

Kompendium • 07/2013

CPU-CPU Kommunikation mit SIMATIC Controllern

SIMATIC S7

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 kundenspezifischen 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 diesem 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.

Security-hinweise

Siemens bietet Produkte und Lösungen mit Industrial Security-Funktionen an, die den sicheren Betrieb von Anlagen, Lösungen, Maschinen, Geräten und/oder Netzwerken unterstützen. Sie sind wichtige Komponenten in einem ganzheitlichen Industrial Security-Konzept. Die Produkte und Lösungen von Siemens werden unter diesem Gesichtspunkt ständig weiterentwickelt. Siemens empfiehlt, sich unbedingt regelmäßig über Produkt-Updates zu informieren.

Für den sicheren Betrieb von Produkten und Lösungen von Siemens ist es erforderlich, geeignete Schutzmaßnahmen (z. B. Zellenschutzkonzept) zu ergreifen und jede Komponente in ein ganzheitliches Industrial Security-Konzept zu integrieren, das dem aktuellen Stand der Technik entspricht. Dabei sind auch eingesetzte Produkte von anderen Herstellern zu berücksichtigen. Weitergehende Informationen über Industrial Security finden Sie unter <http://www.siemens.com/industrialsecurity>.

Um stets über Produkt-Updates informiert zu sