 m, passiv 1000 m)	2,4 kBit/s bis 19,2 kBit/s
	341 (*2)	3964R, ASCII, RK512 Drucker (*3)	integriert	4096 Byte	RS 422/485 (1200m) RS 232C (15m) 20mA TTY (aktiv und passiv 1000m)	300 Bit/s bis 115,2 kBit/s
		Modbus Seriell (RTU Format), Modbus Master und Modbus Slave	ladbar	abhängig vom Funk- tionscode (*1)		

Zusätzliche Möglichkeit: In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden. Siehe 23.11.

[Zurück zum Sprungverteiler SERIELL](#)

Erläuterungen zur Tabelle

(*1): Siehe dazu die entsprechenden Handbücher.

(*2): Auswahl der Physik der Schnittstelle über Baugruppenvariante

(*3): Wird im Dokument nicht weiter betrachtet, da damit keine CPU-CPU Kommunikation realisiert werden kann.

23.4 S7-400 an PtP

Tabelle 23-3

Controller an PtP: S7-400		Protokolle		max. Anzahl Daten	Physik (max. Abstand Kommunikationspartner)	Übertragungsrate
		Typ	auf CP ladbar / integriert			
CP	440	3964, ASCII	integriert	400 Byte	RS 422/485 (1200m)	300 Bit/s bis 115,2 kBit/s
	441-1 (*1)	3964, ASCII Drucker (*2)	integriert	ASCCII, 3964: 4096 Byte	RS 232C (10m) RS 422/485 (1200m)	300 Bit/s bis 115,2 kBit/s
	441-2 (*1)	3964, ASCII, RK512 Drucker (*2)	integriert	RK 512, Senden:4096 Byte RK 512, Holen: 450 Byte	20mA-TTY (1000m)	bei 20mA-TTY: bis 19,2 kBit/s
		Modbus Seriell (RTU Format), Mod- bus Master und Modbus Slave	ladbar	abhängig vom Funktionscode (*3)		

Zusätzliche Möglichkeit: In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden. Siehe 23.11.

[Zurück zum Sprungverteiler SERIELL](#)

Erläuterungen zur Tabelle

(*1): Auswahl der Physik der Schnittstelle über auf der Baugruppe steckbare IF-Module

(*2): Wird im Dokument nicht weiter betrachtet, da damit keine CPU-CPU Kommunikation realisiert werden kann.

(*3): Siehe dazu die entsprechenden Handbücher.

23.5 S7-1200 an PtP

Tabelle 23-4

Controller an PtP: S7-1200		Protokoll		maximale Anzahl Daten	Physik (max. Abstand Kommunikationspartner)	Übertragungsrate
		Typ	auf CP integriert / ladbar			
CP	CM 1241	USS Antriebsprotokoll (*2) Anwenderdefiniertes Protokoll Modbus Seriell (RTU Format), Modbus Master und Slave	integriert	1024 Byte	RS 485 (1000 m) RS 232 (10 m) (*1)	300 Bit/s bis 57,6 kBit/s

[Zurück zum Sprungverteiler SERIELL](#)

Erläuterungen zur Tabelle

(*1): Auswahl der Physik: Unterschiedliche Baugruppenvarianten

(*2): Wird im Dokument nicht weiter betrachtet, da damit keine CPU-CPU Kommunikation realisiert werden kann.

23.6 S7-mEC an PtP

Tabelle 23-5

Controller an PtP: S7-mEC		Protokoll		maximale Anzahl Daten	Physik (max. Abstand Kommu- nikationspartner)	Übertragungsrate
		Typ	auf CP integriert / ladbar			
CP	CP 340	(*1)	(*1)	(*1)	(*1)	(*1)

Zusätzliche Möglichkeit: In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden. Siehe 23.11.

[Zurück zum Sprungverteiler SERIELL](#)

Erläuterungen zur Tabelle

(*1): Siehe S7-300 / CP 340: Kapitel 23.3.

23.7 Box PC an PtP

In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden. Siehe 23.11.

[Zurück zum Sprungverteiler SERIELL](#)

23.8 Panel PC an PtP

In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden. Siehe 23.11.

[Zurück zum Sprungverteiler SERIELL](#)

23.9 WinAC MP an PtP

In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden. Siehe 23.11.

[Zurück zum Sprungverteiler SERIELL](#)

23.10 WinAC RTX an PtP

In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden. Siehe 23.11.

[Zurück zum Sprungverteiler SERIELL](#)

23.11 Dezentrale Station

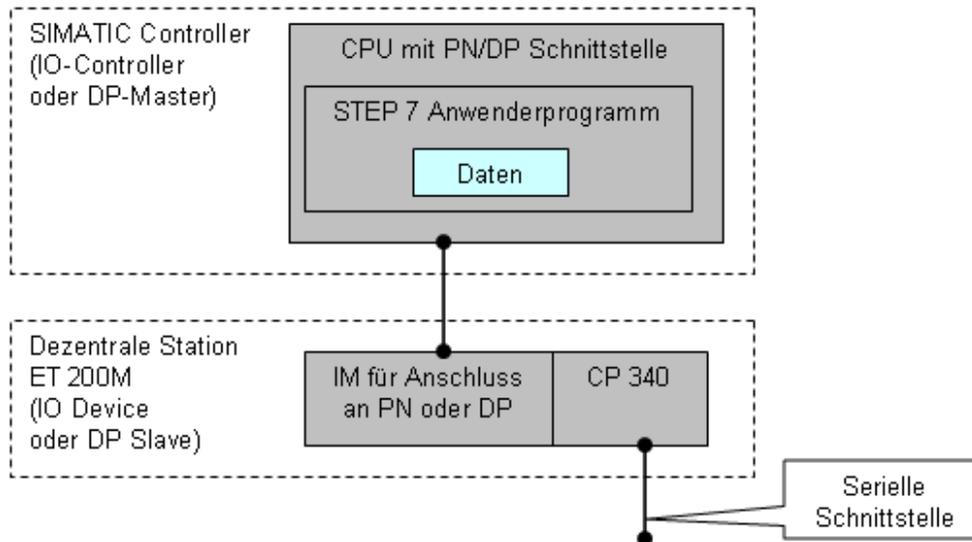
In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden.

Tabelle 23-6

Serielle Schnittstellenbaugruppe	dezentrale Station	Medium
1-SI Modul	ET 200S	PROFINET, PROFIBUS
CP340, CP341	ET 200M	PROFINET, PROFIBUS

Damit ergeben sich weitere Möglichkeiten für eine CPU-CPU Kommunikation über die Serielle Schnittstelle. Alle Controller mit einer PN bzw. DP Schnittstelle können diese Möglichkeit als IO Controller bzw. DP-Master nutzen. Die Abbildung zeigt ein Beispiel.

Abbildung 23-1



[Zurück zum Sprungverteiler SERIELL](#)

24 Informationen Teil 2

In den Tabellen sind Verweise auf Informationen zu den Themen aus Teil 2 zu finden.

Alle Verweise /x/ sind zentral im Kapitel 47 hinterlegt. Dort sind auch die entsprechenden Links ins Internet zu finden.

24.1 Kommunikationsarten

Tabelle 24-1

Verweis	Titel / Inhalt	Informationen zu
/7/	CPU 31xC und CPU 31xTechnische Daten Gerätehandbuch	Kommunikation über: MPI, PB, IE
/8/	Automatisierungssystem S7-400 CPU-Daten Gerätehandbuch	
/9/	S7-CPs für Industrial Ethernet Projektieren und in Betrieb nehmen	
/10/	S7-CPs für PROFIBUS Projektieren und in Betrieb nehmen	
/19/	Punkt-zu-Punkt-Kopplung CP 340 Aufbauen und Parametrieren	Kommunikation über: Serielle Schnittstelle
	Punkt-zu-Punkt-Kopplung CP 341 Aufbauen und Parametrieren	
/20/	Punkt-zu-Punkt-Kopplung CP 440 Aufbauen und Parametrieren	
	Punkt-zu-Punkt-Kopplung CP 441 Aufbauen und Parametrieren	
/21/	S7-300 CPU 31xC Technologische Funktionen (CPU 312C, CPU 313C, CPU 314C)	
/14/	Funktionsbausteine, Beispiele und Handbücher der seriellen Schnittstelle ET200S 1SI	
/28/	Handbuch ET 200S Serielle Schnittstellenbaugruppen	
/31/	Projektierungssoftware „PtP-Param“	
/23/	SIMATIC S7-300/S7-400 Ladbarer Treiber für Punkt-zu-Punkt-CPs: Modbus-Protokoll, RTU-Format, S7 ist Slave Betriebsanleitung	
/24/	SIMATIC S7-300/S7-400 Ladbarer Treiber für Punkt-zu-Punkt-CPs: Modbus-Protokoll, RTU-Format, S7 ist Master Betriebsanleitung	

24.2 Konkrete Anwendungsbeispiele

Tabelle 24-2

/x/	Titel / Inhalt
/200/	Applikationen zum Thema Kommunikation
/25/	SIMATC Net CD

24.3 Leistungsdaten

Tabelle 24-3

/x/	Titel / Inhalt
/18/	Leistungsdaten zur Kommunikation

Bereits in der Planungsphase einer Anlage ist es wichtig, die Übertragungszeit bei CPU-CPU Kommunikation in einem PROFIBUS, PROFINET IO bzw. Industrial Ethernet Netzwerks zu kennen.

Um sichere Aussagen zu praxisnahen Anlagen mit unterschiedlichen Topologien machen zu können, wurden bis zu 10.000 Konfigurationen aufgebaut und gemessen.

Im Beitrag /18/ sind verschiedene Tools mit leicht zu bedienenden Oberflächen zu finden. Damit können Anlagen hinsichtlich des Kommunikationsverhaltens optimal auslegt werden und unterschiedliche Anlagenkonfigurationen miteinander verglichen werden.

Beispiele:

- Übertragungszeit für typische Konfigurationen am Industrial Ethernet
- PN-Reaktionszeit für typische Konfigurationen am PROFINET IO
- DP-Reaktionszeit, HMI-Aktualisierungszeit und HMI-Bedienzeit für typische Konfigurationen am nicht-taktsynchronen PROFIBUS

TEIL 3: Kommunikationsarten

Teil 3 dient der Vertiefung

Detaillierte Informationen zu allen Kommunikationsarten

TEIL 3: Gliederung und Inhalt

Tabelle 24-4

Kapitel	Gliederung		Inhalt
26	SIMATIC S7 spezifische Kommunikation		Übersicht
27		Globaldaten-Kommunikation	Beschreibung der Kommunikationsarten
28		S7-Basiskommunikation	
29		S7-Kommunikation	
30	PROFINET/Industrial Ethernet		Übersicht
31		Offene-Kommunikation mit Send/Receive-Bausteinen	Beschreibung der Kommunikationsarten
32		Offene-Kommunikation mit T-Bausteinen	
---		PN-Kommunikation	
33		CBA	
34		PNIO	
35	PROFIBUS		
36		Offene-Kommunikation mit Send/Receive-Bausteinen	Beschreibung der Kommunikationsarten
37		FMS-Kommunikation	
38		DP-Kommunikation	
39	Serielle Schnittstelle		Übersicht
40		ASCII, 3964(R, RK 512	Beschreibung der Kommunikationsarten
41		Anwenderdefiniertes Protokoll	
42	Informationen		Gerätehandbücher, FAQs, Applikationen

Die Kopplung zu Controllern mit Modbus Schnittstelle wird in Teil 4 beschrieben:

Tabelle 24-5

Kapitel	Gliederung	Inhalt
44	Modbus/TCP	Kommunikation über Netz PN/IE
45	Modbus Seriell (RTU Format)	Kommunikation über Serielle Schnittstelle

25 Vorbemerkungen

Für jede Kommunikationsart gibt es die folgenden Kapitel:

- Merkmale
- Eigenschaften (Tabelle Kommunikationsarten-Detail)
- Anwendung
- Überblick Anwenderschnittstellen
- Anwenderschnittstellen (Beschreibung und Übersicht der Parameter)

Im Folgenden wird beschrieben, welche Inhalte diese Kapitel haben.

25.1 Merkmale

In diesem Kapitel wird die Kommunikationsart stichpunktartig charakterisiert.

Hier werden keine Details genannt, sondern nur Schlagworte (Highlights).

25.2 Eigenschaften (Tabelle Kommunikationsarten-Detail)

Das Kapitel enthält eine Tabelle mit der Bezeichnung Kommunikationsarten-Detail. In dieser Tabelle werden die wichtigsten Eigenschaften einer Kommunikationsart beschrieben werden.

25.2.1 Zweck der Tabelle

Die Tabelle beantwortet folgende Fragen:

- Welche Eigenschaften hat die Kommunikationsart?
- Welche Eigenschaften haben die Anwenderschnittstellen (Kommunikationsbausteine) der Kommunikationsart?

Die Tabelle ist Grundlage für eine weitere Tabelle mit der Bezeichnung Kommunikationsarten-Kompakt (17.4). In der Tabelle Kommunikationsarten-Kompakt werden alle Kommunikationsarten eines Mediums (PN/IE, ...) gegenübergestellt..

25.2.2 Aufbau der Tabelle

Der Aufbau wird an Hand eines Beispielles erläutert:

- Medium PN/IE
- Kommunikationsart Offene-Kommunikation mit T-Bausteinen.

Das Bild zeigt einen Ausschnitt der Tabelle (Tabelle 32-3).

Abbildung 25-1

Kommunikationsart:		Offene-Kommunikation mit T-Bausteinen		
Protokoll:		ISO on TCP	TCP	
Allgemeines				
Medien		PN/IE	PN/IE	
Schnittstellen		CPU, CP	CPU, CP	
Anbindung	SIMATIC S5	ja	ja	
	Fremd (offene Standards)	ja	ja	
Anwenderschnittstelle				
Kommunikationsbausteine		TSEND / TRCV	TSEND_C / TRCV_C	TSEND / TRCV
maximale Anzahl Daten (*1)		<= 32 KByte	= 8192 Byte	<= 64 KByte
Anzahl Variablen bei Aufruf Kommunikationsbaustein		1	1	1
dynamische Adressierung Daten		ja	ja	ja
remote Quittierung		Transport	Transport	Transport
Modell		Client / Client	Client / Client	Client / Client

Die Tabelle besteht aus zwei Bereichen, die im Folgenden erläutert werden.

Bereich 1: Kommunikationsart / Protokoll

Bezeichnung der Kommunikationsart und der möglichen Protokolle.

Bereich 2: Kriterien

Die Kriterien sind unter folgenden Überschriften zusammengefasst:

- Allgemeines
- Anwenderschnittstelle
- Protokoll
- Anwendertätigkeit

Die Kriterien werden im folgenden Kapitel (Kapitel 25.2.3) beschrieben.

25.2.3 Erläuterung der Kriterien

Eine Untermenge dieser Kriterien erscheint in den Tabellen Kommunikationsarten-Kompakt. Diese sind mit einem Unterstrich gekennzeichnet (Beispiel: Schnittstellen).

Tabelle 25-1

Kriterium		Bedeutung	Wertebereich (*1)
Allgemeines			
Medien		Hier werden die Medien eingetragen, über welche die beiden Kommunikationspartner Daten austauschen.	PN/IE, PB, MPI, Rückwandbus, Serielle Schnittstelle
<u>Schnittstellen</u>		Hier wird eingetragen, an welcher Baugruppe die Medien angeschlossen sind. CPU: integrierte Schnittstelle CP: externe Schnittstelle	CPU, CP
Anbindung	SIMATIC S5	Kommunikation mit SIMATIC S5 möglich?	ja, nein
	Fremd	Ist eine Kommunikation mit Controllern fremder Hersteller über <u>offene Standards</u> möglich?	ja, nein
Anwenderschnittstelle			
<u>Kommunikationsbausteine</u>		Hier werden alle für die Datenübertragung zur Verfügung stehenden Funktionsbausteine (FB, SFB, FC, SFC) aufgeführt (Kommunikationsbausteine). Funktionsbausteine, die eventuell für den Verbindungsaufbau und Verbindungsabbau benötigt werden, werden hier nicht erwähnt (Verbindungsbausteine). <u>Bedeutung der Schreibweise:</u> a / b: Zur Datenübertragung werden beide Bausteine benötigt (Beispiel: BSEND / BRCV) a, b: Jeder einzelne Baustein kann eine Datenübertragung ausführen (Beispiel: PUT, GET)	BSEND / BRCV AG_SEND / AG_RECV PUT, GET usw.
<u>maximale Anzahl Daten</u>		Hier wird eingetragen, wie viele Daten ein Kommunikationsbaustein <u>maximal</u> auf einmal übertragen kann: Von „Anstoß Auftrag“, bis „Auftrag fertig“ <u>Bedeutung der Schreibweise:</u> = x Byte: immer x Byte (unter allen Umständen) <= x Byte: höchstens x Byte, es können aber auch weniger sein (abhängig von CPU, CP, ...) >= x Byte: mindestens x Byte, es können aber auch mehr sein (abhängig von CPU, CP, ...)	= x Byte <= x Byte >= x Byte

25.2 Eigenschaften (Tabelle Kommunikationsarten-Detail)

Kriterium	Bedeutung	Wertebereich (*1)
Anzahl Variablen bei Aufruf Kommunikationsbaustein	Hier wird angegeben, wie viele unterschiedliche Datenbereiche (Variablen) am Kommunikationsbaustein vom Anwender parametrisiert werden können. (Übergabebereiche).	1 1 bis 4
Dynamische Adressierung Daten	Ist eine Änderung der Adressierung von Datenbereichen zur Laufzeit (im RUN der CPU) möglich?	ja, nein
<u>remote Quittierung</u>	<p>Hier wird angegeben, ob eine Datenübertragung vom remote Kommunikationspartner (CPU, CP) quittiert wird, und was die Quittierung bedeutet.</p> <p>Fallunterscheidung:</p> <p><u>keine Quittung von remote („nein“)</u></p> <ul style="list-style-type: none"> Die Daten wurden gesendet, haben lokalen Partner verlassen. Der Sender erfährt nicht, ob die Daten im remote Anwenderdatenbereich (in der remote Applikation auf der CPU) angekommen sind. <p><u>Transport Quittung von remote („Transport“)</u></p> <ul style="list-style-type: none"> Die Daten wurden gesendet und vom remote Partner (CPU, CP) empfangen. Der Sender erfährt nicht, ob die Daten im remote Anwenderdatenbereich (in der remote Applikation auf der CPU) angekommen sind. <p><u>Applikation Quittung von remote („Applikation“)</u></p> <ul style="list-style-type: none"> Die Daten wurden gesendet, und die Daten sind im remote Anwenderdatenbereich (in der remote Applikation auf der CPU) angekommen. 	nein, Transport, Applikation
Modell	Hier wird angegeben, welches Kommunikationsmodell der Datenübertragung zu Grunde liegt. Beschreibung der Modelle: Siehe Kapitel 50.2.	Client / Client Client / Server, S7 nur Server, Master / Slave, Consumer / Provider
Protokoll		
dynamische Datenlänge	Kann zur Laufzeit (RUN der CPU) am Kommunikationsbaustein die Datenlänge geändert werden? Dies ist gleich bedeutend mit: Kann das Protokoll Anfang und Ende der übermittelten Daten erkennen?	Ja, nein
Multicast / Broadcast	Hier wird eingetragen, ob Multicast oder Broadcast möglich ist. Multicast: Gleichzeitiges Senden an mehrere Kommunikationspartner Broadcast: Gleichzeitiges Senden an alle Kommunikationspartner	Multicast, Broadcast

25.2 Eigenschaften (Tabelle Kommunikationsarten-Detail)

Kriterium		Bedeutung	Wertebereich (*1)
Verbindungen	zum remote Partner?	Details zu Verbindungen: siehe Kapitel 5. <u>Fallunterscheidung:</u> nein: Zur Kommunikation wird keine Verbindung zum remote Partner aufgebaut ja: Zur Kommunikation wird eine Verbindung zum remote Partner aufgebaut	nein, Ja
	dynamisch / statisch?	<u>Fallunterscheidung:</u> dynamisch: Die Verbindung wird nach der Datenübertragung abgebaut. statisch: Die Verbindung bleibt nach der Datenübertragung aufgebaut. dynamisch + statisch: Beide obigen Fälle sind möglich.	dynamisch, statisch, dynamisch + statisch
Routingfähig		Können CPU-CPU Kommunikation über Netzgrenzen (über Router) hinweg erfolgen? Das Kriterium ist nur bei PN/IE relevant.	ja, nein
Anwendertätigkeit			
Kommunikationsbeziehung	festlegen mit	Wie (mit welchem Tool) wird eine Kommunikationsbeziehung festgelegt? Im Sinne von: Welche CPU kommuniziert mit welcher CPU?	GD-Editor, Hardware Konfig, NetPro, iMAP
Verbindung	Projektieren mit NetPro?	Wird die Verbindung mit NetPro projektiert?	ja, nein, --- (Verbindung nicht erforderlich)
	Programmieren in STEP 7?	Sind zum Einrichten einer Verbindung Verbindungsbausteine erforderlich?	ja, nein, --- (Verbindung nicht erforderlich)
Datenübertragung	Kommunikationsbaustein in STEP 7?	Sind zur Datenübertragung Kommunikationsbausteine erforderlich?	ja, nein

Erläuterungen zur Tabelle

(*1: Allgemein gilt für den Wertebereich:

„---“ bedeutet: nicht relevant / entfällt / hier ohne Bedeutung / nicht zutreffend

25.3 Anwendung

In diesem Kapitel werden die wichtigsten Schritte gezeigt, die der Anwender ausführen muss, um eine CPU-CPU Kommunikation zu realisieren. Es wird die prinzipielle Vorgehensweise gezeigt.

Detailliertere Informationen sind in folgenden Quellen zu finden:

- Online-Hilfe von STEP 7
- Handbücher zu STEP 7
- Handbücher zu SIMATIC Controllern
- Handbücher zu SIMATIC CPs

25.4 Überblick Anwenderschnittstellen

Das Kapitel gibt einen Überblick zu allen Kommunikationsbausteinen der betreffenden Kommunikationsart. Es werden folgende Fragen beantwortet:

- Für welche Familie und Schnittstelle (CPU, CP) sind die Kommunikationsbausteine geeignet?
- Wo in STEP 7 (Bibliothek) sind diese Kommunikationsbausteine zu finden?

25.5 Anwenderschnittstellen

Die Anwenderschnittstellen der jeweiligen Kommunikationsarten werden stichpunktartig beschrieben:

- Funktionalität der Kommunikationsbausteine
- Parameter der Kommunikationsbausteine

Detailliertere Informationen sind zu finden in:

- Online-Hilfe von STEP 7
- Handbuch System- und Standardfunktionen für S7-300/400 [/6/](#)

Im Dokument werden zur Beschreibung einheitliche Begriffe verwendet. Es werden die beiden Fälle unterschieden:

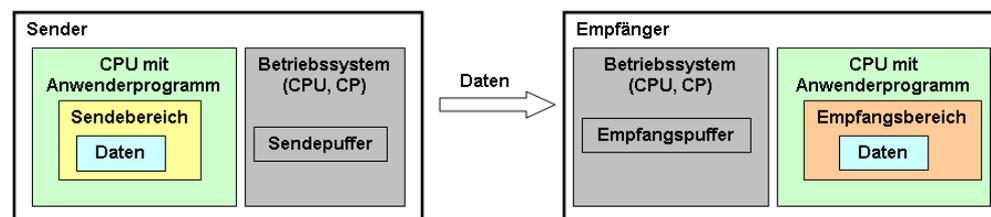
- Dem Partner werden keine Adressinformationen der Daten mitgeteilt
- Dem Partner werden Adressinformationen der Daten mitgeteilt

Die folgenden Bilder zeigen die in diesem Zusammenhang verwendeten Begriffe.

Keine Adressinformationen der Daten

Daten senden und empfangen

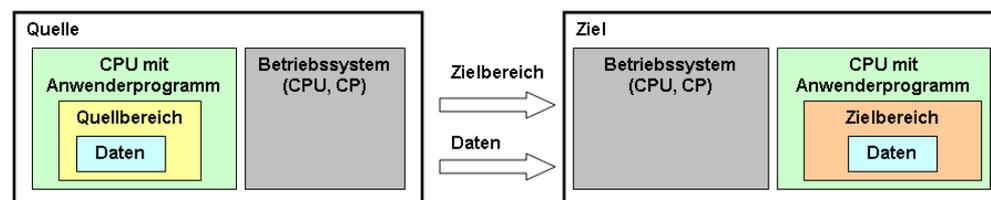
Abbildung 25-2



Adressinformationen der Daten

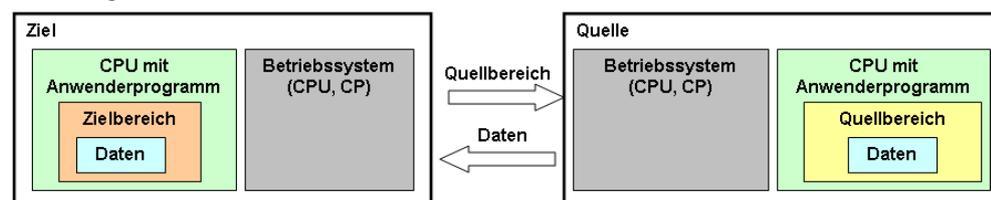
Daten schreiben

Abbildung 25-3



Daten lesen, holen

Abbildung 25-4



26 SIMATIC S7 spezifische Kommunikation

26.1 Merkmale

Die SIMATIC S7 spezifische Kommunikation zeichnet sich durch folgende Merkmale aus:

- Optimierte Kommunikation zwischen SIMATIC S7 Controllern
- Herstellerspezifische Kommunikation (kein offener Standard)

26.2 Übersicht

Folgende Kommunikationsarten und Medien stehen zur Verfügung:

Tabelle 26-1

Kommunikationsart	Medien				
	Netz			Rückwandbus (*1)	Serielle Schnittstelle
	PN/IE	PB	MPI		
Globaldaten-Kommunikation	---	---	x	x	---
S7-Basiskommunikation	---	x (DP)	x	---	---
S7-Kommunikation	x	x	x	x	---

(*1): nur möglich bei SIMATIC S7-400 (Multicomputing, Kapitel 4.5)

In den folgenden Kapiteln werden die Kommunikationsarten beschrieben:

Tabelle 26-2

	Kapitel
Globaldaten-Kommunikation	27
S7-Basiskommunikation	28
S7-Kommunikation	29

27 Globaldaten-Kommunikation

27.1 Merkmale

Globaldaten

Die in dieser Form der Kommunikation ausgetauschten Daten werden als Globaldaten (GD) bezeichnet.

Globaldaten können sein:

- Eingänge, Ausgänge (Prozessabbild)
- Merker
- Bereiche aus Datenbausteinen
- Zeiten, Zähler

Der Datenaustausch erfolgt zyklisch während der Aktualisierung des Prozessabbildes der Eingänge und Ausgänge. Bei S7-400 ist ereignisgesteuerter Datenaustausch möglich.

Globaldaten werden in GD-Paketen übertragen. Ein GD-Paket ist ein Telegramm, das von einer SIMATIC CPU an eine oder mehrere andere SIMATIC CPUs gesendet wird.

Merkmale

Die GD-Kommunikation zeichnet sich durch folgende Merkmale aus:

- Sehr einfache Anwendung
- Anzahl Daten: ≤ 54 Byte

27.2 Eigenschaften

Erläuterungen zum Aufbau und Inhalt der Tabelle sind in Kapitel 25.2 zu finden.

Tabelle Kommunikationsarten-Detail:

Tabelle 27-1

Kommunikationsart:		GD-Kommunikation
Protokoll:		S7-Protokoll
Allgemeines		
Medien		MPI, Rückwandbus (*1)
Schnittstellen		CPU
Anbindung	SIMATIC S5	nein
	Fremd (offene Standards)	nein
Anwenderschnittstelle		
Kommunikationsbausteine		zyklische Übertragung: keine Kommunikationsbausteine erforderlich azyklische Übertragung (nur S7-400): GD_SND, GD_RCV
maximale Anzahl Daten		S7-300: Anzahl Daten pro GD-Paket: = 22 Byte
		S7-400: Anzahl Daten pro GD-Paket: = 54 Byte
Anzahl Variablen bei Aufruf Kommunikationsbaustein		---
dynamische Adressierung Daten		nein
remote Quittierung		nein
Modell		---

Kommunikationsart:		GD-Kommunikation
Protokoll:		S7-Protokoll
Protokoll		
dynamische Datenlänge		nein
Multicast / Broadcast		Multicast
Verbindungen	zum remote Partner?	nein
	dynamisch / statisch	---
Routingfähig		nein
Anwendertätigkeit		
Kommunikationsbeziehung.	Festlegen mit	GD-Editor
Verbindungen	proj. mit NetPro?	---
	prog. in STEP 7?	---
Datenübertragung	Kommunikationsbaustein in STEP 7?	zyklische Übertragung: nein azyklische Übertragung (nur S7-400): ja

[Zurück zum Sprungverteiler MPI](#)

[Zurück zum Sprungverteiler Rückwandbus](#)

Erläuterungen zur Tabelle

(*1): nur möglich bei SIMATIC S7-400 (Multicomputing, Kapitel 4.5)

27.3 Anwendung

Es folgt eine Übersicht der wichtigsten Tätigkeiten, um eine CPU-CPU Kommunikation zu realisieren.

Zyklische Übertragung

Tabelle 27-2

Tätigkeit	Engineering Tool
Vernetzung und Adressvergabe	STEP 7, HW Konfig
Projektierung der Globaldaten-Kreise	STEP 7, GD-Editor

Azyklische Übertragung

Wie oben, und zusätzlich:

Tabelle 27-3

Tätigkeit	Engineering Tool
Aufruf Kommunikationsbaustein im STEP 7 Anwenderprogramm	STEP 7, Sprachen Editor

27.4 Überblick Anwenderschnittstellen

Zyklische Übertragung

Es sind keine Kommunikationsbausteine erforderlich.

Azyklische Übertragung

Überblick Kommunikationsbausteine:

Tabelle 27-4

Kommunikationsbausteine	S7-400
	CPU
GD_SND	SFC 60
GD_RCV	SFC 61

Kommunikationsbausteine in STEP 7:

Tabelle 27-5

Schnittstelle	STEP 7 Bibliothek
CPU	Standard Library / System Function Blocks

27.5 Anwenderschnittstelle GD_SND, GD_RCV

27.5.1 Beschreibung

Mit den Kommunikationsbausteinen wird eine azyklische GD-Kommunikation zwischen CPUs der S7-400 ermöglicht.

GD_SND

Programmiertes Senden eines GD-Paketes

GD_RCV

Programmiertes Empfangen eines GD-Paketes

27.5.2 Parameter für GD_SND

Tabelle 27-6

INPUT	Typ	Bemerkung
CIRCLE_ID	BYTE	Nummer des GD-Kreises, in dem sich das zu sendende GD-Paket befindet.
BLOCK_ID	BYTE	Nummer des zu sendenden GD-Paketes im ausgewählten GD-Kreis.
OUTPUT	Typ	Bemerkung
RET_VAL	INT	Fehlerinformation

27.5.3 Parameter für GD_RCV

Tabelle 27-7

INPUT	Typ	Bemerkung
CIRCLE_ID	BYTE	Nummer des GD-Kreises, in den das angekommene GD-Paket eingetragen werden soll.
BLOCK_ID	BYTE	Nummer des GD-Paketes, in das die angekommenen Daten eingetragen werden sollen.
OUTPUT	Typ	Bemerkung
RET_VAL	INT	Fehlerinformation

28 S7-Basiskommunikation

28.1 Merkmale

Kommunikationspartner

Mögliche Kommunikationspartner für eine CPU-CPU Kommunikation mit Hilfe der S7-Basiskommunikation:

- CPU außerhalb der eigenen Station (*1)
- CPU innerhalb der eigenen Station (*1):
 - CPU im Zentralgerät oder Erweiterungsgerät
 - CPU dezentral

(*1): Definition „eigene Station“

Unter „eigene Station“ ist hier folgendes zu verstehen:

- zentrale Station mit CPU, CP, zentraler Peripherie, und
- dezentrale Station.

In einer dezentralen Station kann eine CPU gesteckt sein. In diesem Fall wird diese dezentrale Station auch als „intelligenter Slave (I-Slave)“ bezeichnet.

CPU außerhalb der eigenen Station

Die zugehörigen Kommunikationsbausteine werden mit „X-Bausteine“ bezeichnet.

CPU innerhalb der eigenen Station

Die zugehörigen Kommunikationsbausteine werden mit „I-Bausteine“ bezeichnet.

Zwischen DP-Master und DP-Slave werden Daten ausgetauscht:

- DP-Master liest mit I-Baustein Daten aus DP-Slave
- DP-Master schreibt mit I-Baustein Daten in DP-Slave

Im DP-Slave sind keine Kommunikationsbausteine erforderlich.

Merkmale

Die S7-Basiskommunikation zeichnet sich durch folgende Merkmale aus:

- Kommunikation nur über die integrierte Schnittstelle der CPU
- Kommunikation über Verbindungen. Die Verbindungen werden vom Kommunikationsbaustein eingerichtet (Projektierung in STEP 7 ist nicht erforderlich).
- Anzahl Daten pro K-Auftrag: < 94 Byte
- Das Anwenderprogramm im Sender erhält eine Information, wenn die Daten im Anwenderdatenbereich des Empfängers angekommen sind (Quittierung von der remote Applikation).

28.2 Eigenschaften

Erläuterungen zum Aufbau und Inhalt der Tabelle sind in Kapitel 25.2 zu finden.

Tabelle Kommunikationsarten-Detail:

Tabelle 28-1

Kommunikationsart:		S7-Basiskommunikation		
Protokoll:		S7-Protokoll		
Allgemeines				
Medien		PB (DP)	MPI	MPI
Schnittstellen		CPU	CPU	CPU
Anbindung	SIMATIC S5	nein	nein	nein
	Fremd (offene Standards)	nein	nein	nein
Anwenderschnittstelle				
Kommunikationsbausteine		I_PUT, I_GET	X_PUT, X_GET	X_SEND / X_RCV
maximale Anzahl Daten		I_PUT: = 84 Bytes I_GET: = 94 Bytes	= 76 Bytes	= 76 Bytes
Anzahl Variablen bei Aufruf Kommunikationsbaustein		1	1	1
dynamische Adressierung Daten		ja	ja	ja
remote Quittierung		Applikation	Applikation	Applikation
Modell		Client / Server	Client / Server	Client / Client

Kommunikationsart:		S7-Basiskommunikation
Protokoll:		S7-Protokoll
Protokoll		
dynamische Datenlänge		ja
Multicast / Broadcast		nein
Verbindungen	zum remote Partner?	ja
	dynamisch / statisch	dynamisch + statisch
Routingfähig		nein
Anwendertätigkeit		
Kommunikationsbeziehung	Festlegen mit	Hardware Konfig
Verbindungen	proj. mit NetPro?	nein
	prog. in STEP 7?	nein
Datenübertragung	Kommunikationsbaustein erforderlich?	Client: ja Server: nein

[Zurück zum Sprungverteiler MPI](#)

[Zurück zum Sprungverteiler Rückwandbus](#)

[Zurück zum Sprungverteiler PB](#)

28.3 Anwendung

Es folgt eine Übersicht der wichtigsten Tätigkeiten, um eine CPU-CPU Kommunikation zu realisieren. Fallunterscheidung:

Tabelle 28-2

Fall	Kommunikationsbausteine	Kommunikation
X-Bausteine	X_PUT, X_GET X_SEND/X_RCV	Kommunikation außerhalb der eigenen Station
I-Bausteine	I_PUT, I_GET	Kommunikation innerhalb der eigenen Station

28.3.1 X-Bausteine

Client / Client Kommunikation

Auf beiden Seiten der Kommunikationsbeziehung:

Tabelle 28-3

Tätigkeit	Engineering Tool
Vernetzung und Adressvergabe	STEP 7, HW Konfig
Programmierung Datenaustausch: Aufruf Kommunikationsbausteine im STEP 7 Anwenderprogramm (X_SEND, X_RCV)	STEP 7, Sprachen Editor

Client / Server Kommunikation

Auf der Client Seite der Kommunikationsbeziehung:

Tabelle 28-4

Tätigkeit	Engineering Tool
Vernetzung und Adressvergabe	STEP 7, HW Konfig
Programmierung Datenaustausch: Aufruf Kommunikationsbausteine im STEP 7 Anwenderprogramm (X_PUT, X_GET)	STEP 7, Sprachen Editor

Auf der Server Seite der Kommunikationsbeziehung:

Wie Tabelle oben , aber ohne „Programmierung Datenaustausch“.

28.3.2 I-Bausteine

Client / Server Kommunikation

Auf der Client Seite der Kommunikationsbeziehung:

Tabelle 28-5

Tätigkeit	Engineering Tool
Vernetzung und Adressvergabe	STEP 7, HW Konfig
I-Slave projektieren (DP-Slave mit CPU) DP-Master projektieren, DP-Slave projektieren	
Programmierung Datenaustausch: Aufruf Kommunikationsbausteine im STEP 7 Anwenderprogramm (I_PUT, I_GET)	STEP 7, Sprachen Editor

Auf der Server Seite der Kommunikationsbeziehung:

Wie Tabelle oben , aber ohne „Programmierung Datenaustausch“.

28.4 Überblick Anwenderschnittstellen

Überblick Kommunikationsbausteine:

Tabelle 28-6

Kommunikationsbaustein		S7-300	S7-400
		CPU	CPU
X-Bausteine	X_SEND / X_RCV	SFC 65 / SFC 66	SFC 65 / SFC 66
	X_PUT	SFC 68	SFC 68
	X_GET	SFC 67	SFC 67
I-Bausteine	I_PUT	SFC 73	SFC 73
	I_GET	SFC 72	SFC 72

Kommunikationsbausteine in STEP 7:

Tabelle 28-7

Schnittstelle	STEP 7 Bibliothek
CPU	Standard Library / System Function Blocks

28.5 Anwenderschnittstelle X_SEND / X_RCV

28.5.1 Beschreibung

Mit den Kommunikationsbausteinen kommuniziert eine CPU mit einer anderen CPU außerhalb der eigenen Station (Client / Client Kommunikation).

X_SEND

Daten an CPU außerhalb der eigenen Station senden.

X_RCV

Daten von CPU außerhalb der eigenen Station empfangen.

28.5.2 Parameter für X_SEND

Tabelle 28-8

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
CONT	BOOL	Verbindung nach Beendigung des Auftrags: halten / abbauen
DEST_ID	WORD	MPI-Adresse des Kommunikationspartners.
REQ_ID	DWORD	Auftragskennung für Empfänger
SD	ANY	Sendebereich (*2)
OUTPUT	Typ	Bemerkung
RET_VAL	INT	Fehlerinformation
BUSY	BOOL	Auftrag läuft / Auftrag fertig (*1)

(*1): remote Quittierung: Applikation

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

28.5.3 Parameter für X_RCV

Tabelle 28-9

INPUT	Typ	Bemerkung
EN_DT	BOOL	Anstoß Empfangsauftrag
OUTPUT	Typ	Bemerkung
RET_VAL	INT	Fehlerinformation
REQ_ID	DWORD	Auftragskennung vom Sender
NDA	BOOL	Daten im Eingangspuffer: ja / nein
INOUT	Typ	Bemerkung
RD	ANY	Empfangsbereich (*1)

(*1): SIMATIC S7 Speicherbereiche: E, A, M, D

28.6 Anwenderschnittstelle X_PUT, X_GET

28.6.1 Beschreibung

Mit den Kommunikationsbausteinen kommuniziert eine CPU mit einer anderen CPU außerhalb der eigenen Station (Client / Server Kommunikation).

X_PUT

Daten in CPU außerhalb der eigenen Station schreiben.

X_GET

Daten aus CPU außerhalb der eigenen Station lesen.

28.6.2 Parameter für X_PUT

Tabelle 28-10

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Schreibauftrag
CONT	BOOL	Verbindung nach Beendigung des Auftrags: halten / abbauen
DEST_ID	WORD	MPI-Adresse des Kommunikationspartners
VAR_ADDR	DWORD	Zielbereich (*2)
SD	ANY	Quellbereich (*2)
OUTPUT	Typ	Bemerkung
RET_VAL	RET_VAL	Fehlerinformation
BUSY	BUSY	Auftrag läuft / Auftrag fertig (*1)

(*1): remote Quittierung: Applikation

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

28.6.3 Parameter für X_GET

Tabelle 28-11

INPUT	Datentyp	Bemerkung
REQ	BOOL	Anstoß Leseauftrag
CONT	BOOL	Verbindung nach Beendigung des Auftrags: halten / abbauen
DEST_ID	WORD	MPI-Adresse des Kommunikationspartners
VAR_ADDR	DWORD	Quellbereich (*2)
OUTPUT	Typ	Bemerkung
RET_VAL	INT	Fehlerinformation
BUSY	BOOL	Auftrag läuft / Auftrag fertig (*1)
INOUT	Typ	Bemerkung
RD	ANY	Zielbereich (E, A, M, D)

(*1): remote Quittierung: Applikation

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

28.7 Anwenderschnittstelle I_PUT, I_GET

28.7.1 Beschreibung

Mit den Kommunikationsbausteinen kommuniziert eine CPU mit einer anderen CPU innerhalb der eigenen Station (Client / Server Kommunikation).

I_PUT

Daten in CPU innerhalb der eigenen Station schreiben.

I_GET

Daten aus CPU innerhalb der eigenen Station lesen.

28.7.2 Parameter für I_PUT

Tabelle 28-12

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Schreibauftrag
CONT	BOOL	Verbindung nach Beendigung des Auftrags: halten / abbauen
IOID	BYTE	Adressbereichs der Partnerbaugruppe (PE, PA)
LADDR	WORD	Logisch Adresse der Partnerbaugruppe
VAR_ADDR	ANY	Zielbereich (*2)
SD	ANY	Quellbereich (*2)
OUTPUT	Typ	Bemerkung
RET_VAL	INT	Fehlerinformation
BUSY	BOOL	Auftrag läuft / Auftrag fertig (*1)

(*1): remote Quittierung: Applikation

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

28.7.3 Parameter für I_GET

Tabelle 28-13

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Leseauftrag
CONT	BOOL	Verbindung nach Beendigung des Auftrags: halten / abbauen
IOID	BYTE	Adressbereichs der Partnerbaugruppe (PE, PA)
LADDR	WORD	Logisch Adresse der Partnerbaugruppe
VAR_ADDR	ANY	Quellbereich (*2)
OUTPUT	Typ	Bemerkung
RET_VAL	INT	Fehlerinformation
BUSY	BOOL	Auftrag läuft / Auftrag fertig (*1)
RD	ANY	Zielbereich (*2)

(*1): remote Quittierung: Applikation

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

29 S7-Kommunikation

29.1 Merkmale

Die S7-Kommunikation zeichnet sich durch folgende Merkmale aus:

- Netzunabhängige Anwenderschnittstelle:
Identisches Handling für PN/IE, PB und MPI
- Kommunikation über integrierte Schnittstelle der CPU oder über CP
- Kommunikation über Verbindungen.
Verbindungen werden in STEP 7 projiziert (NetPro).
- Anzahl Daten pro K-Auftrag: ≤ 64 KByte
- Das Anwenderprogramm im Sender erhält eine Information, wenn die Daten im Anwenderdatenbereich des Empfängers angekommen sind (Quittierung von der remote Applikation).

29.2 Eigenschaften

Erläuterungen zum Aufbau und Inhalt der Tabelle sind in Kapitel 25.2 zu finden.

Tabelle Kommunikationsarten-Detail:

Tabelle 29-1

Kommunikationsart:		S7-Kommunikation			
Protokoll:		S7-Protokoll (*1)			
Allgemeines					
Medien		MPI, PB, PN/IE, Rückwandbus (*3)			
Schnittstellen		CPU, CP			
Anbindung	SIMATIC S5	nein			
	Fremd (offene Standards)	nein			
Anwenderschnittstelle					
Kommunikationsbausteine		BSEND / BRCV	USEND / URCV USEND_E / URCV_E (*4)		PUT, GET PUT_E, GET_E (*4)
maximale Anzahl Daten (*2)		<= 64 KByte	>= 160 Byte		>= 160 Byte
Anzahl Variablen bei Aufruf Kommunikationsbaustein		1	S7-300:	USEND/URCV: 1	S7-300: PUT, GET: 1
				USEND_E/URCV_E: 1 bis 4	PUT_E, GET_E: 1 bis 4
			S7-400: 1 bis 4		S7-400: 1 bis 4
dynamische Adressierung Daten		S7-300: ja	S7-300: ja		S7-300: ja
		S7-400: nein	S7-400: nein		S7-400: nein
remote Quittierung		Applikation	Transport		Applikation
Modell		Client / Client	Client / Client		Client / Server

Kommunikationsart:		S7-Kommunikation
Protokoll:		S7-Protokoll (*1)
Protokoll		
dynamische Datenlänge		ja
Multicast / Broadcast		nein
Verbindungen	zum remote Partner?	ja
	dynamisch / statisch	statisch
Routingfähig		nur bei Netz PN/IE und Protokoll ISO on TCP
Anwendertätigkeit		
Kommunikationsbeziehung	Festlegen mit	NetPro
Verbindungen	proj. mit NetPro?	ja (Server für PUT/GET: nein)
	prog. in STEP 7?	nein
Datenübertragung	Kommunikationsbaustein erforderlich?	Client: ja Server: nein

[Zurück zum Sprungverteiler PN/IE](#)

[Zurück zum Sprungverteiler PB](#)

[Zurück zum Sprungverteiler MPI](#)

[Zurück zum Sprungverteiler Rückwandbus](#)

Erläuterungen zur Tabelle:

(*1): Das S7-Protokoll setzt auf folgenden Protokollen auf:

Tabelle 29-2

Netz	Kommunikation über CPU	Kommunikation über CP
PN/IE	ISO on TCP	ISO, ISO on TCP
MPI, PB	FDL	FDL

(*2): Die maximale Anzahl Daten pro K-Auftrag ist abhängig von:

- Netz (MPI, PB, PN/IE)
- Schnittstelle (Kommunikation über CPU oder CP)
- Kommunikationspartner (S7-300, S7-400, ...)
- Kommunikationsbaustein (BSEND, PUT, ...)

Bei einigen Kommunikationsbausteinen wird angegeben, wie viele Daten **mindestens** ($\geq x$) übertragen werden können. Wenn diese Mindestanzahl ($\geq x$) nicht ausreicht, kann die exakte maximale Anzahl mit Hilfe von [/6/](#) (Kapitel S7-Kommunikation) ermittelt werden.

Tabelle 29-3

Kommunikationsbaustein	S7-300		S7-400
	CPU	CP	CPU, CP
BSEND / BRCV	= 64 KByte	≤ 32 KByte	= 64 KByte
USEND / URCV	≥ 160 Byte	≤ 160 Byte	≥ 440 Byte
PUT, GET	≥ 160 Byte	≤ 160 Byte	≥ 400 Byte
USEND_E	≥ 160 Byte	nicht vorhanden	nicht vorhanden
PUT_E, GET_E	≥ 160 Byte	nicht vorhanden	nicht vorhanden

(*3): nur möglich bei SIMATIC S7-400 (Multicomputing, Kapitel 4.5)

(*4): Die Kommunikationsbausteine unterstützen derzeit nur Verbindungen über die integrierte PN-Schnittstelle der CPU.

29.3 Anwendung

Es folgt eine Übersicht der wichtigsten Tätigkeiten, um eine CPU-CPU Kommunikation zu realisieren.

Client / Client Kommunikation

Auf beiden Seiten der Kommunikationsbeziehung:

Tabelle 29-4

Tätigkeit	Engineering Tool
Vernetzung und Adressvergabe	STEP 7, HW Konfig
Projektierung der Verbindungen: <ul style="list-style-type: none"> • Auswahl der Baugruppen, die kommunizieren sollen • Auswahl Verbindungstyp • Parametrierung Verbindung 	STEP 7, NetPro
Programmierung Datenaustausch: Aufruf Kommunikationsbausteine im STEP 7 Anwenderprogramm (BSEND/BRCV, USEND/URCV)	STEP 7, Sprachen Editor

Client / Server Kommunikation

Auf Client Seite der Kommunikationsbeziehung:

Tabelle 29-5

Tätigkeit	Engineering Tool
Vernetzung und Adressvergabe	STEP 7, HW Konfig
Projektierung der Verbindungen: <ul style="list-style-type: none"> • Auswahl der Baugruppen, die kommunizieren sollen • Auswahl Verbindungstyp • Parametrierung Verbindung 	STEP 7, NetPro
Programmierung Datenaustausch: Aufruf Kommunikationsbausteine im STEP 7 Anwenderprogramm (PUT, GET)	STEP 7, Sprachen Editor

Auf der Server Seite der Kommunikationsbeziehung:

Wie Tabelle oben , aber ohne:

- Projektierung Verbindungen
- Programmierung Datenaustausch

29.4 Überblick Anwenderschnittstellen

Überblick Kommunikationsbausteine:

Tabelle 29-6

Kommunikationsbaustein	S7-300, ET 200 CPU		S7-400
	CPU (*1)	CP (*2)	CPU, CP (*3)
USEND / URCV	FB 8 / FB 9	FB 8 / FB 9	SFB 8 / SFB 9
BSEND / BRCV	FB 12 / FB 13	FB 12 / FB 13	SFB 12 / SFB 13
PUT	FB 15	FB 15	SFB 15
GET	FB 14	FB 14	SFB 14
USEND_E / URCV_E	FB 28 / FB 29 (*1)	---	---
PUT_E	FB 35 (*1)	---	---
GET_E	FB 34 (*1)	---	---

(*1): Die Kommunikationsbausteine unterstützen derzeit nur Verbindungen über die PN-Schnittstelle der S7-300 CPU.

Kommunikationsbausteine in STEP 7:

Tabelle 29-7

Schnittstelle		Aus STEP 7 Bibliothek
S7-300	CPU	Standard Library / Communication Blocks
	CP	SIMATIC_NET_CP / CP300
S7-400	CPU, CP	Standard Library / System Function Blocks

29.5 Anwenderschnittstelle: USEND / URCV

29.5.1 Beschreibung

Der Kommunikationsbaustein USEND sendet Daten an einen Kommunikationsbaustein URCV. URCV empfängt Daten vom USEND, und kopiert sie in die projektierten Empfangsbereiche.

USEND und URCV sind mit einander nicht koordiniert:

- Ein Sendeauftrag ist abgeschlossen, wenn die Daten im Empfangspuffer angekommen sind. Die Daten liegen dann im Allgemeinen noch nicht im Empfangsbereich. D.h. bei einem abgeschlossenen Sendeauftrag ist nicht sichergestellt, dass mit URCV die Daten vom Empfangspuffer in den Empfangsbereich kopiert wurden.
- Ein neuer Sendeauftrag (USEND) kann gestartet werden, auch wenn die vorher gesendeten Daten vom URCV noch nicht in den Empfangsbereich kopiert wurden. D.h. Daten können, unbemerkt vom Sender, im Empfangspuffer überschrieben werden. Der URCV, im Empfänger, bringt in diesem Fall eine Overrun Warnung.

USEND

Unkoordiniertes Senden von Daten

URCV

Unkoordiniertes Empfangen von Daten

29.5.2 Parameter für USEND

Tabelle 29-8

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
ID	WORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
R_ID	DWORD	Zuordnung Sende SFB/FB und Empfangs SFB/FB. Dies ermöglicht die Kommunikation mehrerer SFB/FB Paare über dieselbe logische Verbindung.
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
SD_i	ANY	Sendebereich (i=1,2,3,4) (*2)

(*1): remote Quittierung: Transport

(*2): Tabelle 29-9

	S7-300	S7-400
SIMATIC S7 Speicherbereiche	M, D	E, A, M, D, T, Z
Anzahl Sendebereiche	1 Sendebereich	bis zu 4 Sendebereiche

29.5.3 Parameter für URCV

Tabelle 29-10

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
ID	WORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
R_ID	DWORD	Zuordnung Sende SFB/FB und Empfangs SFB/FB. Dies ermöglicht die Kommunikation mehrerer SFB/FB Paare über dieselbe logische Verbindung.
OUTPUT	Typ	Bemerkung
NDR	BOOL	Daten im Eingangspuffer: ja / nein
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
RD_i	ANY	Empfangsbereich (i=1,2,3,4) (*1)

(*1):Tabelle 29-11

	S7-300	S7-400
SIMATIC S7 Speicherbereiche	M, D	E, A, M, D, T, Z
Anzahl Empfangsbereiche	1 Empfangsbereich	bis zu 4 Empfangsbereiche

29.6 Anwenderschnittstelle: BSEND / BRCV

29.6.1 Beschreibung

Der Kommunikationsbaustein BSEND sendet Daten an einen Kommunikationsbaustein BRCV. BRCV empfängt Daten vom BSEND, und kopiert sie in die projektierten Empfangsbereiche.

BSEND und BRCV sind mit einander koordiniert:

- Ein Sendeauftrag ist abgeschlossen, wenn die Daten im Empfangsbereich angekommen sind. D.h. bei einem abgeschlossenen Sendeauftrag ist sichergestellt, dass mit BRCV die Daten in den Empfangsbereich geschrieben wurden.
- Ein neuer Sendeauftrag (BSEND) kann erst gestartet werden, wenn die vorher gesendeten Daten vom BRCV in den Empfangsbereich übernommen wurden.

Hinweis zur internen Arbeitsweise

Die zu sendenden Daten wird in Datenblöcke unterteilt. Jeder Datenblock wird einzeln an den Kommunikationspartner gesendet (blockorientiertes Senden). Nach jedem empfangenen Datenblock wird eine Quittung an den BSEND geschickt, und der Parameter LEN wird aktualisiert (blockorientiertes Empfangen).

BSEND

Blockorientiertes Senden von Daten

BRCV

Blockorientiertes Empfangen von Daten

29.6.2 Parameter für BSEND

Tabelle 29-12

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Sendeauftrag
ID	WORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
R_ID	DWORD	Zuordnung Sende SFB/FB und Empfangs SFB/FB. Dies ermöglicht die Kommunikation mehrerer SFB/FB Paare über dieselbe logische Verbindung.
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
SD_1	ANY	Sendebereich (*2)
LEN	WORD	Länge der zu sendenden Daten

(*1): remote Quittierung: Applikation

(*2): SIMATIC S7 Speicherbereiche:

- S7-300: M, D
- S7-400: E, A, M, D, T, Z

29.6.3 Parameter für BRCV

Tabelle 29-13

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
ID	WORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
R_ID	DWORD	Zuordnung Sende SFB/FB und Empfangs SFB/FB. Dies ermöglicht die Kommunikation mehrerer SFB/FB Paare über dieselbe logische Verbindung.
OUTPUT	Typ	Bemerkung
NDR	BOOL	Daten im Empfangsbereich: ja / nein
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
RD_1	ANY	Empfangsbereich (*1)
LEN	WORD	Länge der empfangenen Daten

(*1): SIMATIC S7 Speicherbereiche

- S7-400: E, A, M, D, T, Z
- S7-300: M, D

29.7 Anwenderschnittstelle PUT, GET

29.7.1 Beschreibung

Mit den Kommunikationsbausteinen kommuniziert eine CPU mit einer anderen CPU (Client / Server Kommunikation).

PUT

Daten in CPU schreiben

GET

Daten aus CPU lesen

29.7.2 Parameter für PUT

Tabelle 29-14

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Schreibauftrag
ID	WORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
ADDR_i	ANY	Zielbereich (i=1,2,3,4) (*2)
SD_i	ANY	Quellbereich (i=1,2,3,4) (*2)

(*1): remote Quittierung: Applikation

(*2): Tabelle 29-15

	S7-300	S7-400
SIMATIC S7 Speicherbereiche	M, D	E, A, M, D, T, Z
Anzahl Bereiche	1 Bereich	bis zu 4 Bereiche

29.7.3 Parameter für GET

Tabelle 29-16

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Leseauftrag
ID	WORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
OUTPUT	Typ	Bemerkung
NDR	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
ADDR_i	ANY	Zielbereich (i=1,2,3,4) (*2) (*3)
RD_i	ANY	Quellbereich (i=1,2,3,4) (*2) (*3)

(*1): remote Quittierung: Applikation

(*2): Tabelle 29-17

	S7-300	S7-400
SIMATIC S7 Speicherbereiche	M, D	E, A, M, D, T, Z
Anzahl Bereiche	1 Bereich	bis zu 4 Bereiche

29.8 Anwenderschnittstelle: USEND_E / URCV_E

Hinweis

Die Kommunikationsbausteine unterstützen derzeit nur Verbindungen über die PN-Schnittstelle.

29.8.1 Beschreibung

Siehe Kapitel 29.5.1.

29.8.2 Parameter für USEND_E

Siehe Kapitel 29.5.2.

Unterschiede der Kommunikationsbausteine:

Tabelle 29-18

	S7-300 CPUs, ET 200 CPUs	
	USEND	USEND_E
SIMATIC S7 Speicherbereiche	M, D	E, A, M, D, T, Z
Anzahl Sendebereiche	1 Sendebereich	bis zu 4 Sendebereiche

29.8.3 Parameter für URCV_E

Siehe Kapitel 29.5.3.

Unterschiede der Kommunikationsbausteine:

Tabelle 29-19

	S7-300 CPUs, ET 200 CPUs	
	URCV	URCV_E
SIMATIC S7 Speicherbereiche	M, D	E, A, M, D, T, Z
Anzahl Empfangsbereiche	1 Empfangsbereich	bis zu 4 Empfangsbereiche

29.9 Anwenderschnittstelle PUT_E, GET_E

Hinweis

Die Kommunikationsbausteine unterstützen derzeit nur Verbindungen über die PN-Schnittstelle.

29.9.1 Beschreibung

Siehe Kapitel 29.7.1.

29.9.2 Parameter für PUT_E

Siehe Kapitel 29.7.2.

Unterschiede der Kommunikationsbausteine:

Tabelle 29-20

	S7-300 CPUs, ET 200 CPUs	
	PUT	PUT_E
SIMATIC S7 Speicherbereiche	M, D	E, A, M, D, T, Z
Anzahl Bereiche	1 Bereich	bis zu 4 Bereiche

29.9.3 Parameter für GET_E

Siehe Kapitel 29.7.3

Unterschiede der Kommunikationsbausteine:

Tabelle 29-21

	S7-300 CPUs, ET 200 CPUs	
	GET	GET_E
SIMATIC S7 Speicherbereiche	M, D	E, A, M, D, T, Z
Anzahl Bereiche	1 Bereich	bis zu 4 Bereiche

30 PROFINET/Industrial Ethernet (PN/IE)

30.1 Merkmale

Die Kommunikation über PN/IE zeichnet sich durch folgende Merkmale aus:

- Kommunikation zwischen SIMATIC Controllern, und
- Kommunikation mit fremden Controllern

30.2 Übersicht

Über PN/IE sind folgende Kommunikationsarten möglich:

Tabelle 30-1

	Kommunikationsart		Kapitel
SIMATIC S7 spezifisch	S7-Kommunikation		29
Offener Standard	Offene-Kommunikation mit Send/Receive-Bausteinen		31
	Offene-Kommunikation mit T-Bausteinen		32
	PN-Kommunikation	CBA	33
PNIO		34	

Die Kommunikationsarten zum „Offenen Standard“ werden im Folgenden beschrieben.

Hinweis

Die Kommunikation mit MODICON Geräten über das Netz PN/IE, wird im Teil 4 der Dokumentation beschrieben.

Tabelle 30-2

	Kommunikationsart	Kapitel
Kopplung über Modbus	Modbus/TCP	44

31 PN/IE: Offene-Kommunikation mit Send/Receive-Bausteinen

31.1 Merkmale

Die Offene-Kommunikation mit Send/Receive-Bausteinen zeichnet sich durch folgende Merkmale aus:

- Offener Standard (Kommunikation mit fremden Controllern möglich)
- Kommunikation immer über CP
- Kommunikation über unterschiedliche Protokolle: TCP, ISO on TCP, UDP
- Kommunikation über Verbindungen (nicht beim Protokoll UDP)
- Verbindungen werden projiziert
- Anzahl Daten pro Kommunikationsauftrag: ≤ 8 KByte
- Quittierung vom remote Transportsystem (nicht beim Protokoll UDP).
- Keine Quittierung von der remote Applikation (alle Protokolle)

Projektierung von Verbindungen

Verbindungen werden in STEP 7 (NetPro) projiziert. Aufbau und Abbau der Verbindungen erfolgt durch das Betriebssystem des CP. Durch die Projektierung der Verbindungen sind die Verbindungsparameter zur Laufzeit nicht änderbar.

Besonderheit beim Protokoll UDP

In STEP 7 (NetPro) muss eine „UDP-Verbindung“ projiziert werden. Eine „UDP-Verbindung“ ist jedoch keine Verbindung im Sinne des Dokumentes (Kapitel 5). Es wird keine Verbindung zum remote Kommunikationspartner aufgebaut.

Verbindungstypen

Tabelle 31-1

Protokoll	Verbindungstypen	
TCP	B#16#01, B#16#11	nach RFC 793
ISO on TCP	B#16#12	nach RFC 1006
UDP	B#16#13	nach RFC 768

Besonderheit beim Protokoll UDP

In STEP 7 (NetPro) muss eine „UDP-Verbindung“ projiziert werden. Eine „UDP-Verbindung“ ist jedoch keine Verbindung im Sinne des Dokumentes (Kapitel 5). Es wird keine Verbindung zum remote Kommunikationspartner aufgebaut.

Merkmale Protokolle

Tabelle 31-2

Protokoll	Verbindungsaufbau zum remote Partner	Übertragung von Länge und Ende der Daten
TCP	ja	nein (*1)
ISO on TCP	ja	ja
UDP	nein	ja

(*1): Beim Senden ist dies unproblematisch, da der Sender weiß, wie viele Daten er verschicken will. Der Empfänger jedoch hat keine Möglichkeit zu erkennen, wo die Daten enden.

31.2 Eigenschaften

Erläuterungen zum Aufbau und Inhalt der Tabelle sind in Kapitel 25.2 zu finden.

Tabelle Kommunikationsarten-Detail:

Tabelle 31-3

Kommunikationsart:		Offene-Kommunikation mit Send/Receive-Bausteinen						
Protokoll:		ISO		ISO on TCP		TCP		UDP
Allgemeines								
Medien		PN/IE		PN/IE		PN/IE		PN/IE
Schnittstellen		CP		CP		CP		CP
Anbindung	SIMATIC S5	ja		ja		ja		ja
	Fremd (offene Standards)	nein		ja		ja		ja
Anwenderschnittstelle								
Kommunikationsbausteine (*2)		AG_xSEND / AG_xREC	FETCH, WRITE	AG_xSEND / AG_xREC	FETCH, WRITE	AG_xSEND / AG_xREC	FETCH, WRITE	AG_xSEND / AG_xREC
maximale Anzahl Daten (*2)		<= 8 KByte		<= 8 KByte		<= 8 KByte		<= 2 KByte
Anzahl Variablen bei Aufruf Kommunikationsbaustein		1	---	1	---	1	---	1
dynamische Adressierung Daten		ja	---	ja	---	ja	---	ja
remote Quittierung		Transport	---	Transport	---	Transport	---	nein
Modell		Client / Client	S7 nur Server	Client / Client	S7 nur Server	Client / Client	S7 nur Server	Client / Client

Kommunikationsart:		Offene-Kommunikation mit Send/Receive-Bausteinen			
Protokoll:		ISO	ISO on TCP	TCP	UDP
Protokoll					
dynamische Datenlänge		ja	ja	nein	ja
Multicast / Broadcast		nein	nein	nein	ja / ja
Verbindungen	zum remote Partner?	ja	ja	ja	nein
	dynamisch / statisch	statisch	statisch	statisch	statisch
Routingfähig		nein	ja	ja	ja
Anwendertätigkeit					
Kommunikationsbeziehung	Festlegen mit	NetPro	NetPro	NetPro	NetPro
Verbindungen	proj. mit NetPro?	ja	ja	ja	ja (*1)
	prog. in STEP 7?	nein	nein	nein	nein
Datenübertragung	Kommunikationsbaustein in STEP 7?	Client: ja Server: nein	Client: ja Server: nein	Client: ja Server: nein	Client: ja Server: nein

[Zurück zum Sprungverteiler PN/IE](#)

Erläuterungen zur Tabelle:

(*1): In STEP 7 (NetPro) muss eine „UDP-Verbindung“ projiziert werden. Eine „UDP-Verbindung“ ist jedoch keine Verbindung im Sinne des Dokumentes (Kapitel 5). Es wird keine Verbindung zum remote Kommunikationspartner aufgebaut.

(*2): Die maximale Anzahl Daten ist abhängig von:

- Kommunikationspartner (S7-300, S7-400, ...)
- Kommunikationsbaustein (AG_SEND, AG_LSEND, AG_SSEND)
- Protokoll (ISO, ISO on TCP, ...)

Übersicht:

Tabelle 31-4

Kommunikationsbaustein	Protokoll	S7-300	S7-400	Bemerkung
		CP	CP	
AG_SEND / AG_RECV	ISO, TCP, ISO on TCP	8 KByte	240 Byte	----
	UDP	2 KByte	240 Byte	inklusive IP-Header und UDP-Header
AG_LSEND / AG_LRECV	ISO, TCP, ISO on TCP	---	8 KByte	---
	UDP	---	2 KByte	inklusive IP-Header und UDP-Header
AG_SSEND / AG_SRECV	ISO, TCP, ISO on TCP	---	1452 Byte	---
	UDP	---	1452 Byte	inklusive IP-Header und UDP-Header

31.3 Anwendung

Es folgt eine Übersicht der wichtigsten Tätigkeiten, um eine CPU-CPU Kommunikation zu realisieren.

Client / Client Kommunikation

Auf beiden Seiten der Kommunikationsbeziehung:

Tabelle 31-5

Tätigkeit	Engineering Tool
Vernetzung und Adressvergabe	STEP 7, HW Konfig
Projektierung der Verbindungen: <ul style="list-style-type: none"> • Auswahl der Baugruppen, die kommunizieren sollen • Auswahl Verbindungstyp • Parametrierung Verbindung 	STEP 7, NetPro
Programmierung Datenaustausch: Aufruf Kommunikationsbausteine im STEP 7 Anwenderprogramm	STEP 7, Sprachen Editor

Client / Server Kommunikation

Auf Client Seite der Kommunikationsbeziehung:

SIMATIC S7 Controller sind nur Server.

Auf Server Seite der Kommunikationsbeziehung:

Wie Tabelle oben , aber ohne:

- Programmierung Datenaustausch

31.4 Überblick Anwenderschnittstellen

Send/Receive-Bausteine

Überblick Kommunikationsbausteine:

Tabelle 31-6

Kommunikationsbaustein	S7-300	S7-400
	CP	CP
AG_SEND / AG_RECV	FC 5 / FC 6	FC 5 / FC 6
AG_LSEND / AG_LREC	---	FC 50 / FC 60
AG_SSEND / AG_SREC	---	FC 53 / FC 63

Je nach Familie (S7-300, S7-400) sind unterschiedliche Kommunikationsbausteine zu verwenden. Die Kommunikationsbausteine sind in STEP 7 unter verschiedenen Bibliotheken abgelegt.

Kommunikationsbausteine in STEP 7:

Tabelle 31-7

Schnittstelle	STEP 7 Bibliothek
S7-300, CP	SIMATIC_NET_CP / CP300
S7-400, CP	SIMATIC_NET_CP / CP400

Server für Fetch/Write

Ein SIMATIC S7-CP kann Server für FETCH/WRITE Aufträge eines anderen Controllers (fremder Controller, SIMATIC S5) sein.

In der S7-CPU des Servers sind für den Datenaustausch keine Kommunikationsbausteine nötig.

31.5 Anwenderschnittstelle AG_xSEND, AG_xRECV

Für die Namen der Kommunikationsbausteine werden folgende Abkürzungen verwendet:

- AG_xSEND steht für: AG_SEND, AG_LSEND, AG_SSEND
- AG_xRECV steht für: AG_RECV, AG_LRECV, AG_SSRECV

Bedeutung von x = L

Die Kommunikationsbausteine sind optimiert für die Übertragung umfangreicher Daten (L steht für „long“).

Bedeutung von x = S

Die Kommunikationsbausteine sind zeitoptimiert (S steht für „speed“):

- Optimierte Kommunikation zwischen CPU und CP in der Station
- Ohne Auswirkung auf die Kommunikation über das Netz

31.5.1 Beschreibung

Der Kommunikationsbaustein AG_xRECV sendet Daten an den Kommunikationsbaustein AG_xRECV.

Die Arbeitsweise der Kommunikationsbausteine ist abhängig vom verwendeten CP (/13/).

AG_xSEND

Der Kommunikationsbaustein übergibt Daten an den CP, die über eine projektierte Verbindung gesendet werden.

AG_xRECV

Der Kommunikationsbaustein übernimmt Daten vom CP, die über eine projektierte Verbindung empfangen wurden.

31.5.2 Parameter für AG_SEND, AG_LSEND, AG_SSEND

Tabelle 31-8

INPUT	Typ	Erläuterung
ACT	BOOL	Anstoß Sendeauftrag
ID	INT	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
LADDR	WORD	Adresse der Baugruppe (aus Hardware-Konfiguration in STEP 7)
SEND	ANY	Sendebereich (*2)
LEN	INT	Länge der zu sendenden Daten
OUTPUT	Typ	Erläuterung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): „Auftrag fertig“:

Unterschiedliche Bedeutung, je nach verwendetem Protokoll:

- Daten liegen auf dem remote CP: ISO, ISO on TCP, TCP
- Daten wurden vom lokalen CP gesendet: UDP

(*2): SIMATIC S7 Speicherbereiche: M, D

31.5.3 Parameter für AG_RECV, AG_LRECV, AG_SSRECV

Tabelle 31-9

INPUT	Typ	Erläuterung
ID	INT	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
LADDR	WORD	Adresse der Baugruppe (aus Hardware-Konfiguration in STEP 7)
RECV	ANY	Empfangsbereich (*2)
OUTPUT	Typ	Erläuterung
NDR	BOOL	Daten im Empfangsbereich: ja (NDR=1) / nein (NDR=0) (*1)
LEN	INT	Länge der empfangenen Daten
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): "ja": Die Daten wurden aus dem Empfangspuffer (CP) in den Empfangsbereich (CPU) kopiert.

Sonderfall TCP-Verbindung:

Hier ist die Längenangabe im Parameter RECV maßgebend.

Ein Empfangsauftrag wird mit der Anzeige NDR=1 abgeschlossen, sobald eine der Längenangabe entsprechende Anzahl Daten in den Empfangsbereich geschrieben wurde. D.h. NDR wird erst gesetzt, wenn der Empfangsbereich komplett gefüllt worden ist. LEN zeigt immer die Länge des Empfangsbereichs.

(*2): SIMATIC S7 Speicherbereiche: M, D

31.6 Anwenderschnittstelle FETCH, WRITE (Server)

In der S7-CPU des Servers sind für den Datenaustausch keine Kommunikationsbausteine nötig.

FETCH (Daten holen)

Der Verbindungspartner (fremder Controller, SIMATIC S5) kann lesend auf Daten in der S7-CPU (Server) zugreifen.

WRITE (Daten schreiben)

Der Verbindungspartner (fremder Controller, SIMATIC S5) kann schreibend auf Daten in der SIMATIC S7 zugreifen.

Daten

Auf folgende Daten kann in der S7-CPU (Server) zugegriffen werden:

- Datenbausteine
- Merker
- Prozessabbild Eingänge und Ausgänge
- Peripheriebereich Eingänge und Ausgänge
- Zähler, Zeiten

Verbindungstypen

Für den Zugriff mit FETCH oder WRITE muss in der S7-CPU (Server) eine Verbindung in der Betriebsart FETCH passiv oder WRITE passiv projektiert werden. Möglich sind folgende Verbindungstypen:

- ISO
- ISO on TCP
- TCP

Zugriffskordinierung über das STEP 7 Anwenderprogramm

Für die Zugriffskordinierung stehen die Bausteine (FC) AG_LOCK und AG_UNLOCK zur Verfügung. Mit diesen Bausteinen kann der Zugriff auf Daten durch Sperren oder Freigeben der Verbindungen koordiniert werden.

32 PN/IE: Offene-Kommunikation mit T-Bausteinen

32.1 Merkmale

Die Offene-Kommunikation mit T-Bausteinen zeichnet sich durch folgende Merkmale aus:

- Offener Standard (Kommunikation mit fremden Controllern möglich)
- Kommunikation über CPU oder CP
- Kommunikation über unterschiedliche Protokolle: TCP, ISO on TCP, UDP
- Kommunikation über Verbindungen (nicht beim Protokoll UDP)
- Verbindungen werden programmiert
- Anzahl Daten pro Kommunikationsauftrag: ≤ 64 KByte
- Quittierung vom remote Transportsystem (nicht beim Protokoll UDP).
- Keine Quittierung von der remote Applikation. (gilt für alle Protokolle)

Programmierung von Verbindungen

Aufbau und Abbau der Verbindungen werden im STEP 7 Anwenderprogramm programmiert. Dazu sind spezielle Bausteine aufzurufen (Verbindungsbausteine). Durch die Programmierung der Verbindungen sind die Verbindungsparameter zur Laufzeit (im RUN der CPU) änderbar.

Hinweis zur S7-1200

Dort gibt es zusätzlich die T-Compact-Bausteine. In diesen Kommunikationsbausteinen sind die Verbindungsbausteine integriert.

Besonderheit beim Protokoll UDP:

Hier wird der lokale Kommunikationszugangspunkt mit den Verbindungsbausteinen parametrieret. Es wird keine Verbindung zum remote Kommunikationspartner aufgebaut.

Verbindungstypen

Tabelle 32-1

Protokoll	Verbindungstypen	
TCP	B#16#01, B#16#11	nach RFC 793
ISO on TCP	B#16#12	nach RFC 1006
UDP	B#16#13	nach RFC 768

Besonderheit beim Protokoll UDP:

Hier wird der lokale Kommunikationszugangspunkt mit den Verbindungsbausteinen parametrieret. Es wird keine Verbindung zum remote Kommunikationspartner aufgebaut.

Merkmale Protokolle

Tabelle 32-2

Protokoll	Verbindungsaufbau zum remote Partner	Übertragung von Länge und Ende der Daten
TCP	ja	nein (*1)
ISO on TCP	ja	ja
UDP	nein	ja

(*1): Ist beim Senden unproblematisch, da der Sender weiß, wie viele Daten er verschicken will. Der Empfänger jedoch hat keine Möglichkeit zu erkennen, wo die Daten enden.

32.2 Eigenschaften

Erläuterungen zum Aufbau und Inhalt der Tabelle sind in Kapitel 25.2 zu finden.

Tabelle Kommunikationsarten-Detail:

Tabelle 32-3

Kommunikationsart:		Offene-Kommunikation mit T-Bausteinen				
Protokoll:		ISO on TCP		TCP		UDP
Allgemeines						
Medien		PN/IE		PN/IE		PN/IE
Schnittstellen		CPU, CP		CPU, CP		CPU, CP
Anbindung	SIMATIC S5	ja		ja		nein
	Fremd (offene Standards)	ja		ja		ja
Anwenderschnittstelle						
Kommunikationsbausteine		TSEND / TRCV	TSEND_C / TRCV_C	TSEND / TRCV	TSEND_C / TRCV_C	TUSEND / TURCV
maximale Anzahl Daten (*1)		<= 32 KByte	= 8192 Byte	<= 64 KByte	= 8192 Byte	= 1472 Byte
Anzahl Variablen bei Aufruf Kommunikationsbaustein		1	1	1	1	1
dynamische Adressierung Daten		ja	ja	ja	ja	ja
remote Quittierung		Transport	Transport	Transport	Transport	nein
Modell		Client / Client	Client / Client	Client / Client	Client / Client	Client / Client

Kommunikationsart:		Offene-Kommunikation mit T-Bausteinen		
Protokoll:		ISO on TCP	TCP	UDP
Protokoll				
dynamische Datenlänge		ja	nein	ja
Multicast / Broadcast		nein	nein	nein
Verbindungen	zum remote Partner?	ja	ja	nein
	dynamisch / statisch	TSEND/TRCV: dynamisch + statisch TSEND_C/TRCV_C: dynamisch	TSEND/TRCV: dynamisch + statisch TSEND_C/TRCV_C: dynamisch	dynamisch + statisch
Routingfähig		ja	ja	ja
Anwendertätigkeit				
Kommunikationsbeziehung	Festlegen mit	HW Konfig bzw. Geräte & Netze (S7-1200)	HW Konfig bzw. Geräte & Netze (S7-1200)	HW Konfig bzw. Geräte & Netze (S7-1200)
Verbindungen	projektieren?	nein	nein	nein
	prog. in STEP 7?	ja (Verbindungsbausteine)	ja (Verbindungsbausteine)	ja (Verbindungsbausteine) (*2)
Datenübertragung	Kommunikationsbaustein in STEP 7?	ja (Kommunikationsbausteine)	ja (Kommunikationsbausteine)	ja (Kommunikationsbausteine)

[Zurück zum Sprungverteiler PN/IE](#)

Erläuterungen zur Tabelle:

(*1): Die maximale Anzahl Daten ist abhängig von:

- Kommunikationspartner (S7-300, S7-400, ...) und verwendete Schnittstelle (CPU, CP)
- Kommunikationsbaustein (TSEND, ...)
- Protokoll (TCP, ...) und Verbindungstyp

Übersicht:

Tabelle 32-4

Kommunikationsbaustein	Protokoll	Verbindungstyp	S7-1200	ET 200 CPU	S7-300	S7-400		S7-mEC	WinAC RTX 2009
			CPU	CPU	CPU	CPU	CP	CPU	CPU / CP (Submodul)
				151-8, 154-8		EC31			
TSEND/TRCV	TCP	B#16#01	1460 Byte	1460 Byte	1460 Byte	---	---	---	---
		B#16#11	8 KByte	32 KByte	32 KByte	32 KByte	---	8 KByte	64 KByte
	ISO on TCP	B#16#12	8 KByte	32 KByte	32 KByte	32 KByte	1452 Byte	---	---
TUSEND/TURCV	UDP	B#16#13	1472 Byte	1472 Byte	1472 Byte	1472 Byte	---	1472 Byte	1472 Byte

(*2): Hier wird der lokale Kommunikationszugangspunkt mit den Verbindungsbausteinen parametrierter. Es wird keine Verbindung zum remote Kommunikationspartner aufgebaut.

32.3 Anwendung

Es folgt eine Übersicht der wichtigsten Tätigkeiten, um eine CPU-CPU Kommunikation zu realisieren.

Fallunterscheidung:

- TSEND/TRCV und TUSEND/TURCV
- TSEND_C/TRCV_C

Bei allen Fällen handelt es sich um eine Client / Client Kommunikation.

TSEND/TRCV und TUSEND/TURCV

Tabelle 32-5

Tätigkeit	Engineering Tool	
	SIMATIC (ohne S7-1200)	S7-1200
Vernetzung und Adressvergabe	STEP 7, HW Konfig	STEP 7 Basic, Geräte & Netze
Programmierung der Verbindungen: <ul style="list-style-type: none"> • Aufruf Verbindungsbausteine im STEP 7 Anwenderprogramm • Parameter für jede Verbindung in einem Datenbaustein ablegen (*1) 	STEP 7, Sprachen Editor	STEP 7 Basic
Programmierung Datenaustausch: Aufruf Kommunikationsbausteine im STEP 7 Anwenderprogramm	STEP 7, Sprachen Editor	STEP 7 Basic

(*1): Der Anwender wird unterstützt durch Wizards:

- SIMATIC (ohne S7-1200): Open Communication Wizard /29/
- S7-1200: Wizard in STEP 7 Basic

TSEND_C/TRCV_C

Tabelle 32-6

Tätigkeit	Engineering Tool
	S7-1200
Vernetzung und Adressvergabe	STEP 7 Basic, Geräte & Netze
Programmierung: <ul style="list-style-type: none"> • Aufruf Kommunikationsbausteine im STEP 7 Anwenderprogramm. Die Bausteine realisieren den Datenaustausch <u>und</u> den Verbindungsaufbau (Verbindungsbausteine sind in den Kommunikationsbausteinen integriert). • Parameter für Verbindungen und Datenübertragung in einem Datenbaustein ablegen (*1) 	STEP 7 Basic

(*1): Der Anwender wird durch einen Wizard in STEP 7 Basic unterstützt.

32.4 Überblick Anwenderschnittstellen

T-Bausteine

Überblick Kommunikationsbausteine

Tabelle 32-7

Kommunikationsbaustein	S7-300	S7-400	S7-1200
	CPU	CPU, CP	CPU
TSEND / TRCV	FB 63 / FB 64	FB 63 / FB 64	(*1)
TSEND_C / TRCV_C	---	---	(*1)
TUSEND / TURCV	FB 67 / FB 68	FB 67 / FB 68	(*1)

(*1): Die Bausteine gehören zum Sprachumfang der S7-1200: Programmieranweisungen / Kommunikationsoperationen / Open Ethernet Kommunikation

Überblick: Verbindungsbausteine (nur für Kommunikation über CPU).

Tabelle 32-8

Verbindungsbaustein	S7-300	S7-400	S7-1200
	CPU	CPU	CPU
TCON	FB 65	FB 65	(*1)
TDISCON	FB 66	FB 66	(*1)

(*1): Die Bausteine gehören zum Sprachumfang der S7-1200: Programmieranweisungen / Kommunikationsoperationen / Open Ethernet Kommunikation

Kommunikationsbausteine in STEP 7:

Tabelle 32-9

Schnittstellen	STEP 7 Bibliothek
CPU, CP	Standard Library / Communication Blocks

Server für Fetch/Write

Überblick: Kommunikationsbausteine

Tabelle 32-10

Kommunikationsbaustein	S7-300	S7-400
	CPU	CPU
FW_TCP	FB 210	FB 210
FW_IOT	FB 220	FB 220

Kommunikationsbausteine in STEP 7:

Tabelle 32-11

Schnittstellen	STEP 7 Bibliothek
CPU, CP	Standard Library / Communication Blocks

Hinweis

Eine SIMATIC S7-CPU kann Server für FETCH/WRITE Aufträge eines anderen Controllers (fremder Controller, SIMATIC S5) sein.

Die Kommunikationsbausteine des Servers verwenden intern die T-Bausteine:

- TSEND / TRCV (Senden und Empfangen von Daten)
- TCON, TDISCON (Verbindung aufbauen und abbauen)

32.5 Anwenderschnittstelle TSEND / TRCV

32.5.1 Beschreibung

Der Kommunikationsbaustein TSEND sendet Daten an den Kommunikationsbaustein TRCV.

Die Datenübertragung erfolgt über eine Verbindung, welche mit Verbindungsbausteinen programmiert wird.

TSEND

Senden von Daten

TRCV

Empfangen von Daten

32.5.2 Parameter für TSEND

Tabelle 32-12

INPUT	Typ	Erläuterung
REQ	BOOL	Anstoß Sendeauftrag
ID	WORD	Referenz auf die zugehörige Verbindung
LEN	INT	Länge der zu sendenden Daten
OUTPUT	Typ	Erläuterung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
BUSY	BOOL	
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Erläuterung
DATA	ANY	Sendebereich (*2)

(*1): remote Quittierung: Transport

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

32.5.3 Parameter für TRCV

Tabelle 32-13

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß: Empfangsauftrag
ID	WORD	Referenz auf die zugehörige Verbindung
LEN	INT	Länge des Empfangsbereiches
OUTPUT	Typ	Bemerkung
NDR	BOOL	Daten im Empfangsbereich: ja (NDR=1) / nein (NDR=0) (*1)
BUSY	BOOL	
ERROR	BOOL	Fehleranzeige
STATUS	WORD	
IN_OUT	Typ	Bemerkung
DATA	ANY	Empfangsbereich (*2)

(*1): „ja“: Fallunterscheidung:

Tabelle 32-14

Protokoll	LEN	Festlegung Empfangsbereich	Empfang von Daten
TCP	= 0	Adresse: DATA Länge: DATA	Unmittelbar nach Empfang von „DATA“ Daten, werden diese in den Empfangsbereich kopiert, und NDR wird auf 1 gesetzt. Maximal werden 1472 Byte empfangen.
	<> 0	Adresse: DATA Länge: LEN	Unmittelbar nach Empfang von „LEN“ Daten, werden diese in den Empfangsbereich kopiert, und NDR wird auf 1 gesetzt.
ISO on TCP	---	Adresse: DATA Länge: DATA	Sobald sämtliche Daten eines Auftrages empfangen wurden, werden diese in den Empfangsbereich kopiert, und NDR auf 1 gesetzt.

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

32.6 Anwenderschnittstelle TUSEND / TURCV

32.6.1 Beschreibung

Der Kommunikationsbaustein TUSEND sendet Daten an den Kommunikationsbaustein TURCV.

Die Datenübertragung erfolgt über eine Verbindung, welche mit Verbindungsbausteinen programmiert wird.

TUSEND

Senden von Daten

TURCV

Empfangen von Daten

32.6.2 Parameter für TUSEND

Tabelle 32-15

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
ID	WORD	Referenz auf zugehörige Verbindung
LEN	INT	Länge der zu sendenden Daten
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
BUSY	BOOL	
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
DATA	ANY	Sendebereich (*2)
ADDR	ANY	Adresse des Empfänger

(*1): remote Quittierung: keine

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

32.6.3 Parameter für TURCV

Tabelle 32-16

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
ID	WORD	Referenz auf zugehörige Verbindung
LEN	INT	Länge des Empfangsbereiches
OUTPUT	Typ	Bemerkung
NDR	BOOL	Daten im Empfangsbereich: ja (NDR=1) / nein (NDR=0) (*1)
BUSY	BOOL	
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
RCVD_LEN	INT	Anzahl der tatsächlich empfangenen Daten
IN_OUT	Typ	Bemerkung
DATA	ANY	Empfangsbereich (*2)
ADDR	ANY	Adresse des Senders

(*1): remote Quittierung: keine

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

32.7 Anwenderschnittstelle Verbindungsbausteine

32.7.1 Beschreibung

Mit dem Verbindungsbaustein TCON wird eine Verbindung aufgebaut, mit dem Verbindungsbaustein TDISCON wird eine Verbindung abgebaut.

Besonderheit beim Protokoll UDP

Bei UDP wird der lokale Kommunikationszugangspunkt mit den Verbindungsbausteinen parametrisiert. Es wird keine Verbindung zum remote Kommunikationspartner aufgebaut.

TCON

Aufbau einer Verbindung

TDISCON

Abbau einer Verbindung

32.7.2 Parameter für TCON

Tabelle 32-17

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Aufbau Verbindung
ID	WORD	Referenz auf die Verbindung
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig
BUSY	BOOL	
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
CONNECT	ANY	Verbindungsbeschreibung (UDT 65)

32.7.3 Parameter für TDISCON

Tabelle 32-18

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Abbau Verbindung
ID	WORD	Referenz auf die Verbindung
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig
BUSY	BOOL	
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

32.8 Anwenderschnittstelle TSEND_C / TRCV_C

Zur Unterscheidung von den anderen T-Bausteinen, werden diese Bausteine im Folgenden mit T-Compact-Bausteine bezeichnet.

32.8.1 Beschreibung

Der Kommunikationsbaustein TSEND_C sendet Daten an den Kommunikationsbaustein TRCV_C.

Die Datenübertragung erfolgt über eine Verbindung, welche von diesen Kommunikationsbausteinen aufgebaut und abgebaut wird.

Die Kommunikationsbausteine führen also folgende Funktionen aus:

- Aufbau und Abbau von Verbindungen
- Daten senden und empfangen

TSEND_C

Stellt eine Verbindung zum Partner her, sendet Daten und kann die Verbindung auch wieder beenden.

TSEND_C verbindet die Funktionen von TCON, TDISCON und TSEND.

TRCV_C

Stellt eine Verbindung zum Partner her, empfängt Daten und kann die Verbindung auch wieder beenden.

TRCV_C verbindet die Funktionen von TCON, TDISCON und TRCV.

32.8.2 Parameter für TSEND_C

Tabelle 32-19

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
CONT	BOOL	Verbindung aufbauen / abbauen
LEN	INT	Länge der zu sendenden Bytes
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
BUSY	BOOL	
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
DATA	Variante	Sendebereich (*2)
CONNECT	TCON Parameter	Verbindungsbeschreibung
COM_RST	BOOL	Neustart des Bausteines

(*1): remote Quittierung: Transport

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

32.8.3 Parameter für TRCV_C

Tabelle 32-20

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß: Empfangauftrag
CONT	BOOL	Verbindung aufbauen / abbauen
LEN	INT	Länge des Empfangsbereiches
OUTPUT	Typ	Bemerkung
DONE	BOOL	Daten im Empfangsbereich: ja (DONE=1) / nein (DONE=0) (*1)
BUSY	BOOL	
RCVD_LEN	INT	Anzahl empfangener Daten
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
DATA	Variante	Empfangsbereich (*2)
CONNECT	TCON Parameter	Verbindungsbeschreibung
COM_RST	BOOL	Neustart des Bausteines

(*1): "ja": Fallunterscheidung:

Tabelle 32-21

Protokoll	LEN	Festlegung Empfangsbereich	Empfang von Daten
TCP	= 0	Adresse: DATA Länge: DATA	Unmittelbar nach Empfang von „DATA“ Daten, werden diese in den Empfangsbereich kopiert, und DONE wird auf 1 gesetzt. Maximal werden 1472 Byte empfangen.
	<> 0	Adresse: DATA Länge: LEN	Unmittelbar nach Empfang von „LEN“ Daten, werden diese in den Empfangsbereich kopiert, und DONE wird auf 1 gesetzt.
ISO on TCP	---	Adresse: DATA Länge: DATA	Sobald sämtliche Daten eines Auftrages empfangen wurden, werden diese in den Empfangsbereich kopiert, und DONE auf 1 gesetzt.

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

32.9 Anwenderschnittstelle FETCH, WRITE (Server)

32.9.1 Beschreibung

Eine SIMATIC S7-CPU kann Server für FETCH/WRITE Aufträge eines anderen Controllers (fremder Controller, SIMATIC S5) sein.

Die Kommunikationsbausteine des Servers verwenden intern die T-Bausteine:

- TSEND / TRCV (Senden und Empfangen von Daten)
- TCON, TDISCON (Verbindung aufbauen und abbauen)

FW_TCP

Kommunikationsbaustein für Server, verwendetes Protokoll ist TCP

FW_IOT

Kommunikationsbaustein für Server, verwendetes Protokoll ist ISO on TCP

32.9.2 Parameter für FW_TCP

Tabelle 32-22

INPUT	Typ	Bemerkung
ENABLE	BOOL	Aufbau bzw. Abbau der Verbindung
CONNECT	ANY	Beschreibung TCP-Verbindung
ADDRMODE	INT	Adressiermodus S5 oder S7
OUTPUT	Typ	Bemerkung
NDR	BOOL	Daten des WRITE-Auftrages wurden übernommen
ERROR	BOOL	Fehleranzeige
MODE	BYTE	Ausführung FETCH oder WRITE Auftrag
STATUS	WORD	Statusanzeige

32.9.3 Parameter für FW_IOT

Tabelle 32-23

INPUT	Typ	Bemerkung
ENABLE	BOOL	Aufbau bzw. Abbau der Verbindung
CONNECT	ANY	Beschreibung IoT-Verbindung
ADDRMODE	INT	Adressiermodus S5 oder S7
OUTPUT	Typ	Bemerkung
NDR	BOOL	Daten des WRITE-Auftrages wurden übernommen
ERROR	BOOL	Fehleranzeige
MODE	BYTE	Ausführung FETCH oder WRITE Auftrag
STATUS	WORD	Statusanzeige

33 PN/IE: CBA

Im Dokument werden folgende Abkürzungen verwendet:

- CBA für PROFINET CBA
- PNIO für PROFINET IO

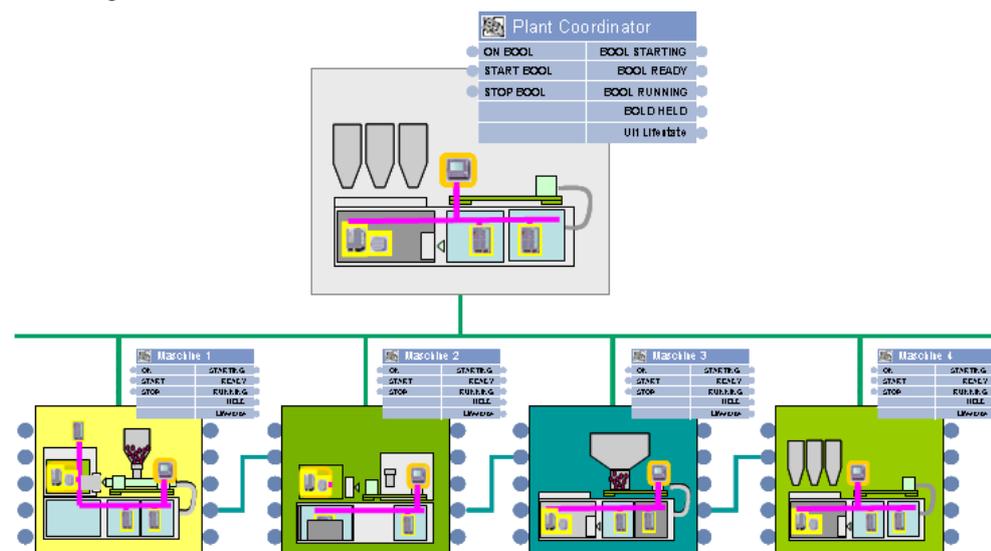
33.1 Merkmale

CBA

CBA (Component Based Automation) ist ein Automatisierungs-Konzept für die Realisierung modularer Applikationen auf Basis des offenen PROFINET Standards:

- Einfache Modularisierung von Anlagen und Produktionslinien durch dezentrale Intelligenz. Die Modularisierung erfolgt mit PROFINET-Komponenten.
- Maschine-Maschine Kommunikation entlang der Produktionslinie
- Grafische Konfiguration der Kommunikation

Abbildung 33-1

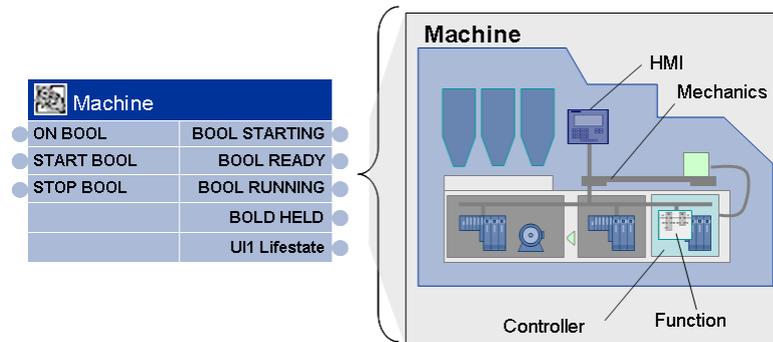


PROFINET-Komponente

Eine PROFINET-Komponente ist eine wiederverwendbare Funktionseinheit:

- Kapselung der Automatisierungsfunktionalität in einem Softwareprogramm
- Eindeutige Komponenten-Interfaces für den Datenaustausch mit anderen Komponenten.

Abbildung 33-2



Merkmale

- Grafische Konfiguration der Kommunikation (projektieren statt programmieren)
- Die Leistungsfähigkeit der Kommunikation ist offline kalkulierbar.
- Herstellerübergreifende Kommunikation

CBA und PNIO

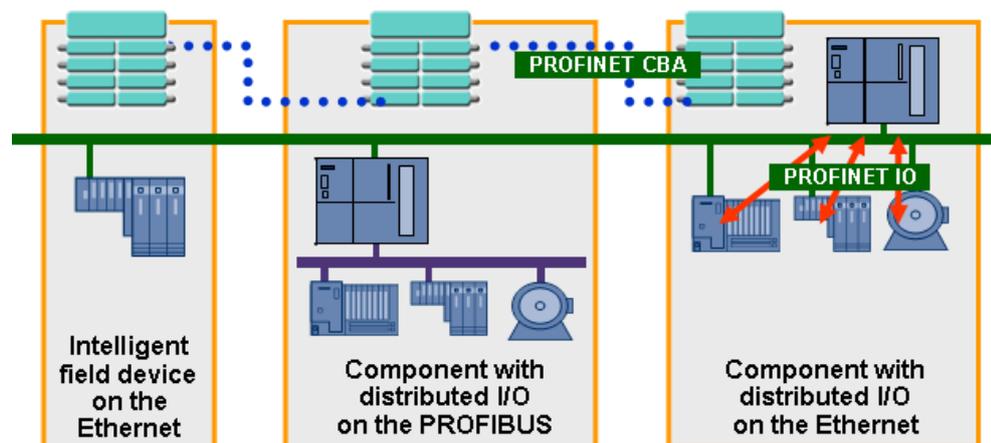
CBA (PROFINET CBA):

- Zyklischer und azyklischer Datenaustausch zwischen Controllern

PNIO (PROFINET IO):

- Zyklischer Datenaustausch zwischen einem PN Controller und den zugehörigen PN Devices.

Abbildung 33-3



33.2 Anwendung

CPU-CPU-Kommunikation

Es folgt eine Übersicht der wichtigsten Tätigkeiten, um eine CPU-CPU Kommunikation zu realisieren.

Tabelle 33-1

Tätigkeit		Engineering Tool
Komponenten Erstellung in STEP 7	Festlegen der Geräte und deren Funktionen Erstellung der Hardwarekonfiguration	STEP 7, HW Konfig
	PROFINET Interface erstellen Komponente erstellen	STEP 7, SIMATIC Manager
Komponente in iMAP importieren und verwenden Grafisches Verschalten der Komponenten.		SIMATIC iMAP

SIMATIC iMAP

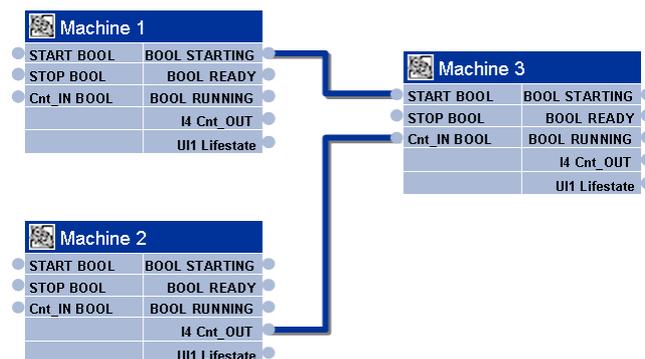
Mit SIMATIC iMAP werden PROFINET-Komponenten erzeugt und verschaltet.

PROFINET-Komponenten sind aufgrund ihrer standardisierten Schnittstellenbeschreibung herstellerunabhängig. Im Engineering Tool SIMATIC iMap können somit auch Fremdkomponenten importiert und verwendet werden.

33.3 Anwenderschnittstellen

Die Kommunikation zwischen PROFINET-Komponenten erfolgt durch grafisches Verschalten. D. h. im STEP 7 Anwenderprogramm muss zur Kommunikation nichts programmiert werden. Somit entfallen die Anwenderschnittstellen.

Abbildung 33-4



Ausnahme: Kommunikation über S7-CP.

In diesem Fall müssen sogenannte Copy-Bausteine (FB88, FB90) im STEP 7 Anwenderprogramm aufgerufen werden.

Die Bausteine stehen nach der Installation von SIMATIC iMap in der STEP 7 Bibliothek "PROFINET System-Library" in den Ordnern "CP 300" bzw. "CP 400" zur Verfügung.

34 PN/IE: PNIO

Im Dokument werden folgende Abkürzungen verwendet:

- CBA für PROFINET CBA
- PNIO für PROFINET IO

34.1 Merkmale

Vorbemerkung

Die Kommunikation mit PNIO ist ein Sonderfall der CPU-CPU Kommunikation.

Hier werden die Kommunikationsmechanismen der „dezentralen Peripherie“ für eine CPU-CPU Kommunikation verwendet:

- Eine CPU steckt in der zentralen Station
- Die andere CPU steckt in der dezentralen Station

Siehe dazu das Funktionsmodell: 4.4

Merkmale

Die Kommunikation mit PNIO zeichnet sich durch folgende Merkmale aus:

- zyklischer Austausch von Daten zwischen IO Controller und IO Device über das Protokoll PROFINET IO:
 - IO Controller sendet Daten an IO Device (Ausgangsbereich)
 - IO Device sendet Daten an IO Controller (Eingangsbereich)
- Der Datenaustausch erfolgt konsistent über die gesamte Länge (systembedingte Datenkonsistenz)

34.2 Eigenschaften

Erläuterungen zum Aufbau und Inhalt der Tabelle sind in Kapitel 25.2 zu finden.

Tabelle Kommunikationsarten-Detail:

Tabelle 34-1

Kommunikationsart:		PNIO	
Protokoll:		PN	
Allgemeines			
Medien		PN	
Schnittstellen		CPU, CP	
Anbindung	SIMATICS5	ja	
	Fremd (offene Standards)	ja	
Anwenderschnittstelle			
Kommunikationsbausteine		S7-CPU und S7-400 CP:	Ladebefehle / Transferbefehle DPRD_DAT, DPWR_DAT
		S7-300 CP:	PNIO_SEND, PNIO_RECV
maximale Anzahl Daten		Ladebefehl / Transferbefehl:	1, 2, 4 Byte
		DPRD_DAT, DPWR_DAT:	IO Controller <= 8192 Byte
		PNIO_SEND, PNIO_REC:	IO Device <= 1440 Byte
Anzahl Variablen bei Aufruf Kommunikationsbaustein		1 Variable	
dynamische Adressierung Daten		nein	
remote Quittierung		Applikation	
Modell		Consumer / Provider	

Kommunikationsart:		PNIO
Protokoll:		PN
Protokoll		
dynamische Datenlänge		nein
Multicast / Broadcast		nein
Verbindungen	zum remote Partner?	nein
	dynamisch / statisch	---
Routingfähig		nein
Anwendertätigkeit		
Kommunikationsbeziehung	Festlegen mit	HW Konfiguration
Verbindungen	proj. mit NetPro?	---
	prog. in STEP 7?	---
Datenübertragung	Kommunikationsbaustein in STEP 7?	ja (*1)

[Zurück zum Sprungverteiler PN/IE](#)

Erläuterungen zur Tabelle

(*1): Für eine CPU-CPU Kommunikation sind Kommunikationsbausteine nur in einer CPU erforderlich (nicht in beiden CPUs)

34.3 Anwendung

Es folgt eine Übersicht der wichtigsten Tätigkeiten, um eine CPU-CPU Kommunikation zu realisieren.

Tabelle 34-2

Tätigkeit	Engineering Tool
Projektierung der Hardware der Station: IO Controller	STEP 7, HW Konfig
Projektierung der Hardware der Station: IO Device	
Programmierung Datenaustausch	STEP 7, Sprachen Editor

34.4 Überblick Anwenderschnittstellen

Fallunterscheidung

S7-CPUs und S7-400 CPs:

- Ladebefehle und Transferbefehle
- Kommunikationsbausteine DPRD_DAT, DPWR_DAT

Nur bei S7-300 CP:

- Kommunikationsbausteine PNIO_SEND, PNIO_RECV

Kommunikationsbausteine

Überblick Kommunikationsbausteine:

Tabelle 34-3

Kommunikationsbaustein	S7-300		S7-400
	CPU	CP	CPU, CP
PNIO_SEND	---	FC 11	---
PNIO_RECV	---	FC 12	---
DPRD_DAT	SFC 14	---	SFC 14
DPWR_DAT	SFC 15	---	SFC 15

Kommunikationsbausteine in STEP 7:

Tabelle 34-4

Schnittstelle		STEP 7 Bibliothek
S7-300	CPU	Standard Library / System Function Blocks
S7-400	CPU, CP	
S7-300	CP	SIMATIC_NET_CP / CP300

34.5 Anwenderschnittstelle PNIO_SEND, PNIO_RECV

34.5.1 Beschreibung

Die Kommunikationsbausteine dienen der Kommunikation zwischen lokaler CPU und lokalem CP. Der CP ist IO Controller oder IO Device

PNIO_SEND

Der Kommunikationsbaustein übergibt Daten an den CP.

Fallunterscheidung:

CP ist IO Controller:

Die Daten, welche mit PNIO_SEND an den CP übergeben werden, werden vom CP an die IO Devices gesendet.

CP ist IO Device:

Die Daten, welche mit PNIO_SEND an den CP übergeben werden, werden an einen IO Controller gesendet.

PNIO_RECV

Der Kommunikationsbaustein übernimmt Daten vom CP.

Fallunterscheidung:

CP ist IO Controller:

Die Daten, welche mit PNIO_RECV vom CP übernommen werden, wurden von IO Devices an den CP gesendet.

CP ist IO Device:

Die Daten, welche mit PNIO_RECV vom CP übernommen werden, wurden von einem IO Controller an den CP gesendet.

34.5.2 Parameter für PNIO_SEND

Tabelle 34-5

INPUT	Typ	Erläuterung
CPLADDR	WORD	projektierte Anfangsadresse des CP
MODE	BYTE	CP ist IO Controller oder IO Device
LEN	INT	Länge der zu sendenden Daten
OUTPUT	Typ	Erläuterung
CHECK_IOCS	BOOL	IOCS Statusbereich
IOCS	ANY	
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
DONE	BOOL	Anzeige, ob Auftrag fehlerfrei ausgeführt wurde (*1)
IN_OUT	Typ	Erläuterung
SEND	ANY	Sendebereich (*2)

(*1): remote Quittierung: Applikation

(*2): SIMATIC S7 Speicherbereiche: M, D

34.5.3 Parameter für PNIO_RECV

Tabelle 34-6

INPUT	Typ	Erläuterung
CPLADDR	WORD	projektierte Anfangsadresse des CP
MODE	BYTE	CP ist IO Controller oder IO Device
LEN	INT	Länge der zu empfangenen Daten
OUTPUT	Typ	Erläuterung
CHECK_IOPS	BOOL	IOCS Statusbereich
IOPS	ANY	
NDR	BOOL	Anzeige, ob Auftrag fehlerfrei ausgeführt wurde (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
ADD_INFO	WORD	
IN_OUT	Typ	Erläuterung
RECV	ANY	Empfangsbereich (*2)

(*1): remote Quittierung: Applikation

(*2): SIMATIC S7 Speicherbereiche: M, D

34.6 Anwenderschnittstelle DPRD_DAT, DPWR_DAT

34.6.1 Beschreibung

Mit den Kommunikationsbausteinen hat ein IO Controller Zugriff auf die Daten eines IO Devices.

DPRD_DAT

Konsistente Daten eines IO Devices lesen

DPWR_DAT

Konsistente Daten in IO Device schreiben

34.6.2 Parameter für DPRD_DAT

Tabelle 34-7

INPUT	Typ	Bemerkung
LADDR	WORD	Projektierte Anfangsadresse im Eingangsbereich, aus dem gelesen werden soll
OUTPUT	Typ	Bemerkung
RET_VAL	INT	Anzeige, ob Auftrag fehlerfrei ausgeführt wurde (*1)
RECORD	ANY	Zielbereich (*2)

(*1): remote Quittierung: Applikation

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

34.6.3 Parameter für DPWR_DAT

Tabelle 34-8

INPUT	Typ	Bemerkung
LADDR	WORD	Projektierte Anfangsadresse im Ausgangsbereich, auf den geschrieben werden soll
RECORD	ANY	Quellbereich (*2)
OUTPUT	Typ	Bemerkung
RET_VAL	INT	Anzeige, ob Auftrag fehlerfrei ausgeführt wurde (*1)

(*1): remote Quittierung: Applikation

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

35 PROFIBUS (PB)

35.1 Merkmale

Die Kommunikation über PROFIBUS zeichnet sich durch folgende Merkmale aus:

- Kommunikation zwischen SIMATIC Controllern, und
- Kommunikation mit fremden Controllern

35.2 Übersicht

Über PB sind folgende Kommunikationsarten möglich:

Tabelle 35-1

	Kommunikationsart	Kapitel
SIMATIC S7 spezifische Kommunikation	S7-Basiskommunikation	28
	S7-Kommunikation	29
Offener Standard	Offene-Kommunikation mit Send/Receive-Bausteinen	36
	FMS-Kommunikation	37
	DP-Kommunikation	38

Die Kommunikationsarten zum „Offenen Standard“ werden im Folgenden beschrieben.

36 PB: Offene-Kommunikation mit Send/Receive-Bausteinen

36.1 Merkmale

Die Offene-Kommunikation mit Send/Receive-Bausteinen zeichnet sich durch folgende Merkmale aus:

- Offener Standard (Kommunikation mit fremden Controllern möglich)
- Kommunikation immer über CP
- Kommunikation über Verbindungen (FDL-Verbindungen)
- Verbindungen werden projiziert
- Anzahl Daten pro Kommunikationsauftrag: ≤ 240 Byte
- Quittierung vom remote Transportsystem
- Keine Quittierung von der remote Applikation

FDL-Verbindungen

Bei der Projektierung können einer FDL-Verbindung folgende Eigenschaften zugewiesen werden:

- Unspezifiziert
- Spezifiziert
- Multicast / Broadcast

Unspezifiziert

Eine unspezifizierte FDL-Verbindung kann auf 2 Arten genutzt werden:

- Verbindung zu einer Station in einem anderen STEP 7 Projekt
- Freier Layer 2 Zugang

Freier Layer 2 Zugang

Der remote Kommunikationspartner wird nicht bei der Projektierung festgelegt, sondern im STEP 7 Anwenderprogramm.

Dazu muss ein Auftragsheader (4 Byte) in den Daten eingebaut werden. Im Auftragsheader werden die Adresse der Zielstation, und der Dienst (SDA oder SDN) festgelegt.

Spezifiziert

Bei einer spezifizierten FDL-Verbindung wird der remote Kommunikationspartner bei der Projektierung festgelegt.

Die Daten enthalten keinen Auftragsheader.

Multicast / Broadcast

Die Daten enthalten einen Auftragsheader.

Der Auftragsheader hat keine Funktion.

36.2 Eigenschaften

Erläuterungen zum Aufbau und Inhalt der Tabelle sind in Kapitel 25.2 zu finden.

Tabelle Kommunikationsarten-Detail:

Tabelle 36-1

Kommunikationsart:		Offene-Kommunikation mit Send/Receive-Bausteinen
Protokoll:		FDL
Allgemeines		
Medien		PB
Schnittstellen		CP
Anbindung	SIMATIC S5	ja
	Fremd (offene Standards)	ja
Anwenderschnittstelle		
Kommunikationsbausteine		AG_SEND / AG_REC, AG_LSEND / AG_LREC
maximale Anzahl Daten (*1)		= 240 Byte
Anzahl Variablen bei Aufruf Kommunikationsbaustein		1 Variable
dynamische Adressierung Daten		ja
remote Quittierung		Transport
Modell		Client / Client

Kommunikationsart:		Offene-Kommunikation mit Send/Receive-Bausteinen
Protokoll:		FDL
Protokoll		
dynamische Datenlänge		ja
Multicast / Broadcast		ja / ja
Verbindungen	zum remote Partner?	ja
	dynamisch / statisch	statisch
Routingfähig		nein
Anwendertätigkeit		
Kommunikationsbeziehung	Festlegen mit	NetPro
Verbindungen	proj. mit NetPro?	spezifizierte FDL-Verbindung: ja unspezifizierte FDL-Verbindung: nein
	prog. in STEP 7?	spezifizierte FDL-Verbindung: nein unspezifizierte FDL-Verbindung: ja
Datenübertragung	Kommunikationsbaustein in STEP 7?	ja

[Zurück zum Sprungverteiler PB](#)

Erläuterungen zur Tabelle:

(*1): Der eventuell enthaltene Auftragsheader (4 Byte) ist darin enthalten.

36.3 Anwendung

Es folgt eine Übersicht der wichtigsten Tätigkeiten, um eine CPU-CPU Kommunikation zu realisieren.

Client / Client Kommunikation

Auf beiden Seiten der Kommunikationsbeziehung:

Tabelle 36-2

Tätigkeit	Engineering Tool
Vernetzung und Adressvergabe	STEP 7, HW Konfig
Projektierung der Verbindungen: <ul style="list-style-type: none"> • Auswahl der Baugruppen, die kommunizieren sollen • Auswahl Verbindungstyp • Parametrierung Verbindung und Dienst 	STEP 7, NetPro
Programmierung Datenaustausch: Aufruf Kommunikationsbausteine im STEP 7 Anwenderprogramm	STEP 7, Sprachen Editor

36.4 Überblick Anwenderschnittstellen

Überblick Kommunikationsbausteine:

Tabelle 36-3

Kommunikationsbaustein	S7-300	S7-400
	CP	CP
AG_SEND / AG_RECV	FC 5 / FC 6	FC 5 / FC 6
AG_LSEND / AG_LREC (*3)	-----	FC 50 / FC 60

(*3): AG_LSEND / AG_LREC kann verwendet werden, hat aber keine andere Funktion wie AG_SEND / AG_RECV

Je nach Familie (S7-300, S7-400) sind unterschiedliche Kommunikationsbausteine zu verwenden. Die Kommunikationsbausteine sind in STEP 7 unter verschiedenen Bibliotheken abgelegt.

Kommunikationsbausteine in STEP 7:

Tabelle 36-4

Schnittstelle	STEP 7 Bibliothek
S7-300, CP	SIMATIC_NET_CP / CP300
S7-400, CP	SIMATIC_NET_CP / CP400

36.5 Anwenderschnittstelle: AG_xSEND, AG_xRECV

Für die Namen der Kommunikationsbausteine werden folgende Abkürzungen verwendet:

- AG_xSEND steht für: AG_SEND, AG_LSEND
- AG_xRECV steht für: AG_RECV, AG_LRECV

Bedeutung von x = L

Die Kommunikationsbausteine sind optimiert für die Übertragung umfangreicher Daten (L steht für „long“).

36.5.1 Beschreibung

Der Kommunikationsbaustein AG_xRECV sendet Daten an den Kommunikationsbaustein AG_xRECV.

Die Arbeitsweise der Kommunikationsbausteine ist abhängig vom verwendeten CP (/13/).

AG_xSEND

Der Kommunikationsbaustein übergibt Daten an den CP, die über eine projektierte Verbindung gesendet werden.

AG_xRECV

Der Kommunikationsbaustein übernimmt Daten vom CP, die über eine projektierte Verbindung empfangen wurden.

36.5.2 Parameter für AG_SEND, AG_LSEND

Tabelle 36-5

INPUT	Typ	Erläuterung
ACT	BOOL	Anstoß Sendeauftrag
ID	INT	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
LADDR	WORD	Adresse der Baugruppe (aus Hardware-Konfiguration in STEP 7)
SEND	ANY	Sendebereich (*2)
LEN	INT	Länge der zu sendenden Daten
OUTPUT	Typ	Erläuterung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): "Auftrag fertig":

Unterschiedliche Bedeutung, je nach Typ der FDL-Verbindung:

- Daten liegen auf dem remote CP: spezifiziert
- Daten wurden vom lokalen CP gesendet:
 - Broadcast / Multicast
 - Unspezifiziert und Dienst SDN

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

36.5.3 Parameter für AG_RECV, AG_LRECV

Tabelle 36-6

INPUT	Typ	Erläuterung
ID	INT	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
LADDR	WORD	Adresse der Baugruppe (aus Hardware-Konfiguration in STEP 7)
RECV	ANY	Empfangsbereich (*2)
OUTPUT	Typ	Erläuterung
NDR	BOOL	Daten im Empfangsbereich: ja (NDR=1) / nein (NDR=0) (*1)
LEN	INT	Länge der empfangenen Daten
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): "ja": Die Daten wurden aus dem Empfangspuffer (CP) in den Empfangsbereich (CPU) kopiert.

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

37 PB: FMS-Kommunikation

37.1 Merkmale

Die FMS-Kommunikation zeichnet sich durch folgende Merkmale aus:

- Offener Standard (Kommunikation mit fremden Controllern möglich)
- Daten werden in einer geräteneutralen Form (FMS Variable) übertragen. In den Kommunikationspartnern erfolgt die Konvertierung von FMS Variablen in die gerätespezifische Form, und umgekehrt.
- Zusätzlicher Projektierungsaufwand zur Definition der FMS Variablen
- Kommunikation über CP
- Kommunikation über Verbindungen
- Verbindungen werden projiziert

37.2 Eigenschaften

Erläuterungen zum Aufbau und Inhalt der Tabelle sind in Kapitel 25.2 zu finden.

Tabelle Kommunikationsarten-Detail:

Tabelle 37-1

Kommunikationsart:		FMS-Kommunikation	
Protokoll:		FMS	
Allgemeines			
Medien		PB	
Schnittstellen		CP	
Anbindung	SIMATIC S5	ja	
	Fremd (offene Standards)	ja	
Anwenderschnittstelle			
Kommunikationsbausteine		READ, WRITE	REPORT
maximale Anzahl Daten (*1)		READ <= 237 Byte WRITE <= 233 Byte	<= 233 Byte
Anzahl Variablen bei Aufruf Kommunikationsbaustein		1 Variable	1 Variable
dynamische Adressierung Daten		ja	ja
remote Quittierung		Applikation	nein
Modell		Client / Server	Client / Server

Kommunikationsart:		FMS-Kommunikation
Protokoll:		FMS
Protokoll		
dynamische Datenlänge		ja
Multicast / Broadcast		nein / ja
Verbindungen	zum remote Partner?	ja
	dynamisch / statisch	statisch
Routingfähig		nein
Anwendertätigkeit (*3)		
Kommunikationsbeziehung	Festlegen mit	NetPro
Verbindungen	proj. mit NetPro?	ja
	prog. in STEP 7?	nein
Datenübertragung	Kommunikationsbaustein in STEP 7?	ja

[Zurück zum Sprungverteiler PB](#)

Erläuterungen zur Tabelle:

(*1): Darin enthalten sind Informationen, welche die Daten beschreiben (FMS Variable). Nähere Angaben: siehe /12/

(*3): Zusätzlich: Projektieren von FMS Variablen

37.3 Anwendung

Es folgt eine Übersicht der wichtigsten Tätigkeiten, um eine CPU-CPU Kommunikation zu realisieren.

Tabelle 37-2

Tätigkeit		Engineering Tool
Vernetzung und Adressvergabe		STEP 7, HW Konfig
Projektieren	Anlegen der Kommunikationsobjekte	STEP 7, DB-Editor
FMS Variable	Spezifizieren der Kommunikationsobjekte	STEP 7, Symbolik Editor
Projektierung der Verbindungen: <ul style="list-style-type: none"> • Auswahl der Baugruppen, die kommunizieren sollen • Auswahl Verbindungstyp • Parametrierung Verbindung und Dienst 		STEP 7, NetPro
Programmierung Datenaustausch: Aufruf Kommunikationsbausteine im STEP 7 Anwenderprogramm		STEP 7, Sprachen Editor

37.4 Überblick Anwenderschnittstellen

Überblick: Kommunikationsbausteine

Tabelle 37-3

Kommunikationsbaustein	S7-300	S7-400
	CP	CP
READ	FB 3	FB 3
WRITE	FB 6	FB 6
REPORT	FB 4	FB 4

Je nach Familie (S7-300, S7-400) sind unterschiedliche Kommunikationsbausteine zu verwenden. Die Kommunikationsbausteine sind in STEP 7 unter verschiedenen Bibliotheken abgelegt.

Kommunikationsbausteine in STEP 7:

Tabelle 37-4

Schnittstelle	STEP 7 Bibliothek
S7-300, CP	SIMATIC_NET_CP / CP300
S7-400, CP	SIMATIC_NET_CP / CP400

37.5 Anwenderschnittstelle: READ

37.5.1 Beschreibung

Mit dem Kommunikationsbaustein werden Daten aus dem remote Kommunikationspartner gelesen.

Die Strukturbeschreibung der FMS Variablen liegt im remote Kommunikationspartner (FMS Server). Beim Aufbau der FMS-Verbindung liest der lokale Kommunikationspartner die Strukturbeschreibung vom remote Kommunikationspartner. Damit kann der lokale Kommunikationspartner die Daten entsprechend konvertieren.

READ

Variable lesen

37.5.2 Parameter

Tabelle 37-5

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Leseauftrag
ID	DWORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
VAR_1	ANY	Variable, die remote gelesen werden soll
RD_1	ANY	Zielbereich (*2)
OUTPUT	Typ	Bemerkung
NDR	BOOL	Daten im Zielbereich (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): remote Quittierung: Applikation

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

37.6 Anwenderschnittstelle WRITE

37.6.1 Beschreibung

Mit dem Kommunikationsbaustein werden Daten in den remote Kommunikationspartner geschrieben.

Die Strukturbeschreibung der FMS Variablen liegt im remote Kommunikationspartner (FMS Server). Beim Aufbau der FMS-Verbindung liest der lokale Kommunikationspartner die Strukturbeschreibung vom remote Kommunikationspartner. Damit kann der lokale Kommunikationspartner die Daten entsprechend konvertieren.

WRITE

Variable schreiben

37.6.2 Parameter

Tabelle 37-6

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Schreibauftrag
ID	DWORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
VAR_1	ANY	Variable, die remote geschrieben werden soll
SD_1	ANY	Quellbereich (*2)
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): remote Quittierung: Applikation

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

37.7 Anwenderschnittstelle REPORT

37.7.1 Beschreibung

Der Kommunikationsbaustein ermöglicht einem FMS Server die unquittierte Übertragung von Variablen an einen FMS Client. Der Kommunikationsbaustein wird auch zur Broadcast Übertragung auf FMS Verbindungen genutzt.

REPORT

Variable melden

37.7.2 Parameter

Tabelle 37-7

INPUT	Datentyp	Bemerkung
REQ	BOOL	Anstoß Meldeauftrag
ID	DWORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
SD_1	ANY	Quellbereich lokal (*2)
VAR_1	ANY	Variable, die remote geschrieben werden soll (*2)
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): remote Quittierung: Keine

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

38 PB: DP-Kommunikation

38.1 Merkmale

Vorbemerkung

Die DP-Kommunikation ist ein Sonderfall der CPU-CPU Kommunikation.

Hier werden die Kommunikationsmechanismen der „dezentralen Peripherie“ für eine CPU-CPU Kommunikation verwendet:

- Eine CPU steckt in der zentralen Station
- Die andere CPU steckt in der dezentralen Station

Siehe dazu das Funktionsmodell: 4.4

Merkmale

Die DP-Kommunikation zeichnet sich durch folgende Merkmale aus:

- zyklischer Austausch von Daten zwischen DP-Master und DP-Slave über das Protokoll PROFIBUS DP:
 - DP-Master liest Daten aus DP-Slaves (Eingangsbereich)
 - DP-Master schreibt Daten in DP-Slaves (Ausgangsbereich)
- Der Datenaustausch erfolgt konsistent über die gesamte Länge (systembedingte Datenkonsistenz)

38.2 Eigenschaften

Erläuterungen zum Aufbau und Inhalt der Tabelle sind in Kapitel 25.2 zu finden.

Tabelle Kommunikationsarten-Detail:

Tabelle 38-1

Kommunikationsart:		DP-Kommunikation	
Protokoll:		DP	
Allgemeines			
Medien		PB	
Schnittstellen		CPU, CP	
Anbindung	SIMATIC S5	ja	
	Fremd (offene Standards)	ja	
Anwenderschnittstelle			
Kommunikationsbausteine		Ladebefehle / Transferbefehle	S7-CPU und S7-400 CP
		DPRD_DAT, DPWR_DAT	
		DP_SEND, DP_RECV	S7-300 CP
maximale Anzahl Daten (*1)		Ladebefehl / Transferbefehl:	1, 2, 4 Byte
		DPRD_DAT, DPWR_DAT:	<= 128 Byte
		DP_SEND, DP_REC:	DP-Master: <= 4 kByte DP-Slave: <=244Byte
Anzahl Variablen bei Aufruf Kommunikationsbaustein		1 Variable	
dynamische Adressierung Daten		nein	
remote Quittierung		Applikation	
Modell		Master / Slave	

Kommunikationsart:		DP-Kommunikation
Protokoll:		DP
Protokoll		
dynamische Datenlänge		nein
Multicast / Broadcast		nein
Verbindungen	zum remote Partner?	nein
	dynamisch / statisch	---
Routingfähig		nein
Anwendertätigkeit		
Kommunikationsbeziehung	Festlegen mit	HW Konfiguration
Verbindungen	proj. mit NetPro?	---
	prog. in STEP 7?	---
Datenübertragung	Kommunikationsbaustein in STEP 7?	ja (*2)

[Zurück zum Sprungverteiler PB](#)

Erläuterungen zur Tabelle

(*1): Die maximale Anzahl Daten kann je nach Typ CPU, CP unterschiedlich sein.

Pro DP-Slave gilt generell:

- Eingangsbereich: <= 244 Byte
- Ausgangsbereich: <= 244 Byte

(*2): Für eine CPU-CPU Kommunikation sind Kommunikationsbausteine nur in einer CPU erforderlich (nicht in beiden CPUs)

38.3 Anwendung

Es folgt eine Übersicht der wichtigsten Tätigkeiten, um eine CPU-CPU Kommunikation zu realisieren.

Tabelle 38-2

Tätigkeit	Engineering Tool
Projektierung der Hardware der Station: DP Master	STEP 7, HW Konfig
Projektierung der Hardware der Station: DP Slave	
Programmierung Datenaustausch	STEP 7, Sprachen Editor

38.4 Überblick Anwenderschnittstellen

Fallunterscheidung

S7-CPUs und S7-400 CPs:

- Ladebefehle und Transferbefehle
- Kommunikationsbausteine DPRD_DAT, DPWR_DAT

Nur bei S7-300 CP:

- Kommunikationsbausteine DP_SEND, DP_RECV

Kommunikationsbausteine

Überblick Kommunikationsbausteine:

Tabelle 38-3

Kommunikationsbaustein	S7-300		S7-400
	CPU	CP	CPU, CP
DP_SEND	---	FC 1	---
DP_RECV	---	FC 2	---
DPRD_DAT	SFC 14	---	SFC 14
DPWR_DAT	SFC 15	---	SFC 15

Kommunikationsbausteine in STEP 7:

Tabelle 38-4

Schnittstelle		STEP 7 Bibliothek
S7-300	CPU	Standard Library / System Function Blocks
S7-400	CPU, CP	
S7-300	CP	SIMATIC_NET_CP / CP300

38.5 Anwenderschnittstelle DP_SEND, DP_RECV

38.5.1 Beschreibung

Die Kommunikationsbausteine dienen der Kommunikation zwischen lokaler CPU und lokalem CP (*1). Der CP ist DP Master oder DP Slave.

DP_SEND

Der Kommunikationsbaustein übergibt Daten an den CP.

Fallunterscheidung:

CP ist DP-Master:

Die Daten, welche mit DP_SEND an den CP übergeben werden, werden vom CP zyklisch in die DP-Slaves geschrieben.

CP ist DP-Slave:

Die Daten, welche mit DP_SEND an den CP übergeben werden, werden von einem DP-Master zyklisch aus dem CP gelesen.

DP_RECV

Der Kommunikationsbaustein übernimmt Daten vom CP.

Fallunterscheidung:

CP ist DP-Master:

Die Daten, welche mit DP_RECV vom CP übernommen werden, wurden vom CP zyklisch aus den DP-Slaves gelesen.

CP ist DP-Slave:

Die Daten, welche mit DP_RECV vom CP übernommen werden, wurden von einem DP-Master zyklisch in den CP geschrieben.

38.5.2 Parameter für DP_SEND

Tabelle 38-5

INPUT	Typ	Bemerkung
CPLADDR	WORD	projektierte Anfangsadresse des CP
SEND	ANY	Sendebereich (*2)
OUTPUT	Typ	Bemerkung
DONE	BOOL	Anzeige, ob Auftrag fehlerfrei ausgeführt wurde (*1)
ERROR	BOOL	Fehleranzeige
STATUS	WORD	Statusanzeige

(*1): remote Quittierung: Applikation

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

38.5.3 Parameter für DP_RECV

Tabelle 38-6

INPUT	Typ	Bemerkung
CPLADDR	WORD	projektierte Anfangsadresse des CP
RECV	ANY	Empfangsbereich (*2)
OUTPUT	Typ	Bemerkung
NDR	BOOL	Anzeige, ob Auftrag fehlerfrei ausgeführt wurde (*1)
ERROR	BOOL	Fehleranzeige
STATUS	WORD	Statusanzeige
DPSTATUS	BYTE	Statusanzeige

(*1): remote Quittierung: Applikation

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

38.6 Anwenderschnittstelle DPRD_DAT, DPWR_DAT

38.6.1 Beschreibung

Mit den Kommunikationsbausteinen hat ein DP-Master Zugriff auf die Daten eines DP-Slaves.

DPRD_DAT

Konsistente Daten eines DP-Normslaves lesen

DPWR_DAT

Konsistente Daten in DP-Normslave schreiben

38.6.2 Parameter für DPRD_DAT

Tabelle 38-7

INPUT	Typ	Bemerkung
LADDR	WORD	Projektierte Anfangsadresse im Eingangsbereich, aus dem gelesen werden soll
OUTPUT	Typ	Bemerkung
RET_VAL	INT	Anzeige, ob Auftrag fehlerfrei ausgeführt wurde (*1)
RECORD	ANY	Zielbereich (*2)

(*1): remote Quittierung: Applikation

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

38.6.3 Parameter für DPWR_DAT

Tabelle 38-8

INPUT	Typ	Bemerkung
LADDR	WORD	Projektierte Anfangsadresse im Ausgangsbereich, auf den geschrieben werden soll
RECORD	ANY	Quellbereich (*2)
OUTPUT	Typ	Bemerkung
RET_VAL	INT	Anzeige, ob Auftrag fehlerfrei ausgeführt wurde (*1)

(*1): remote Quittierung: Applikation

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D

39 Serielle Schnittstelle

39.1 Merkmale

Die Kommunikation über eine Serielle Schnittstelle zeichnet sich durch folgende Merkmale aus:

- Einfache Möglichkeit einer CPU-CPU Kopplung mit zwei Teilnehmern (Punkt-zu-Punkt Kopplung)
- Mehrpunkt Kopplungen sind möglich (bei RS 422/485)

39.2 Übersicht

Über die Serielle Schnittstelle sind folgende Kommunikationsarten möglich:

Tabelle 39-1

Kommunikationsart	Kapitel
ASCII, 3964(R), RK 512	40
Anwenderdefiniertes Protokoll	41

Die Kommunikationsarten werden im Folgenden beschrieben.

Hinweis

Die Kommunikation mit MODICON Geräten über die Serielle Schnittstelle, wird im Teil 4 der Dokumentation beschrieben:

Tabelle 39-2

Kommunikationsart	Kapitel
Modbus Seriell (RTU Format)	45

40 ASCII und 3964(R) und RK 512

40.1 Merkmale

40.1.1 Abgrenzung

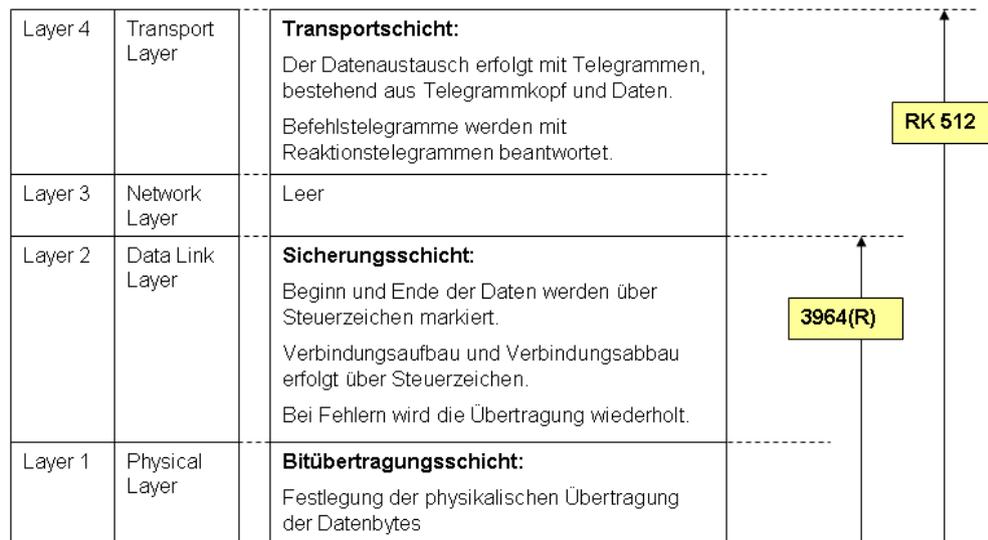
Die Kommunikationsarten lassen sich am Besten mit Hilfe des ISO/OSI-Referenzmodells abgrenzen:

Tabelle 40-1

Kommunikationsart	ISO/OSI-Referenzmodell	Übertragungssicherheit im Vergleich
ASCII	nutzt nur Schicht 1	---
3964(R)	nutzt Schicht 1 und 2	höher wie bei ASCII
RK 512	nutzt Schicht 1, 2 und 4 Schicht 1 und 2 entsprechen 3964(R)	höher wie bei 3964(R)

Das Bild zeigt das ISO/OSI-Referenzmodell:

Abbildung 40-1



40.1.2 Merkmale ASCII

Funktionalität

Der Empfänger erkennt das Ende der Datenübertragung („Ende der Daten“) über ein parametrierbares Endekriterium (Ablauf Zeichenverzugszeit, Empfang Endezeichen, Empfang feste Anzahl Daten).

Codetransparenz

Ein Protokoll ist codetransparent, wenn in den zu übertragenden Daten alle beliebigen Zeichen (00H bis FFH) vorkommen dürfen.

In folgenden Fällen ist ASCII nicht codetransparent:

- Verwendung der Flusskontrolle
- Verwendung von Endezeichen als Endekriterium

Fehlererkennung

Erkannte Fehler

- Paritätsfehler

Nicht erkannte Fehler:

- Keine Ablage der empfangenen Daten
- Unvollständiger Empfang (fehlende Daten)

Remote Quittierung / Rückmeldung am Kommunikationsbaustein

Der Anwender kann nicht erkennen, ob die gesendeten Daten im Anwenderdatenbereich der remote CPU (Applikation) fehlerfrei angekommen sind.

40.1.3 Merkmale 3964(R)

Funktionalität

- Beim Senden werden den Daten Steuerzeichen hinzu gefügt (Startzeichen, Endezeichen, Blockprüfzeichen).
- Verbindungsaufbau / Verbindungsabbau erfolgt über Steuerzeichen
- Bei Fehlern wird die Übertragung wiederholt.

Codetransparenz

Ein Protokoll ist codetransparent, wenn in den zu übertragenden Daten alle beliebigen Zeichen (00H bis FFH) vorkommen dürfen.

3964(R) ist codetransparent.

Fehlererkennung

Erkannte Fehler:

- Paritätsfehler
- Unvollständiger Empfang (fehlende Daten)
Ausnahme: Byte mit „00H“ (siehe unten)

Nicht erkannte Fehler:

- Byte mit „00H“ ist verloren gegangen.

Remote Quittierung / Rückmeldung am Kommunikationsbaustein

Der Anwender kann nicht erkennen, ob die gesendeten Daten im Anwenderdatenbereich der remote CPU (Applikation) fehlerfrei angekommen sind.

40.1.4 Merkmale RK 512

Funktionalität

Das Protokoll arbeitet mit Telegrammen. Die Telegramme enthalten die Daten und einen automatisch eingefügten Telegrammkopf. Einem Befehlstelegramm (SEND/PUT Telegramm, GET Telegramm) folgt ein Reaktionstelegramm (mit oder ohne Daten).

Telegrammkopf Befehlstelegramm:

- Telegrammkennung (SEND/PUT Telegramm, GET Telegramm)
- Datenziel bei SEND/PUT Auftrag
- Datenquelle bei GET Auftrag
- Länge der zu übertragenden Daten

Telegrammkopf Reaktionstelegramm:

- Telegrammkennung (Reaktionstelegramm, Folgetelegramm)
- Fehlernummer

Ablauf Daten senden/schreiben:

- Kommunikationspartner sendet Befehlstelegramm (SEND/PUT) mit Daten
- Kommunikationspartner antwortet mit Reaktionstelegramm ohne Daten

Ablauf Daten holen:

- Kommunikationspartner sendet Befehlstelegramm (GET) ohne Daten
- Kommunikationspartner antwortet mit Reaktionstelegramm mit Daten

Fehlererkennung

Erkannte Fehler:

- Paritätsfehler
- Unvollständiger Empfang (fehlende Daten)

Remote Quittierung / Rückmeldung am Kommunikationsbaustein

Der Anwender kann erkennen, ob die gesendeten Daten im Anwenderdatenbereich der remote CPU (Applikation) fehlerfrei angekommen sind, bzw. ob die Daten aus dem Anwenderdatenbereich abgeholt wurden.

40.2 Anwendung

Die Seriellen Schnittstellen ähneln sich in ihrer Anwendung.

Folgende Tätigkeiten muss der Anwender in STEP 7 vornehmen, um eine CPU-CPU Kommunikation zu realisieren:

- Anlegen S7-Projekt
- Komponenten (CPU, CP) auswählen
- Schnittstelle (Physik) parametrieren
- Protokolle parametrieren
- Anwenderprogramm mit Kommunikationsbausteinen erstellen.

Zusätzlich bei S7-400:

- Verbindung projektieren
(innerhalb einer Station: zwischen S7-400 CPU und CP)

Zusätzlich bei nachladbaren Treibern:

- Treiber installieren

Details zur Anwendung sind zu finden in:

- Handbücher der Komponenten
- STEP 7 Online Hilfe

40.3 Eigenschaften

Erläuterungen zum Aufbau und Inhalt der Tabelle sind in Kapitel 25.2) zu finden.

Tabelle Kommunikationsarten-Detail:

Tabelle 40-2

Kommunikationsart:		ASCII	3964(R)	RK 512
Allgemeines				
Medien		Serielle Schnittstelle	Serielle Schnittstelle	Serielle Schnittstelle
Schnittstellen		CPU, CP	CPU, CP	CPU, CP
Anbindung	SIMATIC S5	ja	ja	ja
	Fremd (offene Standards)	ja	ja	nein
Anwenderschnittstelle				
Kommunikationsbaustein		siehe 40.4	siehe 40.4	siehe 40.11
maximale Anzahl Daten		<= 4096 Byte (siehe 40.4)	<= 4096 Byte (siehe 40.4)	<= 4096 Byte (siehe 40.11)
Anzahl Variablen bei Aufruf Kommunikationsbaustein		1	1	1
dynamische Adressierung Daten		ja, Ausnahme: S7-400 und SFB	ja, Ausnahme: S7-400 und SFB	ja, Ausnahme: S7-400 und SFB
remote Quittierung		nein	Transport	Applikation
Modell		Master / Master	Master / Master	Master / Master

Kommunikationsart:		ASCII	3964(R)	RK 512
Protokoll				
dynamische Datenlänge			ja	
Multicast / Broadcast			---	
Verbindungen	zum remote Partner?		nein (*2)	
	dynamisch / statisch		---	
Routingfähig			---	
Anwendertätigkeit				
Kommunikationsbeziehung.:	Festlegen mit	Hardware Konfig		
Verbindungen	proj. in STEP 7?	nur S7-400: NetPro (lokale Verbindung)		
	prog. in STEP ?	nein		
Datenübertragung	Kommunikationsbaustein in STEP 7?	ja (*1)		

[Zurück zum Sprungverteiler SERIELL](#)

Erläuterungen zur Tabelle:

(*1): Ausnahme: CP441: BSEND / Empfangen ohne BRCV. Dort ist im Empfänger kein Kommunikationsbaustein erforderlich.

(*2): Bei S7-400 wird in NetPro eine „PtP-Verbindung“ projiziert. Dies ist keine Verbindung zum remote Partner.

40.4 Überblick Anwenderschnittstellen ASCII und 3964(R)

Überblick Kommunikationsbausteine und maximale Anzahl Daten (in Klammern).

Tabelle 40-3

Kommunikationsbaustein	ET200S	S7-300			S7-400	
	1SI	CPU	CP 340	CP 341	CP 440	CP 441
S_SEND / S_RCV	FB3 / FB2 (224 Byte)	---	---	---	---	---
SEND_PTP / RCV_PTP	---	SFB60 / SFB61 (1024 Byte)	---	---	---	---
P_SEND / P_RCV	---	---	FB3 / FB2 (1024 Byte)	---	---	---
P_SND_RK / P_RCV_RK	---	---	---	FB8 / FB7 (1024 Byte)	---	---
SEND_440 / REC_440	---	---	---	---	FB10 / FB9 (400 Byte)	---
BSEND / BRCV	---	---	---	---	---	SFB12 / SFB13 (4096 Byte)
BSEND / Empfangsfach	---	---	---	---	---	SFB12 / --- (4096 Byte)

Kommunikationsbausteine in STEP 7:

Tabelle 40-4

Schnittstelle		STEP 7 Bibliothek
ET200S	1SI	ET200sSI / ET200S Serial Interface (*1)
S7-300	CPU	Standard Library / System Function Blocks
	CP 340	CP PtP / CP 340
	CP 341	CP PtP / CP 341
S7-400	CP 440	CP PtP / CP 440
	CP 441	Standard Library / System Function Blocks

(*1): Die Bausteine können aus dem Internet geladen werden: /14/
Nach Installation in angegebener STEP 7 Bibliothek

40.5 ASCII und 3964(R): Anwenderschnittstelle ET 200S

40.5.1 Beschreibung

Der Kommunikationsbaustein S_SEND sendet Daten an den Kommunikationsbaustein S_RCV.

S_SEND

Daten senden

S_RCV

Daten empfangen

40.5.2 Parameter für S_SEND

Tabelle 40-5

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse ET 200S 1SI
DB_NO	INT	Sendebereich (*2)
DBB_NO	INT	
LEN	INT	Länge der zu sendenden Daten (*3)
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
COM_RST	BOOL	Neustart des FB

(*1): remote Quittierung:

- ASCII: keine
- 3964(R): Transport

(*2): SIMATIC S7 Speicherbereiche: D

(*3): maximale Anzahl Daten: = 224 Byte

40.5.3 Parameter für S_RCV

Tabelle 40-6

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse ET 200S 1SI
DB_NO	INT	Empfangsbereich (*2)
DBB_NO	INT	
OUTPUT	Typ	Bemerkung
LEN	INT	Länge der empfangenen Daten (*3)
NDR	BOOL	Daten im Empfangsbereich: ja / nein (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
COM_RST	BOOL	Neustart des FB

(*1): "ja": Daten wurden aus dem Empfangspuffer in den Empfangsbereich kopiert.

(*2): SIMATIC S7 Speicherbereiche: D

(*3): maximale Anzahl Daten: = 224 Byte

40.6 ASCII / 3964(R): Anwenderschnittstelle S7-300 CPU

40.6.1 Beschreibung

Der Kommunikationsbaustein SEND_PTP sendet Daten an den Kommunikationsbaustein RCV_PTP.

SEND_PTP

Daten senden

RCV_PTP

Daten empfangen

40.6.2 Parameter für SEND_PTP

Tabelle 40-7

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Auftrag
LADDR	WORD	E/A Adresse des Submoduls
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformationen
STATUS	WORD	
IN_OUT	Typ	Bemerkung
SD_1	ANY	Sendebereich (*2)
LEN	INT	Länge der zu sendenden Daten (*3)

(*1): remote Quittierung:

- ASCII: keine
- 3964(R): Transport

(*2): SIMATIC S7 Speicherbereiche: D

(*3): maximale Anzahl Daten: = 1024 Byte

40.6.3 Parameter für RCV_PTP

Tabelle 40-8

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
R	BOOL	Abbruch Auftrag
LADDR	WORD	E/A Adresse des Submoduls
OUTPUT	Typ	Bemerkung
NDR	BOOL	Daten im Empfangsbereich: ja / nein (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
RD_1	ANY	Empfangsbereich (*2)
LEN	INT	Länge empfangener Daten (*3)

(*1): „ja“: Daten wurden aus dem Empfangspuffer in den Empfangsbereich kopiert.

(*2): SIMATIC S7 Speicherbereiche: D

(*3): maximale Anzahl Daten: = 1024 Byte

40.7 ASCII / 3964(R): Anwenderschnittstelle CP 340

40.7.1 Beschreibung

Der Kommunikationsbaustein P_SEND sendet Daten an den Kommunikationsbaustein P_RCV.

P_SEND

Daten senden

P_RCV

Daten empfangen

40.7.2 Parameter für P_SEND

Tabelle 40-9

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse CP 340
DB_NO	INT	Sendebereich (*2)
DBB_NO	INT	
LEN	INT	Länge der zu sendenden Daten (*3)
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): remote Quittierung:

- ASCII: keine
- 3964(R): Transport

(*2): SIMATIC S7 Speicherbereiche: D

(*3): maximale Anzahl Daten: = 1024 Byte

40.7.3 Parameter für P_RCV

Tabelle 40-10

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse CP 340
DB_NO	INT	Empfangsbereich (*2)
DBB_NO	INT	
OUTPUT	Typ	Bemerkung
LEN	INT	Länge empfangener Daten (*3)
NDR	BOOL	Daten im Empfangsbereich: ja / nein (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): „ja“: Daten wurden aus dem Empfangspuffer in den Empfangsbereich kopiert.

(*2): SIMATIC S7 Speicherbereiche: D

(*3): maximale Anzahl Daten: = 1024 Byte

40.8 ASCII / 3964(R): Anwenderschnittstelle CP 341

40.8.1 Beschreibung

Der Kommunikationsbaustein P_SND_RK sendet Daten an den Kommunikationsbaustein P_RCV_RK.

P_SND_RK

Daten senden

P_RCV_RK

Daten empfangen

Hinweis

Die Kommunikationsbausteine (P_SND_RK, P_RCV_RK) werden universell eingesetzt für ASCII, 3964(R), und RK 512. Unten aufgeführt sind nur die Parameter, die für ASCII und 3964(R) relevant sind.

40.8.2 Parameter für P_SND_RK

Tabelle 40-11

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse CP 341
DB_NO	INT	Sendebereich (*2)
DBB_NO	INT	
LEN	INT	Länge der zu sendenden Daten (*3)
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): remote Quittierung:

- ASCII: keine
- 3964(R): Transport

(*2): SIMATIC S7 Speicherbereiche: D

(*3): maximale Anzahl Daten: = 1024 Byte

40.8.3 Parameter für P_RCV_RK

Tabelle 40-12

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse CP 341
DB_NO	INT	Empfangsbereich (*2)
DBB_NO	INT	
OUTPUT	Typ	Bemerkung
LEN	INT	Länge der empfangenen Daten (*3)
NDR	BOOL	Daten im Empfangsbereich: ja / nein (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): „ja“: Daten wurden aus dem Empfangspuffer in den Empfangsbereich kopiert.

(*2): SIMATIC S7 Speicherbereiche: D

(*3): maximale Anzahl Daten: = 1024 Byte

40.9 ASCII / 3964(R): Anwenderschnittstelle CP 440

40.9.1 Beschreibung

Der Kommunikationsbaustein SEND_440 sendet Daten an den Kommunikationsbaustein REC_440.

SEND_440

Daten senden

REC_440

Daten empfangen

40.9.2 Parameter für SEND_440

Tabelle 40-13

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse CP 440
DB_NO	INT	Sendebereich (*2)
DBB_NO	INT	
LEN	INT	Länge der zu sendenden Daten (*3)
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): remote Quittierung:

- ASCII: keine
- 3964(R): Transport

(*2): SIMATIC S7 Speicherbereiche: D

(*3): maximale Anzahl Daten: = 400 Byte

40.9.3 Parameter für REC_440

Tabelle 40-14

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse CP 440
DB_NO	INT	Empfangsbereich (*2)
DBB_NO	INT	
OUTPUT	Typ	Bemerkung
LEN	INT	Länge empfangener Daten (*3)
NDR	BOOL	Daten im Empfangsbereich: ja / nein (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): „ja“: Daten wurden aus dem Empfangspuffer in den Empfangsbereich kopiert.

(*2): SIMATIC S7 Speicherbereiche: D

(*3): maximale Anzahl Daten: = 400 Byte

40.10 ASCII / 3964(R): Anwenderschnittstelle CP 441

40.10.1 Beschreibung

Zwei Möglichkeiten stehen zur Verfügung:

- Senden mit BSEND / Empfangen mit BRCV
- Senden mit BSEND / Empfangen mit Empfangsfach

In beiden Fällen wird der Empfangsbereich im Empfänger festgelegt. D.h. der Empfangsbereich wird beim Senden nicht übermittelt.

Senden mit BSEND / Empfangen mit BRCV

Im Sender und Empfänger sind Kommunikationsbausteine erforderlich.

Vorteil durch Verwendung von BRCV im Empfänger:

- Die Applikation (Anwenderprogramm) erkennt den vollständigen Empfang der Daten
- Verhindern des Überschreibens von Daten im Empfangspuffer, die von der Applikation (Anwenderprogramm) noch nicht abgeholt wurden.

Senden mit BSEND / Empfangen mit Empfangsfach

Im Empfänger ist kein Kommunikationsbaustein erforderlich. Das Empfangsfach (Datenbaustein) wird im Empfänger parametrierbar.

Nachteil durch Verwendung eines Empfangsfaches im Empfänger:

- Die Applikation (Anwenderprogramm) kann nicht erkennen, wann eine Datenübertragung stattfindet.

40.10.2 Parameter für BSEND

Tabelle 40-15

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Auftrag
ID	WORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
R_ID	DWORD	Hier ohne Bedeutung
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
SD_1	ANY	Sendebereich (*2)
LEN	WORD	Länge der zu sendenden Daten (*3)

(*1): remote Quittierung:

- ASCII: keine
- 3964(R): Transport

(*2): SIMATIC S7 Speicherbereiche: E, A, M, D, T, Z

(*3): maximale Anzahl Daten: = 4096 Byte

40.10.3 Parameter für BRCV

Tabelle 40-16

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
ID	WORD	Referenz auf die lokale Verbindungsbeschreibung (wird durch Verbindungsprojektierung in STEP 7 vorgegeben)
R_ID	DWORD	Hier ohne Bedeutung
OUTPUT	Typ	Bemerkung
ANDR	BOOL	Daten im Empfangsbereich: ja / nein (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
RD_1	ANY	Empfangsbereich (*2)
LEN	WORD	Länge der empfangenen Daten (*3)

(*1): „ja“: Daten wurden aus dem Empfangspuffer in den Empfangsbereich kopiert.

(*2): SIMATIC S7 Speicherbereiche: D

(*3): maximale Anzahl Daten: = 4096 Byte

40.11 Überblick Anwenderschnittstellen RK 512

Überblick Kommunikationsbausteine und maximale Anzahl Daten (in Klammern).

Tabelle 40-17

Kommunikationsbaustein	S7-300		S7-400
	CPU	CP 341	CP 441
SEND_RK / SERVE_RK	SFB 63 / SFB 65 (1024 Byte)	---	---
FETCH_RK / SERVE_RK	SFB 64 / SFB 65 (1024 Byte)	---	---
P_SND_RK / P_RCV_RK	---	FB 8 / FB 7 (4096 Byte)	---
BSEND / BRCV	---	---	SFB 12 / SFB 13 (4096 Byte)
BSEND / ---	---	---	SFB 12 / --- (4096 Byte)
PUT	---	---	SFB 15 / --- (≤ 450 Byte (*1))
GET	---	---	SFB 14 / --- (≤ 450 Byte (*1))

(*1): maximale Anzahl Daten ist abhängig von verwendeter CPU

Die Kommunikationsbausteine sind zu finden in:

Tabelle 40-18

Schnittstelle		STEP 7 Bibliothek
S7-300	CPU	Standard Library / System Function Blocks
	CP 341	CP PtP / CP 341
S7-400	CP 441	Standard Library / System Function Blocks

40.12 RK 512: Anwenderschnittstelle S7-300 CPU

40.12.1 Beschreibung

Fallunterscheidung:

- Daten senden mit SEND_RK / SERVE_RK
- Daten holen mit FETCH_RK / SERVE_RK

Die beiden Kommunikationspartner werden im Folgenden bezeichnet mit:

- CPU_1
- CPU_2

40.12.2 Daten senden

CPU_1 sendet Daten an CPU_2.

CPU_1 bestimmt, wo die Daten in CPU_2 abgelegt werden.

Kommunikationsbausteine:

- CPU_1: SEND_RK
- CPU_2: SERVE_RK

SEND_RK

Daten senden, mit Angabe des Empfangsbereiches

SERVE_RK

Daten empfangen

40.12.3 Daten holen

CPU_1 holt Daten aus CPU_2.

CPU_1 bestimmt, welche Daten aus CPU_2 geholt werden.

Kommunikationsbausteine:

- CPU_1: FETCH_RK
- CPU_2: SERVE_RK

FETCH_RK

Daten holen, mit Angabe des Quellbereiches

SERVE_RK

Daten bereitstellen

40.12.4 Parameter für SEND_RK

Kommunikationsbaustein in remote CPU: SERVE_RK

Tabelle 40-19

INPUT	Typ	Bemerkung
SYNC_DB	INT	Datenbaustein zur Synchronisierung
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Auftrag
LADDR	WORD	E/A Adresse des Submoduls
R_CPU	INT	Nummer der remote CPU
R_TYPE	CHAR	Zielbereich in remote CPU (*2)
R_DBNO	INT	
R_OFFSET	INT	
R_CF_BYT	INT	Koppelmerker auf remote CPU
R_CF_BIT	INT	
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
SD_1	ANY	Quellbereich in lokaler CPU (*3)
LEN	INT	Länge der zu sendenden Daten (*4)

(*1): remote Quittierung: Applikation

(*2): SIMATIC Speicherbereich: Datenbaustein (D, X)

(*3): SIMATIC Speicherbereich: Datenbaustein (D)

(*4): maximale Anzahl Daten: 1024 Byte

40.12.5 Parameter für SERVE_RK

Kommunikationsbaustein in remote CPU: SEND_RK

Tabelle 40-20

INPUT	Typ	Bemerkung
SYNC_DB	INT	Datenbaustein zur Synchronisierung
EN_R	BOOL	Anstoß Daten empfangen
R	BOOL	Abbruch Auftrag
LADDR	WORD	E/A Adresse des Submoduls
OUTPUT	Typ	Bemerkung
NDR	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
L_TYPE	CHAR	Zielbereich auf lokaler CPU (*2)
L_DBNO	INT	
L_OFFSET	INT	
L_CF_BYT	INT	Koppelmerker
L_CF_BIT	INT	
IN_OUT	Typ	Bemerkung
LEN	INT	Länge der empfangenen Daten (*3)

(*1): „Auftrag fertig“: Daten wurden empfangen

(*2): SIMATIC Speicherbereich: Datenbaustein (D).

(*3): maximale Anzahl Daten: = 1024 Byte

40.12.6 Parameter für FETCH_RK

Kommunikationsbaustein in remote CPU: SERVE_RK

Tabelle 40-21

INPUT	Typ	Bemerkung
SYNC_DB	INT	Datenbaustein zur Synchronisierung
REQ	BOOL	Anstoß Daten holen
R	BOOL	Abbruch Auftrag
LADDR	EORD	E/A Adresse des Submoduls
R_CPU	INT	Nummer der remote CPU
R_TYPE	CHAR	Quellbereich in remote CPU (*2)
R_DBNO	INT	
R_OFFSET	INT	
R_CF_BYT	INT	Koppelmerker auf remote CPU
R_CF_BIT	INT	
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
RD_1	ANY	Zielbereich in lokaler CPU (*3)
LEN	INT	Länge der zu empfangenen Daten (*4)

(*1): remote Quittierung: Applikation

(*2): SIMATIC Speicherbereich: Datenbaustein (D, X), M, E, A, T, Z

(*3): SIMATIC Speicherbereich: Datenbaustein (D)

(*4): maximale Anzahl Daten: = 1024 Byte

40.12.7 Parameter für SERVE_RK

Kommunikationsbaustein in remote CPU: FETCH_RK

Tabelle 40-22

INPUT	Typ	Bemerkung
SYNC_DB	INT	Datenbaustein zur Synchronisierung
EN_R	BOOL	Anstoß Daten bereitstellen
R	BOOL	Abbruch Auftrag
LADDR	WORD	E/A Adresse des Submoduls
OUTPUT	Typ	Bemerkung
NDR	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
L_TYPE	CHAR	Quellbereich auf lokaler CPU (*2)
L_DBNO	INT	
L_OFFSET	INT	
L_CF_BYT	INT	Koppelmerker
L_CF_BIT	INT	
IN_OUT	Typ	Bemerkung
LEN	INT	Länge der bereitgestellten Daten (*3)

(*1): „Auftrag fertig“: Daten wurden abgeholt.

(*2): SIMATIC Speicherbereich: Datenbaustein (D), M, E, A, Z, T

(*3): maximale Anzahl Daten: = 1024 Byte

40.13 RK 512: Anwenderschnittstelle CP 341

40.13.1 Beschreibung

Fallunterscheidung:

- Daten senden mit P_SND_RK / P_RCV_RK
- Daten holen mit P_SND_RK / P_RCV_RK

Die beiden Kommunikationspartner werden im Folgenden bezeichnet mit:

- CPU_1
- CPU_2

Hinweis

Die Kommunikationsbausteine (P_SND_RK, P_RCV_RK) werden universell eingesetzt für ASCII, 3964(R) und RK 512. Unten aufgeführt sind nur die Parameter, die für RK 512 relevant sind.

40.13.2 Daten senden

CPU_1 sendet Daten an CPU_2.

CPU_1 bestimmt, wo die Daten in CPU_2 abgelegt werden.

Kommunikationsbausteine:

- CPU_1: P_SND_RK
- CPU_2: P_RCV_RK

P_SND_RK

Daten senden, mit Angabe des Empfangsbereiches

P_RCV_RK

Daten empfangen

40.13.3 Daten holen

CPU_1 holt Daten aus CPU_2.

CPU_1 bestimmt, welche Daten aus CPU_2 geholt werden.

Kommunikationsbausteine:

- CPU_1: P_SND_RK
- CPU_2: P_RCV_RK

P_SND_RK

Daten holen, mit Angabe des Quellbereiches

P_RCV_RK

Daten bereitstellen

40.13.4 Parameter für P_SND_RK

Kommunikationsbaustein in remote CPU: P_RCV_RK

Tabelle 40-23

INPUT	Typ	Bemerkung
SF	CHAR	SF = "S" (Parameter für Daten Senden)
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse des CP 341
DB_NO	INT	Quellbereich in lokaler CPU(*3)
DBB_NO	INT	
LEN	INT	Länge der zu sendenden Daten (*4)
R_CPU_NO	INT	Nummer der remote CPU
R_TYP	CHAR	Zielbereich in remote CPU (*2)
R_NO	INT	
R_OFFSET	INT	
R_CF_BYT	INT	
R_CF_BIT	INT	Koppelmerker auf remote CPU
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): remote Quittierung: Applikation

(*2): SIMATIC Speicherbereich: Datenbaustein (D, X)

(*3): SIMATIC Speicherbereich: Datenbaustein (D)

(*4): maximale Anzahl Daten: 4096 Byte

40.13.5 Parameter für P_RCV_RK

Kommunikationsbaustein in remote CPU: P_SND_RK, mit Parameter SF = „S“

Tabelle 40-24

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Daten empfangen
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse CP 341
DB_NO	INT	Zielbereich auf lokaler CPU, wenn Datenziel „DX“ beim Sender parametriert(*2)
DBB_NO	INT	
OUTPUT	Typ	Bemerkung
NDR	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
L_TYP	CHAR	Zielbereich auf lokaler CPU, wenn Datenziel „DB“ beim Sender parametriert(*2)
L_NO	INT	
L_OFFSET	INT	
LEN	INT	Länge der empfangenen Daten (*3)
L_CF_BYT	INT	Koppelmerker
L_CF_BIT	INT	

(*1): „Auftrag fertig“: Daten wurden empfangen

(*2): SIMATIC Speicherbereich: Datenbaustein (D, X)

(*3): maximale Anzahl Daten: = 4096 Byte

40.13.6 Parameter für P_SND_RK

Kommunikationsbaustein in remote CPU: P_RCV_RK

Tabelle 40-25

INPUT	Typ	Bemerkung
SF	CHAR	SF = "F" (Parameter für Daten holen)
REQ	BOOL	Anstoß Daten holen
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse des CP 341
DB_NO	INT	Zielbereich lokale CPU (*3)
DBB_NO	INT	
LEN	INT	Datenlänge (*2)
R_CPU_NO	INT	Nummer der remote CPU
R_TYP	CHAR	Quellbereich remote CPU (*2)
R_NO	INT	
R_OFFSET	INT	
R_CF_BYT	INT	Koppelmerker auf remote CPU
R_CF_BIT	INT	
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): remote Quittierung: Applikation

(*2): SIMATIC Speicherbereich: Datenbaustein (D, X), M, E, A, Z, T

(*3): SIMATIC Speicherbereich: Datenbaustein (D)

(*4): maximale Anzahl Daten: 1024 Byte

40.13.7 Parameter für P_RCV_RK

Kommunikationsbaustein in remote CPU: P_SND_RK, mit Parameter SF = „F“

Tabelle 40-26

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Daten bereitstellen
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse CP 341
DB_NO	INT	ohne Bedeutung
DBB_NO	INT	
OUTPUT	Typ	Bemerkung
NDR	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
L_TYP	CHAR	Quellbereich auf lokaler CPU (*2)
L_NO	INT	
L_OFFSET	INT	
LEN	INT	Datenlänge (*3)
L_CF_BYT	INT	Koppelmerker
L_CF_BIT	INT	

(*1): „Auftrag fertig“: Daten wurden abgeholt

(*2): SIMATIC Speicherbereich: Datenbaustein (D), M, E, A, Z, T

(*3): maximale Anzahl Daten: = 4096 Byte

40.14 RK 512: Anwenderschnittstelle CP 441

40.14.1 Beschreibung

Fallunterscheidung

- Daten senden mit BSEND / BRCV
- Daten senden mit BSEND / ---
- Daten senden mit PUT
- Daten holen mit GET

Die beiden Kommunikationspartner werden im Folgenden bezeichnet mit:

- CPU_1
- CPU_2

40.14.2 Daten senden

Daten aus CPU_1 senden, und in CPU_2 empfangen.

CPU_2 bestimmt, wo die Daten abgelegt werden.

Kommunikationsbausteine:

- CPU_1: BSEND
- CPU_2: BRCV

BSEND

Daten senden

BRCV

Daten empfangen, Angabe des Empfangsbereiches

40.14.3 Daten senden

CPU_1 sendet Daten an CPU_2.

CPU_1 bestimmt, wo die Daten abgelegt werden.

Kommunikationsbausteine:

- CPU_1: BSEND
- CPU_2: ---

BSEND

Daten senden, Angabe des Empfangsbereiches

Hinweis: CPU_2 kann nicht erkennen, wann eine Datenübertragung läuft.

Daten senden PUT / ---

CPU_1 sendet Daten an CPU_2

Kommunikationsbausteine:

- CPU_1: PUT
- CPU_2: ---

PUT: Daten senden, Angabe von maximal vier Empfangsbereichen

Hinweis: CPU_2 kann nicht erkennen, wann eine Datenübertragung läuft.

Daten holen GET / ---

CPU_1 holt Daten aus CPU_2.

Kommunikationsbausteine:

- CPU_1: GET
- CPU_2: ---

GET: Daten holen, Angabe von maximal vier Quellbereichen.

Hinweis: CPU_2 kann nicht erkennen, wann eine Datenübertragung läuft.

40.14.4 Parameter für BSEND

Kommunikationsbaustein in remote CPU: BRCV

Tabelle 40-27

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Auftrag
ID	WORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
R_ID	DWORD	Zuordnung Sende SFB/FB und Empfangs SFB/FB. Dies ermöglicht die Kommunikation mehrerer SFB/FB Paare über dieselbe logische Verbindung.
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformationen
STATUS	WORD	
IN_OUT	Typ	Bemerkung
SD_1	ANY	Quellbereich in lokaler CPU (*2)
LEN	WORD	Länge der zu sendenden Daten (*3)

(*1): remote Quittierung: Applikation

(*2): SIMATIC Speicherbereich: Datenbaustein (D), E, A, M, T, Z

(*3): maximale Anzahl gesendeter Daten: 4096 Byte

40.14.5 Parameter für BRCV

Kommunikationsbaustein in remote CPU: BRCV

Tabelle 40-28

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
ID	WORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
R_ID	DWORD	Zuordnung Sende SFB/FB und Empfangs SFB/FB. Dies ermöglicht die Kommunikation mehrerer SFB/FB Paare über dieselbe logische Verbindung.
OUTPUT	Typ	Bemerkung
NDR	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
RD_1	ANY	Zielbereich in lokaler CPU (*2)
LEN	WORD	Länge der empfangenen Daten (*3)

(*1): „Auftrag fertig“: Daten wurden empfangen

(*2): SIMATIC Speicherbereich: Datenbaustein (D)

(*3): maximale Anzahl Daten: 4096 Byte

40.14.6 Parameter für BSEND

Kommunikationsbaustein in remote CPU: ---

Tabelle 40-29

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Auftrag
ID	WORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
R_ID	DWORD	Zielbereich in remote CPU (*2)
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformationen
STATUS	WORD	
IN_OUT	Typ	Bemerkung
SD_1	ANY	Quellbereich in lokaler CPU (*3)
LEN	WORD	Länge der zu sendenden Daten (*4)

(*1): remote Quittierung: Applikation

(*2): SIMATIC Speicherbereich: Datenbaustein (D)

(*3): SIMATIC Speicherbereich: Datenbaustein (D), E, A, M, T, Z

(*4): maximale Anzahl Daten: <= 450 Byte (abhängig von remote CPU)

40.14.7 Parameter für PUT

Kommunikationsbaustein in remote CPU: ---

Tabelle 40-30

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Schreibauftrag
ID	WORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT		
ADDR_i	ANY	Zielbereiche in der remote CPU (i=1, 2, 3, 4) (*2) (*4)
SD_i	ANY	Quellbereiche in der lokalen CPU (i=1, 2, 3, 4) (*3) (*4)

(*1): remote Quittierung: Applikation

(*2): SIMATIC Speicherbereich: Datenbaustein (D)

(*3): SIMATIC Speicherbereich: Datenbaustein (D), E, A, M, T, Z

(*4): maximale Anzahl Daten: <= 450 Byte (abhängig von remote CPU)

40.14.8 Parameter für GET

Kommunikationsbaustein in remote CPU: ---

Tabelle 40-31

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Leseauftrag
ID	WORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
OUTPUT	Typ	Bemerkung
NDR	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT		
ADDR_i	ANY	Quellbereiche in der remote CPU (i=1, 2, 3, 4) (*2) (*4)
RD_i	ANY	Zielbereiche in der lokalen CPU (i=1, 2, 3, 4) (*3) (*4)

(*1): remote Quittierung: Applikation

(*2): SIMATIC Speicherbereich: Datenbaustein (D), E, A, M, T, Z

(*3): SIMATIC Speicherbereich: Datenbaustein (D)

(*4): maximale Anzahl Daten: <= 450 Byte (abhängig von remote CPU)

41 Anwenderdefiniertes Protokoll

41.1 Merkmale

Hinweis

Die Kommunikationsart ist bei S7-1200 realisiert. Deswegen wird im Folgenden als Engineering Tool „STEP 7 Basic“ genannt.

Das Protokoll wird im STEP 7 Basic Anwenderprogramm realisiert. Damit wird eine hohe Flexibilität in der Anwendung erreicht.

STEP 7 Basic bietet Bibliotheken mit Operationen, die für die Programmierung des anwenderdefinierten Protokolls genutzt werden können.

41.2 Anwendung

Folgende Tätigkeiten muss der Anwender in STEP 7 Basic vornehmen, um eine CPU-CPU Kommunikation zu realisieren:

- Geräte konfigurieren
- Schnittstellen der Kommunikationsmodule (CM) konfigurieren:
 - in STEP 7 Basic konfigurieren
 - programmieren mit speziellen Operationen
- Sendeparameter und Empfangsparameter in STEP 7 Basic konfigurieren
- Programmierung der Kommunikation

Details zur Anwendung sind zu finden in:

- Handbücher der Komponenten
- STEP 7 Online Hilfe

41.3 Anwenderschnittstelle S7-1200

41.3.1 Beschreibung

Der Kommunikationsbaustein SEND_PTP sendet Daten an den Kommunikationsbaustein RCV_PTP.

SEND_PTP

Daten senden

RCV_PTP

Daten empfangen

41.3.2 Parameter für SEND_PTP

Tabelle 41-1

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
PORT	PORT	Kennung Kommunikationsport
BUFFER	VARIANTE	Sendebereich (*2)
LENGTH	UINT	Länge der zu sendenden Daten (*3)
PTRCL	BOOL	---
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): „Auftrag fertig“: Die Daten wurden an das lokale CM übergeben

(*2): Puffer der lokalen CPU

(*3): maximale Anzahl Daten: =1024 Byte

41.3.3 Parameter für RCV_PTP

Tabelle 41-2

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
PORT	PORT	Kennung Kommunikationsport
BUFFER	VARIANTE	Empfangsbereich (*2)
OUTPUT	Typ	Bemerkung
NDR	BOOL	Auftrag läuft / Auftrag fertig (*1)
LENGTH	UINT	Länge empfangener Daten (*3)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): „Auftrag fertig“: Die Daten wurden vom lokalen CM abgeholt, und in den Puffer der CPU geschrieben.

(*2): Puffer der lokalen CPU

(*3): maximale Anzahl Daten: =1024 Byte

42 Informationen Teil 3

In den Tabellen sind Verweise auf Informationen zu den Themen aus Teil 3 zu finden.

Alle Verweise /x/ sind zentral im Kapitel 47 hinterlegt. Dort sind auch die entsprechenden Links ins Internet zu finden.

Tabelle 42-1

Verweis	Titel / Inhalt	Informationen zu
---	STEP 7 Online Hilfe	Kommunikationsarten, Kommunikationsbausteine
/6/	System- und Standardfunktionen für S7-300/400	Kommunikationsarten, Kommunikationsbausteine
/13/	SIMATIC NET Funktionen (FC) und Funktionsbausteine (FB) für SIMATIC NET S7-CPs	Kommunikationsarten, Kommunikationsbausteine
/100/	FAQs für S7-300 CPUs	Kommunikation projektieren und programmieren: Kommunikations-Bausteine verwenden
/101/	FAQs für S7-400 CPUs	
/102/	FAQs für Industrial Ethernet S7-300/400CPs	
/103/	FAQs für PROFIBUS S7-300/400CPs	
/105/	FAQs für IE S7-300/400 CPs	Kommunikation projektieren und programmieren: Verbindungen projektieren
/106/	FAQs für PROFIBUS S7-300/400 CPs	

TEIL 4: Kommunikation mit fremden Controllern ohne Verwendung eines offenen Standards

Teil 4 beschreibt die Kommunikation zu fremden Controllern,
ohne Verwendung eines offenen Standards

TEIL 4: Gliederung und Inhalt

Tabelle 42-2

Kapitel	Gliederung	Inhalt
44	Modbus/TCP	Kopplung über Netz PN/IE: <ul style="list-style-type: none">• Merkmale• Anwenderschnittstellen
45	Modbus Seriell (RTU Format)	Kopplung über Serielle Schnittstelle: <ul style="list-style-type: none">• Merkmale• Anwenderschnittstellen
46	Informationen	Gerätehandbücher, FAQs, Applikationen,

43 Vorbemerkungen

Offene Standards

Ein SIMATIC Controller kann über offene Standards (48) mit Controllern fremder Hersteller kommunizieren, wenn die fremden Controller ebenfalls offene Standards implementiert haben.

Dies ist im Teil 3 der Dokumentation bei den Tabellen Eigenschaften berücksichtigt. Dort gibt es das Kriterium „Anbindung Fremd“. „Ja“ bedeutet, dass es sich bei der Kommunikationsart um einen offenen Standard handelt. Somit kann mit fremden Controllern kommuniziert werden, welche diesen offenen Standard ebenfalls implementiert haben.

Beispiele:

- Offene-Kommunikation über T-Bausteine
- Offene-Kommunikation über Send/Receive-Bausteine

Offengelegte Protokolle

Hier, im Teil 4 der Dokumentation wird die Kommunikation über offengelegte Protokolle beschrieben.

Eigenschaften eines offengelegten Protokolls:

- Das Protokoll ist vom Hersteller offengelegt
- Das Protokoll ist herstellerspezifisch.
- Jeder darf das Protokoll verwenden.
- Das Protokoll ist nicht standardisiert (nicht international genormt)

44 Modbus/TCP

44.1 Merkmale

Modbus ist ein weltweit verbreitetes Protokoll, welches allen Anwendern offen gelegt ist. Modbus/TCP ermöglicht die Kommunikation über TCP/IP Netze.

44.2 Überblick Anwenderschnittstellen

44.2.1 Allgemeines

Ein SIMATIC Controller kann sein:

- Modbus Server
- Modbus Client

SIMATIC ohne S7-1200

Für die Modbus/TCP Kommunikation gibt es eigene Funktionsbausteine (Modbus Bausteine). Die Modbus Bausteine sind nicht im Lieferumfang der Programmiersoftware STEP 7 enthalten..

Informationen zur Funktionalität und zur Bestellung: /11/

Maximale Anzahl Daten

Mit einem Modbus TCP Auftrag maximal übertragbare Daten:

Tabelle 44-1

Auftrag	Übertragung bitweise	Übertragung wortweise
Lesender Auftrag	250 Byte	250 Byte
Schreibender Auftrag	100 Byte	200 Byte

Kommunikationsschnittstelle

SIMATIC Controller können über CPU oder CP mit Modbus Controllern kommunizieren.

44.2.2 Anschluss über SIMATIC CPU

Schnittstelle und Anwenderschnittstelle:

Tabelle 44-2

Schnittstelle		Anwenderschnittstelle
S7-300	integrierter PN/IE Schnittstelle der CPU	Modbus Bausteine für CPU
S7-400	integrierter PN/IE Schnittstelle der CPU	
ET 200 CPU	integrierter PN/IE Schnittstelle der CPU	
WinAC RTX	integrierter PN/IE Schnittstelle des PC	
S7-1200	integrierter PN/IE Schnittstelle der CPU	Globale Bibliotheksoperationen in STEP 7 Basic

44.2.3 Anschluss über SIMATIC CP

Schnittstelle und Anwenderschnittstelle:

Tabelle 44-3

Schnittstelle		Anwenderschnittstelle
S7-300	CP 343-1	Modbus Bausteine für CP
S7-400	CP 443-1	

44.2.4 Einrichten der TCP-Verbindung

Es stehen zwei Möglichkeiten zur Verfügung:

- Programmieren der Verbindung mit T-Bausteinen (TCON, TDISCON)
- Projektieren der Verbindung mit „Modbus TCP Wizard“ (/30/)

45 Modbus Seriell (RTU Format)

45.1 Merkmale

Merkmale

Modbus ist ein weltweit verbreitetes Protokoll, welches allen Anwendern offen gelegt ist. Modbus Seriell ermöglicht die Kommunikation über serielle Schnittstellen (RS232C, RS 422/485).

Für Modbus Seriell gibt es zwei Varianten:

- RTU Format: binäre Kodierung der Daten
- ASCII Format: ASCII Kodierung der Daten

Im Folgenden wird die Variante mit RTU Format beschrieben:

- Modbus Seriell (RTU Format)

Modbus Master / Modbus Slave

Modbus Seriell funktioniert nach dem Master / Slave Prinzip.

Ein Modbus Master kann mit einem oder mehreren Modbus Slaves kommunizieren. Nur der vom Modbus Master explizit angesprochene Modbus Slave darf Daten an den Modbus Master zurücksenden.

Der Modbus Master kann Aufträge zum Lesen und Schreiben von Operanden an den Modbus Slave senden:

Tabelle 45-1

Auftrag	Operand	
	Eingänge, Zeiten, Zähler	Ausgänge, Merker, Datenbausteine
Lesen	x	x
Schreiben	---	x

Adressierung Modbus Slave

Die Adresse eines Modbus Slaves kann im Bereich von 1 bis 255 liegen.

Mit der Modbus Slave Adresse Null spricht der Master alle Slaves am Bus an (Broadcast).

Anzahl Teilnehmer

Übersicht:

Tabelle 45-2

SIMATIC Familie	Physik Schnittstelle	Teilnehmer
ET 200S S7-300 S7-400	RS 485	1 Master, maximal 32 Slaves
	RS 422	1 Master, 1 Slave
	RS 232C	1 Master, 1 Slave
S7-1200	RS 485	1 Master, maximal 247 Slaves
	RS 232C	1 Master, 1 Slave

45.2 Überblick Anwenderschnittstellen

45.2.1 Allgemeines

Ein SIMATIC Controller kann dann sein:

- Modbus Master
- Modbus Slave

45.2.2 Anschluss über SIMATIC CP

Zur Kommunikation über Modbus Seriell (RTU Format) werden SIMATIC CPs und die entsprechenden Treiber benötigt. Die Treiber sind entweder auf dem SIMATIC CP integriert, oder sie sind ladbar. Ladbare Treiber müssen getrennt bestellt werden und in STEP 7 nachinstalliert werden (/4/, Kapitel "Ladbare Treiber").

SIMATIC CPs mit Integriertem Treiber für Modbus Seriell:

- ET 200S: 1SI-Modul
- S7-1200: CM 1241

SIMATIC CPs mit der Möglichkeit Treiber für Modbus Seriell zu laden:

- S7-300: CP 341
- S7-400: CP 441-2

45.2.3 Modbus Master

Überblick Kommunikationsbausteine:

Tabelle 45-3

Kommunikationsbaustein	ET200S	S7-300	S7-400	S7-1200
	1SI	CP 341	CP 441-2	CM 1241
S_SEND / S_RCV	FB 3 / FB 2	---	---	---
P_SND_RK / P_RCV_RK	---	FB 8 / FB 7	---	---
BSEND / BRCV	---	---	SFB 12 / SFB 13	---
MB_MASTER	---	---	---	Operation

Die Kommunikationsbausteine sind zu finden in:

Tabelle 45-4

Schnittstelle		STEP 7 Bibliothek	Namenserweiterung
ET200S	1SI	ET200sSI / ET200S Serial Interface (*1)	ET200sSI
S7-300	CP 341	CP PiP / CP 341	CP341
S7-400	CP 441-2	Standard Library / System Function Blocks	COM_FUNC

Tabelle 45-5

Schnittstelle		STEP 7 Basis Bibliothek	Namenserweiterung
S7-1200	CM 1241	Globale Bibliotheksoperationen	---

(*1): Die Bausteine können aus dem Internet geladen werden (/14/)

45.2.4 Modbus Slave

Überblick Kommunikationsbausteine:

Tabelle 45-6

Kommunikationsbaustein	ET200S	S7-300	S7-400	S7-1200
	1SI	CP 341	CP 441-2	CM 1241
S_MODB	FB 81	---	---	---
S_SEND	FB 3	---	---	---
S_RCV	FB 2	---	---	---
	---	FB 80	---	---
P_SND_RK	---	FB 8	---	---
P_RCV_RK	---	FB 7	---	---
	---	---	FB 180	---
MB_SLAVE	---	---	---	Operation

Die Kommunikationsbausteine sind zu finden in:

Tabelle 45-7

Schnittstelle	Kommunikationsbaustein	STEP 7 Bibliothek	Namenserweiterung
ET200S, 1SI	FB 81	ET200sSI / ET200S Serial Interface (*1)	---
	FB 3 / FB 2	ET200sSI / ET200S Serial Interface	ET200sSI
S7-300, CP 341	FB 80	Modbus (*2)	---
	FB 8 / FB 7	CP PtP / CP 341	CP341
S7-400, CP 441-2	FB 180	Modbus (*2)	---
S7-1200, CM 1241	---	STEP 7 Basis: Globale Bibliotheksoperation	---

(*1):Die Bausteine können aus dem Internet geladen werden (/14/)

(*2): Nach Installation der Modbus Slave CD wird der FB in der STEP 7 Bibliothek „Modbus“ bereitgestellt.

45.3 Modbus Master: Anwenderschnittstelle ET 200S

Die Anwenderschnittstelle ist identisch mit der Anwenderschnittstelle für ASCII und 3964(R): Siehe Kapitel 40.5

45.4 Modbus Master: Anwenderschnittstelle CP 341

45.4.1 Beschreibung

Der Modbus Master kann auf Daten in einem oder in mehreren Modbus Slaves zugreifen (schreibend und lesend). Dazu sendet der Modbus Master Aufträge an Modbus Slaves.

P_SND_RK

Auftrag an Modbus Slave senden

P_RCV_RK

Antworttelegramm vom Modbus Slave empfangen

45.4.2 Parameter P_SND_RK

Tabelle 45-8

INPUT	Typ	Bemerkung
SF	CHAR	SF = „S“
REQ	BOOL	Anstoß Auftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse CP 341
DB_NO	INT	Sendebereich in lokaler CPU
DBB_NO	INT	
LEN	INT	Länge der zu sendenden Daten
R_TYP	CHAR	Nummer der remote CPU
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig(*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): „Auftrag fertig“:

Bei schreibenden Funktionscodes: Nach Empfang Antworttelegramm

Bei lesenden Funktionscodes: Nach Empfang des Antworttelegrammes, und nach Übergabe der empfangenen Daten in die CPU

45.4.3 Parameter P_RCV_RK

Tabelle 45-9

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Daten empfangen
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse CP 341
DB_NO	INT	Empfangsbereich auf lokaler CPU
DBB_NO	INT	
OUTPUT	Typ	Bemerkung
NDR	BOOL	Daten im Empfangsbereich: ja / nein
LEN	INT	Länge der empfangenen Daten
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

45.5 Modbus Master: Anwenderschnittstelle CP 441-2

45.5.1 Beschreibung

Der Modbus Master kann auf Daten in einem oder in mehreren Modbus Slaves zugreifen (schreibend und lesend). Dazu sendet der Modbus Master Aufträge an Modbus Slaves.

BSEND

Auftrag an Modbus Slave senden

BRCV

Antworttelegramm vom Modbus Slave empfangen

45.5.2 Parameter BSEND

Tabelle 45-10

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Auftrag
ID	WORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
R_ID	DWORD	Parameter zur Adressierung innerhalb einer Verbindung. Beide Kommunikationspartner müssen hier denselben Wert verwenden.
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
SD_1	ANY	Sendebereich
LEN	WORD	Länge des zu sendenden Daten

(*1): „Auftrag fertig“:

Bei schreibenden Funktionscodes: Nach Empfang Antworttelegramm

Bei lesenden Funktionscodes: Nach Empfang des Antworttelegrammes, und nach Übergabe der empfangenen Daten in die CPU

45.5.3 Parameter BRCV

Tabelle 45-11

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
ID	WORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
R_ID	DWORD	Parameter zur Adressierung innerhalb einer Verbindung. Beide Kommunikationspartner müssen hier denselben Wert verwenden.
OUTPUT	Typ	Bemerkung
NDR	BOOL	Daten im Empfangsbereich: ja / nein
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
RD_1	ANY	Empfangsbereich in lokaler CPU
LEN	WORD	Länge der empfangenen Daten

45.6 Modbus Master: Anwenderschnittstelle CM 1241

45.6.1 Beschreibung

Der Modbus Master kann auf Daten in einem oder in mehreren Modbus Slaves zugreifen (schreibend und lesend). Dazu sendet der Modbus Master Aufträge an Modbus Slaves.

MB_MASTER

Auftrag an Modbus Slave senden

Antworttelegramm vom Modbus Slave empfangen

45.6.2 Parameter MB_MASTER

Tabelle 45-12

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Auftrag
MB_ADR	USINT	Modbus Stationsadresse
MODE	USINT	Auswahl Modus
DATA_ADDR	UDINT	Anfangsadresse im Slave
DATA_LEN	UINT	Länge der zu lesenden / schreibenden Daten
DATA_PTR	VARIANTE	Empfangspuffer (lesen) / Sendepuffer (schreiben)
OUTPUT	Typ	Bemerkung
NDR	BOOL	Auftrag läuft / Auftrag fertig (*1)
BUSY	BOOL	
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

(*1): „Auftrag fertig“:

Bei schreibenden Funktionscodes: Nach Empfang Antworttelegramm

Bei lesenden Funktionscodes: Nach Empfang des Antworttelegrammes, und nach Übergabe der empfangenen Daten in die CPU

45.7 Modbus Slave: Anwenderschnittstelle ET 200S

45.7.1 Beschreibung

Der Modbus Master kann auf Daten in einem oder in mehreren Modbus Slaves zugreifen (schreibend und lesend). Dazu sendet der Modbus Master Aufträge an Modbus Slaves. Der Modbus Slave führt den Auftrag aus, und reagiert mit einem Antworttelegramm.

S_MODB

Ausführen eines Auftrages vom Modbus Master.

S_MODB ruft intern auf: S_SEND, S_RCV (Siehe Kapitel 40.5).

45.7.2 Parameter S_MODB

Tabelle 45-13

INPUT	Typ	Bemerkung
LADDR	INT	Basisadresse ET 200S 1SI
START_TIMER	TIMER	Überwachungszeit
START_TIME	S5TIME	
DB_NO	INT	Modbus Umwandlungstabelle
OB_MASK	BOOL	Peripheriezugriffsfehler maskieren, Alarime verzögern.
CP_START	BOOL	Initialisierung
CP_START_FM	BOOL	
OUTPUT	Typ	Bemerkung
CP_NDR	BOOL	Modbus Schreibauftrag: läuft / beendet
CP_START_OK	BOOL	Fehlerinformation
CP_START_ERROR	BOOL	
ERROR_NR	WORD	Fehlerinformation
ERROR_INFO	WORD	

45.8 Modbus Slave: Anwenderschnittstelle CP 341

45.8.1 Beschreibung

Der Modbus Master kann auf Daten in einem oder in mehreren Modbus Slaves zugreifen (schreibend und lesend). Dazu sendet der Modbus Master Aufträge an Modbus Slaves. Der Modbus Slave führt den Auftrag aus, und reagiert mit einem Antworttelegramm.

FB80

Ausführen eines Auftrages vom Modbus Master

FB80 ruft intern auf: P_SND_RK, P_RCV_RK (Siehe 40.8).

45.8.2 Parameter FB80

Tabelle 45-14

INPUT	Typ	Bemerkung
LADDR	INT	Basisadresse des CP 341
START_TIMER	TIMER	Überwachungszeit
START_TIME	S5TIME	
OB_MASK	BOOL	Peripheriezugriffsfehler maskieren, Alarme verzögern.
CP_START	BOOL	Initialisierung
CP_START_FM	BOOL	
OUTPUT	Typ	Bemerkung
CP_START_NDR	BOOL	Modbus Schreibauftrag: läuft / beendet
CP_START_OK	BOOL	Fehlerinformation
CP_START_ERROR	BOOL	
ERROR_NR	WORD	Fehlerinformation
ERROR_INFO	WORD	

45.9 Modbus Slave: Anwenderschnittstelle CP 441-2

45.9.1 Beschreibung

Der Modbus Master kann auf Daten in einem oder in mehreren Modbus Slaves zugreifen (schreibend und lesend). Dazu sendet der Modbus Master Aufträge an Modbus Slaves. Der Modbus Slave führt den Auftrag aus, und reagiert mit einem Antworttelegramm.

FB180

Ausführen eines Auftrages vom Modbus Master.

45.9.2 Parameter FB180

Tabelle 45-15

INPUT	Typ	Bemerkung
ID	INT	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
START_TIMER	TIMER	Überwachungszeit
START_TIME	S5TIME	
STATUS_TIMER	TIMER	Überwachungszeit
STATUS_TIME	S5TIME	
OB_MASK	BOOL	Peripheriezugriffsfehler maskieren, Alarime verzögern.
CP_START	BOOL	Initialisierung
CP_START_FM	BOOL	
OUTPUT	Typ	Bemerkung
CS_START_NDR	BOOL	Modbus Schreibauftrag: läuft / beendet
CP_START_OK	BOOL	Fehlerinformation Initialisierung
CP_START_ERROR	BOOL	
ERROR_NR	WORD	Fehlerinformation Auftrag
ERROR_INFO	WORD	

45.10 Modbus Slave: Anwenderschnittstelle CM 1241

45.10.1 Beschreibung

Der Modbus Master kann auf Daten in einem oder in mehreren Modbus Slaves zugreifen (schreibend und lesend). Dazu sendet der Modbus Master Aufträge an Modbus Slaves. Der Modbus Slave führt den Auftrag aus, und reagiert mit einem Antworttelegramm.

MB_SLAVE

Ausführen eines Auftrages vom Modbus Master.

45.10.2 Parameter MB_SLAVE

Tabelle 45-16

INPUT	Typ	Bemerkung
MB_ADDR	USINT	Modbus Stationsadresse
MB_HOLD_REG	VARIANT	Modbus Halteregeister DB
OUTPUT	Typ	Bemerkung
NDR	BOOL	Modbus Schreibauftrag: läuft / beendet
DR	BOOL	Modbus Leseauftrag: läuft / beendet
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

46 Informationen Teil 4

In den Tabellen sind Verweise auf Informationen zu den Themen aus Teil 4 zu finden.

Alle Verweise /x/ sind zentral im Kapitel 47 hinterlegt. Dort sind auch die entsprechenden Links ins Internet zu finden.

Tabelle 46-1

/x/	Titel / Inhalt	Informationen zu
/17/	Kommunikation zwischen SIMATIC S7 und einer Modicon M340 über Modbus TCP	Infos zu: Modbus TCP
/104/	Wie kann ich von einer SIMATIC S7 eine OPEN Modbus / TCP Kommunikation aufbauen und wo erhalte ich weitere Informationen?	
/107/	Welche Ports sind für Modbus/TCP Kommunikation freigegeben und wie viele Modbus Clients können mit einer SIMATIC S7-CPU als Modbus Server kommunizieren?	
/30/	Wizard für die Kommunikation über Modbus TCP	
/14/	Funktionsbausteine, Beispiele und Handbücher der seriellen Schnittstelle ET200S 1SI	Infos zu: Modbus RTU
/23/	SIMATIC S7-300/S7-400 Ladbarer Treiber für Punkt-zu-Punkt-CPs: Modbus-Protokoll, RTU-Format, S7 ist Slave Betriebsanleitung	
/24/	SIMATIC S7-300/S7-400 Ladbarer Treiber für Punkt-zu-Punkt-CPs: Modbus-Protokoll, RTU-Format, S7 ist Master Betriebsanleitung	
/26/	Ladbarer Treiber Modbus Slave (RTU)	
/27/	Ladbarer Treiber Modbus Master (RTU)	
/200/	Applikationen zur Kommunikation Aufgabenstellung, Lösung, STEP 7 Projekt	Applikationen zu Modbus

TEIL 5: Anhang

Teil 5 ist der Anhang zum Dokument

TEIL 5: Gliederung und Inhalt

Tabelle 46-2

Kapitel	Gliederung	Inhalt
47	Literaturangaben	Verweis im Text: /x/
48	Zentrale Begriffe	Kurze Erläuterung. Muss ein Begriff detaillierter beschrieben werden, dann siehe Kapitel Hintergrundwissen.
49	Abkürzungen	
50	Hintergrundwissen	Beschreibung wichtiger Zusammenhänge
51	Betrachtete Komponenten	Bestelldaten und Versionen
52	Historie	Änderungen / Versionen der Dokumentation

47 Literaturangaben

Inhalt des Kapitels

- Sammlung von hilfreichen Informationen zur CPU-PU Kommunikation mit SIMATIC Controllern
- Sortiert nach: Informationen, FAQs und Applikationen

Literaturangaben sind im Dokument mit /x/ gekennzeichnet.

47.1 Informationen

Tabelle 47-1

/x/	Titel Inhalt	Link
/0/	Service & Support Portal Industry Automation und Drives Technologies Service & Support Portal	http://www.siemens.com/automation/service&support
/1/	SIMATIC Controller / Die innovative Lösung für alle Automatisierungsaufgaben. Broschüre April 2010 Übersicht zu SIMATIC Controller	http://www.automation.siemens.com/salesmaterial-as/brochure/de/brochure_simatic-controller_de.pdf
/2/	SIMATIC NET / Industrielle Kommunikation Broschüre November 2009 Übersicht zur industriellen Kommunikation	http://www.automation.siemens.com/cms/infocenter/dokumentencenter/sc/ic/Documentsu20Brochures/bs_k-schrift_de_1109.pdf
/3/	SIMATIC / Kommunikation mit SIMATIC Systemhandbuch 09/2006 Grundlagen zur Kommunikation mit SIMATIC	http://support.automation.siemens.com/WW/view/de/25074283
/4/	Katalog ST 70 2009 / Produkte für Totally Integrated Automation und Micro Automation Übersicht und Bestelldaten für SIMATIC Controller	http://www.automation.siemens.com/salesmaterial-as/catalog/de/st7001_d.pdf
/5/	Katalog IK PI 2009 / Industrielle Kommunikation Übersicht und Bestelldaten für Geräte der industriellen Kommunikation	http://www.automation.siemens.com/net/html_00/support/printkatalog.htm
/6/	SIMATIC System- und Standardfunktionen für S7-300/400, Referenzhandbuch, Ausgabe 05/2010 Ausführliche Beschreibung aller Funktionen	http://support.automation.siemens.com/WW/view/de/44240604
/7/	CPU 31xC und CPU 31x Technische Daten Gerätehandbuch, Ausgabe 02/2009	http://support.automation.siemens.com/WW/view/de/12996906
/8/	Automatisierungssystem S7-400 CPU-Daten Gerätehandbuch, Ausgabe 04/2009	http://support.automation.siemens.com/WW/view/de/23904550

/x/	Titel Inhalt	Link
/9/	S7-CPs für Industrial Ethernet Projektieren und in Betrieb nehmen, Ausgabe 07/2010	http://support.automation.siemens.com/WW/view/de/30374198
/10/	S7-CPs für PROFIBUS Projektieren und in Betrieb nehmen, Ausgabe 03/2009	http://support.automation.siemens.com/WW/view/de/1158693
/11/	S7 OpenModbus/TCP Produktbeschreibung, Technische Daten, Bestelldaten, Kontakt, Downloads	http://www.industry.siemens.com/industrial-services/it/de/products/simatic_add_ons/s7_open_modbus_tcp.htm
/12/	SIMATIC NET NCM S7 für PROFIBUS / FMS Band 2	http://support.automation.siemens.com/WW/view/de/1158418
/13/	SIMATIC NET Funktionen (FC) und Funktionsbausteine (FB) für SIMATIC NET S7-CPs, Programmierhandbuch, Ausgabe 08/2009	http://support.automation.siemens.com/WW/view/de/30564821
/14/	Funktionsbausteine, Beispiele und Handbücher der seriellen Schnittstelle ET200S 1SI	http://support.automation.siemens.com/WW/view/de/25358470
/15/	SIMATIC / Hardware konfigurieren und Verbindungen projektieren mit STEP 7, Handbuch, Ausgabe 03/2006	http://support.automation.siemens.com/WW/view/de/18652631
/16/	SIMATIC PROFINET IO / Von PROFIBUS DP nach PROFINET IO / Programmierhandbuch, Ausgabe 10/2006	http://support.automation.siemens.com/WW/view/de/19289930
/17/	Kommunikation zwischen SIMATIC S7 und einer Modicon M340 über Modbus TCP	http://support.automation.siemens.com/WW/view/de/38586568
/18/	Leistungsdaten Ergebnisse von Messungen zur CPU-CPU Kommunikation, in einem PROFIBUS, PROFINET/Industrial Ethernet Netz, für unterschiedliche Konfigurationen.	http://support.automation.siemens.com/WW/view/de/25209605
/19/	Punkt-zu-Punkt-Kopplung CP 340 Aufbauen und Parametrieren, Ausgabe 10/2007	http://support.automation.siemens.com/WW/view/de/1137332
	Punkt-zu-Punkt-Kopplung CP 341 Aufbauen und Parametrieren, Ausgabe 09/2008	http://support.automation.siemens.com/WW/view/de/1117397
/20/	Punkt-zu-Punkt-Kopplung CP 440 Aufbauen und Parametrieren, Ausgabe 09/2007	http://support.automation.siemens.com/WW/view/de/2042641
	Punkt-zu-Punkt-Kopplung CP 441 Aufbauen und Parametrieren, Ausgabe 10/2005	http://support.automation.siemens.com/WW/view/de/1137419
/21/	S7-300 CPU 31xC Technologische Funktionen (CPU 312C, CPU 313C, CPU 314C), Ausgabe 02/2007	http://support.automation.siemens.com/WW/view/de/12429336
/22/	--- nicht belegt ---	

<i>/x/</i>	Titel Inhalt	Link
/23/	SIMATIC S7-300/S7-400 Ladbarer Treiber für Punkt-zu-Punkt-CPs: Modbus-Protokoll, RTU-Format, S7 ist Slave Betriebsanleitung	http://support.automation.siemens.com/WW/view/de/1218007
/24/	SIMATIC S7-300/S7-400 Ladbarer Treiber für Punkt-zu-Punkt-CPs: Modbus-Protokoll, RTU-Format, S7 ist Master Betriebsanleitung	http://support.automation.siemens.com/WW/view/de/1220184
/25/	SIMATIC NET Quick Start Beispiele rund um das Thema Kommunikation Beispielprogramme und Projektierungen	http://www.siemens.com/simatic-net/quickstart
/26/	Ladbarer Treiber Modbus Slave (RTU)	http://support.automation.siemens.com/WW/view/de/27774276
/27/	Ladbarer Treiber Modbus Master (RTU)	http://support.automation.siemens.com/WW/view/de/27774018
/28/	Handbuch ET 200S Serielle Schnittstellenbaugruppen	http://support.automation.siemens.com/WW/view/de/9260793
/29/	Wizard zur Erstellung der Verbindungsdaten für die offene TCP/IP Kommunikation	http://support.automation.siemens.com/WW/view/de/25209116
/30/	Wizard für die Kommunikation über Modbus TCP	http://support.automation.siemens.com/WW/view/de/31535566
/31/	Projektierungssoftware „PtP-Param“	http://support.automation.siemens.com/WW/view/de/27013524

47.2 FAQ

Tabelle 47-2

/x/	Titel	Link
/100/	Kommunikation projektieren und programmieren --- Kommunikations-Bausteine verwenden: S7-300 CPU31x	http://support.automation.siemens.com/WW/view/de/22866139
/101/	Kommunikation projektieren und programmieren --- Kommunikations-Bausteine verwenden: S7-400 CPU41x	http://support.automation.siemens.com/WW/view/de/23522717
/102/	Kommunikation projektieren und programmieren --- Kommunikations-Bausteine verwenden: IE S7-300/400 CPs	http://support.automation.siemens.com/WW/view/de/22548794
/103/	Kommunikation projektieren und programmieren --- Kommunikations-Bausteine verwenden: PB S7-300/400 CPs	http://support.automation.siemens.com/WW/view/de/21629966
/104/	Wie kann ich von einer SIMATIC S7 eine OPEN Modbus / TCP Kommunikation aufbauen und wo erhalte ich weitere Informationen?	http://support.automation.siemens.com/WW/view/de/22660304
/105/	Kommunikation projektieren und programmieren --- Verbindungen projektieren: IE S7-300/400 CPs	http://support.automation.siemens.com/WW/view/de/22387424
/106/	Kommunikation projektieren und programmieren --- Verbindungen projektieren: PB S7-300/400 CPs	http://support.automation.siemens.com/WW/view/de/28526800
/107/	Welche Ports sind für Modbus/TCP Kommunikation freigegeben und wie viele Modbus Clients können mit einer SIMATIC S7-CPU als Modbus Server kommunizieren?	http://support.automation.siemens.com/WW/view/de/34010717

47.3 Applikationen aus Service & Support Portal

Tabelle 47-3

/x/	Inhalt	Link
/200/	Applikationen zur Kommunikation Aufgabenstellung, Lösung, STEP 7 Projekt	http://support.automation.siemens.com/WW/view/de/20229805/136000

48 Begriffe

Im Kapitel werden Begriffe erläutert, die für das Verständnis des Dokumentes von zentraler Bedeutung sind.

Manche Begriffe werden gleichwertig verwendet. Dann sind diese Begriffe mit „=“ gekennzeichnet. Beispiel: Kommunikationspartner = Partner

Einige Begriffe sind an anderen Stellen des Dokumentes ausführlicher beschrieben. In diesem Fall steht in der folgenden Tabelle ein Verweis zum entsprechenden Kapitel (Spalte „Details“).

Tabelle 48-1

Begriff	Erläuterung	Details
Auftrag = Kommunikationsauftrag	Ein Kommunikationsbaustein führt einen Kommunikationsauftrag aus. Beispiel: „Sende x Byte“	
Client, Server Master, Slave Provider, Consumer	Diese Begriffe werden bei Kommunikationsmodellen verwendet. Die Begriffe beschreiben Eigenschaften von Kommunikations-Partnern.	50.2
Controller	Ein Controller ist eine zentrale oder dezentrale Automatisierungsstation (Station) mit den Komponenten: CPU, CP, und Peripherie. <u>Zentrale Station:</u> <ul style="list-style-type: none"> Station mit zentraler Peripherie. Kommuniziert mit dezentralen Stationen über PROFINET IO oder PROFIBUS DP <u>Dezentrale Station:</u> <ul style="list-style-type: none"> Station mit dezentraler Peripherie Kommuniziert mit zentraler Stationen über PROFINET IO oder PROFIBUS DP 	4
CPU-CPU Kommunikation	CPU-CPU Kommunikation findet zwischen zwei CPUs statt: <ul style="list-style-type: none"> SIMATIC CPU_1 <-> SIMATIC CPU_2 SIMATIC CPU <-> CPU eines fremden Controllers 	4
Daten	Mit Daten sind gemeint: Nettodaten, Nutzdaten, Anwenderdaten, SIMATIC Anwenderdatenbereiche Beispiele: Datenbaustein, Merker, Eingänge, Zeiten	---
Kommunikation über einen offenen Standard	Für einen „offenen Standard“ gilt: <ul style="list-style-type: none"> Die Protokolle sind offen gelegt und international genormt. Jeder darf die Protokolle nachbauen Die Protokolle sind herstellerunabhängig. 	---
Kommunikationsbausteine	Funktionsbausteine (FB, SFB, FC, SFC), zur Einbindung in das STEP 7 Anwenderprogramm. Damit wird die Datenübertragung realisiert (senden, empfangen).	---
Master	DP-Master	
Medium	<ul style="list-style-type: none"> Netze: MPI, PB, PN/IE Rückwandbus Serielle Schnittstelle 	---
Netze	Mit Netze sind hier industrielle Netze gemeint. Diese Netze werden in der Automatisierungstechnik eingesetzt. Ein Netz kann aus einem oder mehreren Subnetzen bestehen.	1.1
Partner = Kommunikationspartner	Teilnehmer an der Kommunikation, bei der Daten ausgetauscht werden	---

47.3 Applikationen aus Service & Support Portal

Begriff	Erläuterung	Details
Projekt	Bei der Erstellung einer Automatisierungslösung mit STEP 7 werden die verschiedenen Automatisierungsaufgaben durch Steuerungsprogramme gelöst. STEP 7 fasst alle Steuerungsprogramme und die dafür benötigten Daten in einem Projekt zusammen. Ein Projekt enthält folgende Daten (Beispiel): <ul style="list-style-type: none"> • Konfigurationsdaten über den Hardware-Aufbau. • Parametrierungsdaten für die Baugruppen der Steuerung und für die dezentrale Peripherie. • Projektierungsdaten für die Kommunikation (PROFINET, ...) • Steuerungsprogramm (KOP, FUP, ...) 	---
Rückwandbus	siehe Controller	---
Schnittstelle = Kommunikationsschnittstelle	Controller kommunizieren über Medien (PN/IE, ...). Die Controller werden an Schnittstellen an das Medium angeschlossen. Eine Schnittstelle kann sein: <ul style="list-style-type: none"> • Integrierte Schnittstelle: CPU • Externe Schnittstelle: CP 	---
Send/Receive-Bausteine	Sammelbegriff für folgende Kommunikationsbausteine: AG_SEND, AG_LSEND, AG_SSEND, AG_RECV, AG_LRECV, AG_SRECV	
Slave	DP-Slave	
Subnetz	Ein Subnetz ist im Bereich der LANs (Local Area Networks) angesiedelt. Es ermöglicht die Kommunikation, zum Beispiel zwischen CPUs von Controllern, innerhalb eines räumlich begrenzten Gebietes. Ein Subnetz ist in sich abgeschlossen, es hat einen eigenen Adressraum. Mehrere Subnetze bilden ein Netz.	---
T-Bausteine	Sammelbegriff für folgende Kommunikationsbausteine: TSEND, TUSEND, TRCV, TURCV	
Verbindung	Beziehung zwischen Kommunikations-Partnern	5
Verbindungsbausteine	Funktionsbausteine (FB, SFB, FC, SFC), zur Einbindung in das STEP 7 Anwenderprogramm. Damit werden die Verbindungen realisiert (aufbauen, abbauen).	---

49 Abkürzungen

Im Kapitel werden Abkürzungen erläutert.

Tabelle 49-1

Abkürzung	Erläuterung
CBA	PROFINET CBA (Component Based Automation)
CP	communication processor: Baugruppe, die sich mit Kommunikationsaufgaben befasst, und einen Controller an ein Medium anbindet.
CPU	central procesing unit Baugruppe, auf der ein Anwenderprogramm abläuft. In diesem Anwenderprogramm werden Daten gesendet oder empfangen.
DP	Dezentrale Peripherie
E, A, M, D, T, Z	SIMATIC S7 Speicherbereiche: Prozessabbild Eingänge (E), Prozessabbild Ausgänge (A), Merker (M), Datenbaustein (D), Zeiten (T), Zähler (Z)
FMS	Fieldbus Message Specification
GD	Globaldaten
HW Konfig	Hardwarekonfiguration: Tool zur Konfiguration von Hardware in STEP 7
IOC	PROFINET IO Controller
IOD	PROFINET IO Device
IoT	ISO on TCP
MPI	multi point interface
NetPro	Netzprojektierung: Tool zum Projektieren von Verbindungen in STEP 7
OP	Bediengerät
PB	PROFIBUS
PG	Programmiergerät
PN/IE	PROFINET / Industrial Ethernet
PNIO	PROFINET IO
S/R-Bausteine	Send/Receive-Bausteine
S7-CP	CP der SIMATIC S7
S7-CPU	CPU der SIMATIC S7

50 Hintergrundwissen

Im Kapitel wird Hintergrundwissen vermittelt.

50.1 ISO/OSI-Referenzmodell

Das ISO/OSI-Referenzmodell ist ein genormtes Modell zur Beschreibung von offenen (herstellerunabhängigen) Kommunikationssystemen. Das Modell beschreibt die Anforderungen an ein Kommunikationssystem. Die konkrete Umsetzung wird nicht beschrieben. Die meisten frei nutzbaren Protokolle basieren auf diesem Referenzmodell (zum Beispiel: TCP/IP).

Das Modell besteht aus 7 Schichten, mit folgenden Eigenschaften:

- Jede Schicht hat fest definierte Aufgaben zu erfüllen
- Die Schichten sind von einander unabhängig

Erläuterung der Schichten

Tabelle 50-1

Schicht	Bezeichnung	Aufgabe (Beispiele)	Einordnung
Schicht 7 (layer 7)	Anwendungsschicht (Application Layer)	Schnittstelle zum STEP 7 Anwenderprogramm (Quittierung auf Anwendungsebene, ...).	Anwendungsorientiert
Schicht 6 (layer 6)	Darstellungsschicht (Presentation Layer)	Interpretation der Daten (Umsetzung der normierten Darstellung des Kommunikationssystems in eine gerätespezifische Form.)	
Schicht 5 (layer 5)	Kommunikationssteuerungsschicht (Session Layer)	Organisation der Datenübertragung	
Schicht 4 (layer 4)	Transportschicht (Transport Layer)	Herstellung einer Verbindung zwischen zwei Geräten : <ul style="list-style-type: none"> • Aufbau, Abbau, Aufrechterhalten Übermittlung von Datenpaketen: <ul style="list-style-type: none"> • Zerlegung der Daten in Pakete (Segmentierung) • Verhinderung des Staus von Paketen • Quittierung auf Transportebene 	Transportorientiert
Schicht 3 (layer 3)	Netzwerkschicht (Network Layer)	Vermittlung und Zustellung von Daten: <ul style="list-style-type: none"> • Festlegung der Kommunikationswege • Adressierung der Kommunikationspartner • Routing im Netz 	
Schicht 2 (mac / layer 2)	Datensicherungsschicht (Data Link Layer)	Überwachung und Organisation des Zugriffs auf das Übertragungsmedium (Flusskontrolle, ...) Korrekte Übermittlung der Daten (Prüfsumme, ...)	
Schicht 1 (layer 1)	Physikalische Schicht (Physical Layer)	Festlegung der physikalische Verbindung zwischen zwei Geräten (Übertragungsmedium, Baudrate, ...)	

50.2 Kommunikationsmodelle

Kommunikationsmodelle (kurz: Modelle) beschreiben das Prinzip einer Kommunikationsbeziehung. Sie geben an, welche Rolle die beiden Kommunikationspartner beim Datenaustausch spielen.

50.2.1 Client und Server

Die Begriffe Client und Server werden im Dokument wie folgt verwendet:

Anwendungsbereich

Netze: PN/IE, PB, MPI

Client

Eigenschaften

Ein Client kann mit einem Client oder einem Server Daten austauschen.

Erbringen der Kommunikationsleistung im Client

Im STEP 7 Anwenderprogramm müssen Vorkehrungen getroffen werden:

- Kommunikationsbausteine programmieren, und/oder
- Verbindungen projektieren/programmieren

Server

Eigenschaften

Ein Server kann mit einem Client Daten austauschen.

Der Anstoß zum Datenaustausch kommt immer von einem Client. D.h. ein Server kann keine Initiative für einen Datenaustausch übernehmen.

Erbringen der Kommunikationsleistung im Server

Hier sind zwei Fälle zu unterscheiden.

Fall 1: Die Kommunikation wird ausschließlich vom Betriebssystem erbracht. D.h. die Kommunikation ist Systemleistung.

Fall 2: Im STEP 7 Anwenderprogramm müssen Vorkehrungen getroffen werden:

- Kommunikationsbausteine programmieren, und/oder
- Verbindungen projektieren/programmieren

Client / Client Kommunikation

Beide Kommunikationspartner sind Clients.

Einer der beiden Clients übernimmt die Initiative für die Kommunikation.

Client / Server Kommunikation

Ein Kommunikationspartner ist Client, ein Kommunikationspartner ist Server.

Nur der Client kann die Initiative für die Kommunikation übernehmen.

50.2.2 Master und Slave

Die Begriffe Master und Slave werden im Dokument wie folgt verwendet:

Anwendungsbereich

Netze: PB (Kommunikationsart DP-Kommunikation)

Serielle Schnittstelle: Modbus Seriell, ...

Master

Ein Master hat die Initiative beim Datenaustausch (verhält sich aktiv):

- Sendet Daten an Slave
- Empfängt Daten vom Slave, die er vorher vom Slave angefordert hat

Slave

Ein Slave hat keine Initiative beim Datenaustausch (verhält sich passiv):

- Sendet, nur nach Aufforderung vom Master, Daten an den Master
- Empfängt Daten vom Master

Master / Slave Kommunikation

Ein Kommunikationspartner ist Master, ein Kommunikationspartner ist Slave.

Der Master hat die Initiative.

Master / Master Kommunikation

Beide Kommunikationspartner sind Master.

Beide Kommunikationspartner können aus Eigeninitiative senden

DP-Kommunikation

Master / Master Kommunikation ist möglich, wird im Dokument aber nicht betrachtet. Dazu wäre zusätzliche Hardware erforderlich: DP/DP-Koppler.

Serielle Schnittstele

Master / Master Kommunikation ist nicht möglich.

50.2.3 Consumer und Provider

Die Begriffe Consumer und Provider werden im Dokument wie folgt verwendet:

Anwendungsbereich

Netz: PN/IE (Kommunikationsart PNIO)

Consumer

Bekommt unaufgefordert Daten vom Provider.

Provider

Sendet Daten ohne Aufforderung an den Consumer.

Consumer / Provider Kommunikation

Ein Kommunikationspartner ist Consumer, ein Partner ist Provider.

Consumer und Provider sind gleichberechtigte Teilnehmer am Netz.

50.3 Quittierung

Werden Daten übertragen, dann gibt es unterschiedliche Rückmeldungen (Quittungen) an das STEP 7 Anwenderprogramm.

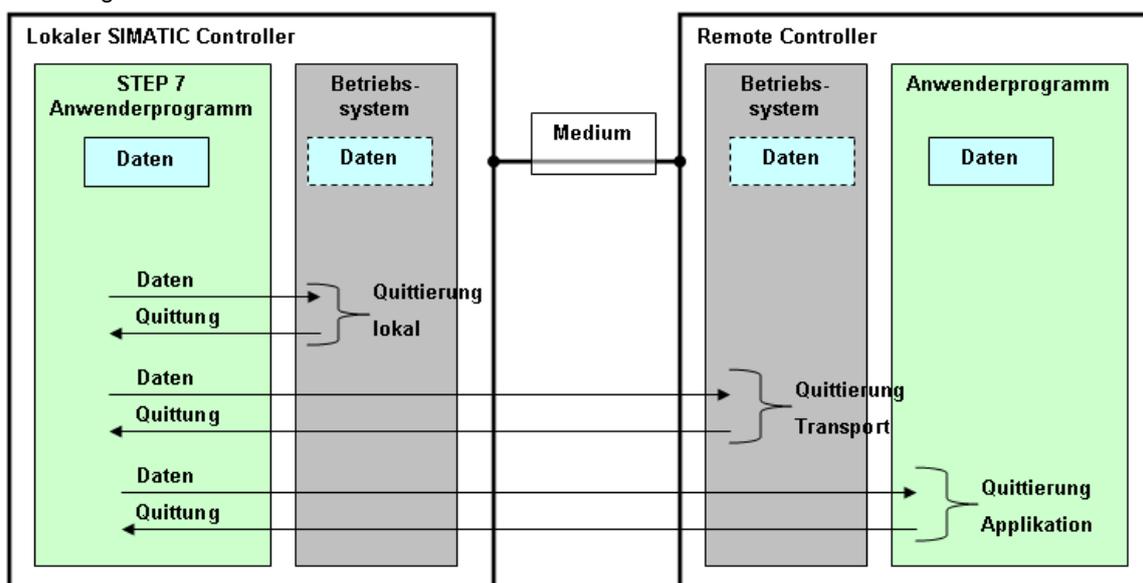
Im Dokument wird zwischen folgenden Quittierungen unterschieden:

Tabelle 50-2

Quittierung	Bedeutung	Erläuterungen
lokal	die Daten liegen im Betriebssystem (CPU oder CP) des lokalen Controllers	Keine Aussage, ob die Daten über das Medium (PROFINET/IE, ...) übertragen wurden.
Transport	die Daten liegen im Betriebssystem (CPU oder CP) des remote Controllers	Die Daten wurden über das Medium (PROFINET/IE, ...) übertragen.
Applikation	die Daten liegen in der Applikation des remote Controllers	

Das Bild verdeutlicht die Zusammenhänge.

Abbildung 50-1



51 Betrachtete Komponenten

Im Kapitel werden die Komponenten aufgelistet, für welche die Aussagen im Dokument gültig sind.

Für die Komponenten wird angegeben:

- Bestell-Code (MLFB)
- Ausgabestand / Version

Stand der Komponenten:

- September 2010

51.1 Programmierwerkzeuge

SIMATIC Familie S7-1200

SIMATIC STEP 7 Basic, V10.5, SP2

Alle anderen SIMATIC Familien

SIMATIC STEP 7, V5.5

SIMATIC iMAP V3.0, SP1

SIMATIC Distributed Safety V5.4, SP5

51.2 SIMATIC CPU

Tabelle 51-1

Controller	Familie	CPU		MLFB	Version
modular	ET 200 CPU	ET 200S	IM151-8(F) PN/DP CPU	6ES7 151-8AB01-0AB0 (6ES7 151-8FB01-0AB0)	FW V3.2
		ET 200S	IM151-7(F) CPU	6ES7 151-7AA20-0AB0 (6ES7 151-7FA20-0AB0)	FW V2.6
		ET 200Pro	IM154-8(F) PN/DP CPU	6ES7 154-8AB01-0AB0 (6ES7 154-8FB01-0AB0)	FW V3.2
	S7-300	CPU 312		6ES7 312-1AE14-0AB0	FW V3.0
		CPU 314		6ES7 314-1AG14-0AB0	FW V3.0
		CPU 312C		6ES7 312-5BE03-0AB0	FW V2.6
		CPU 313C		6ES7 313-5BF03-0AB0	FW V2.6
		CPU 313C-2 DP		6ES7 313-6CF03-0AB0	FW V2.6
		CPU 314C-2 DP		6ES7 314-6CG03-0AB0	FW V2.6
		CPU 313C-2 PtP		6ES7 313-6BF03-0AB0	FW V2.6
		CPU 314C-2 PtP		6ES7 314-6BG03-0AB0	FW V2.6
		CPU 315(F)-2 DP		6ES7 315-2AH14-0AB0 (6ES7 315-6FF04-0AB0)	FW V3.0
		CPU 317(F)-2 DP		6ES7 317-2AJ10-0AB0 (6ES7 317-6FF03-0AB0)	FW V2.6
		CPU 315(F)-2 PN/DP		6ES7 315-2EH14-0AB0 (6ES7 315-2FJ14-0AB0)	FW V3.2
		CPU 317(F)-2 PN/DP		6ES7 317-2EK14-0AB0 (6ES7 317-2FK14-0AB0)	FW V3.2
		CPU 319(F)-3 PN/DP		6ES7 318-3EL01-0AB0 (6ES7 318-3FL01-0AB0)	FW V3.2
	S7-400	CPU 412-1		6ES7 412-1XJ05-0AB0	FW V5.3
		CPU 412-2		6ES7 412-2XJ05-0AB0	FW V5.3
		CPU 414-2		6ES7 414-2XK05-0AB0	FW V5.3
		CPU 414-3		6ES7 414-3XM05-0AB0	FW V5.3
		CPU 416(F)-2		6ES7 416-2XN05-0AB0 (6ES7 416-2FN05-0AB0)	FW V5.3
		CPU 416-3		6ES7 416-3XR05-0AB0	FW V5.3
		CPU 417-4		6ES7 417-4XT05-0AB0	FW V5.3
		CPU 412-2 PN		6ES7412-2EK06-0AB0	FW V6.0
		CPU 414(F)-3 PN/DP		6ES7 414-3EM06-0AB0 (6ES7414-3FM06-0AB0)	FW V6.0
	CPU 416(F)-3 PN/DP		6ES7 416-3ES06-0AB0 (6ES7 416-3FS06-0AB0)	FW V6.0	
	S7-1200	CPU 1211C		6ES7211-xxx-0XB0	
		CPU 1212C		6ES7212-xxx-0XB0	
		CPU 1214C		6ES7214-xxx-0XB0	

Controller	Familie	CPU	MLFB	Version
embedded	S7-mEC (*1)	EC31	6ES7677-1DD00-0BB0	
	Box PC (*1)	IPC427C Bundles mit RTX	6ES7 675-1D...	
	Panel PC (*1)	HMI IPC477C Bundles mit RTX	6AV7 884...	
	WinAC MP (*2)	MP177 mit WinAC MP	6ES7 671-4EE00-0YA0	
MP277 mit WinAC MP		6ES7 671-5EF01-0YA0		
MP377 mit WinAC MP		6ES7 671-7EG01-0YA0		
PC-based	WinAC RTX	WinAC RTX (F) 2009 (Software)	6ES7 671-0RC07-0YA0 (6ES7 671-1RC07-0YA0)	V4.5

Erläuterungen zur Tabelle:

(*1): mit WinAC RTX 2009 als Software Controller

(*2): mit WinAC MP 2008 als Software Controller

51.3 SIMATIC CP

Tabelle 51-2

Controller	Familie	CP			MLFB	Version
modular	ET 200 CPU	PB	ET 200S	DP Mastermodul	6ES7 138-4HA00-0AB0	FW V1.0
		PtP	ET 200S	1 SI 3964/ASCII	6ES7138-4DF01-0AB0	FW V1.4
		PtP	ET 200S	1 SI Modbus/USS	6ES7138-4DF11-0AB0	FW V1.4
	S7-300	PB	CP 342-5		6GK7 342-5DA02-0XE0	FW V5.0
		PB	CP 342-5 FO		6GK7342-5DF00-0XE0	FW V5.0
		PB	CP 343-5		6GK7 343-5FA01-0XE0	FW V4.0
		PN/IE	CP 343-1 Lean		6GK7 343-1CX10-0XE0	FW V2.3
		PN/IE	CP 343-1		6GK7 343-1EX30-0XE0	FW V2.3
		PN/IE	CP 343-1 Advanced		6GK7 343-1GX30-0XE0	FW V1.1
		PN/IE	CP 343-1 ERPC		6GK7343-1FX00-0XE0	FW V1.0
		PtP	CP 340		6ES7340-1xH02-0AE0	FW V1.0
		PtP	CP 341		6ES7341-1xH02-0AE0	FW V2.0
	S7-400	PB	CP 443-5 Basic		6GK7 443-5FX02-0XE0	FW V4.0
		PB	CP 443-5 Extended		6GK7 443-5DX04-0XE0	FW V6.4
		PN/IE	CP 443-1		6GK7 443-1EX20-0XE0	FW V2.1
		PN/IE	CP 443-1 Advanced		6GK7 443-1GX20-0XE0	FW V2.1
		PtP	CP 440		6ES7440-1CS00-0YE0	FW V1.0
		PtP	CP 441-1		6ES7441-1AA04-0AE0	FW V1.0
		PtP	CP 441-2		6ES7441-2AA04-0AE0	FW V1.0
	S7-1200	PtP	CM 1241 RS485		6ES7241-1CH30-0XB0	
PtP		CM 1241 RS232		6ES7241-1AH30-0XB0		
embedded	S7-mEC	PB	EM PCI-104		6ES7677-1DD40-1AA0	
		PN/IE	EM PC		6ES7677-1DD50-2AA0	
		PtP	CP 340		6ES7340-1xH02-0AE0	V1.0
	Box PC	(*1)	---		---	---
	Panel PC	(*1)	---		---	---
	WinAC MP	---	---		---	---
PC-based	WinAC RTX	(*1)	---		---	---

(*1): CPs für Box PC, Panel PC und WinAC RTX

Prinzipiell sind folgende Baugruppen einsetzbar:

Anschluss an PN/IE: CP 1616, CP 1604

Anschluss an PB: CP 56-11-A2, CP 5621, CP 5613, CP 5613-A2, CP 5603

Welche Baugruppen im konkreten Einsatzfall möglich sind, entnehmen Sie bitte dem Katalog.

52 Historie

52.1 Versionen

Tabelle 52-1

Version	Datum	
V1.0	04 / 2004	Erste Ausgabe
V2.0	11 / 2010	Komplette Überarbeitung
V2.01	01 / 2011	Fehlerbeseitigung

52.2 Wesentliche Änderungen

Tabelle 52-2

Version	Änderung
V1.0 -> V2.0	<ul style="list-style-type: none"> • Aktualisierung mit neuen Komponenten • Neue Struktur des Dokumentes
V2.0 -> V2.01	Seite 368: Absatz gelöscht: SIMATIC S7-1200
	Seite 398: SIMATIC CPU ergänzt: IM151-7 F CPU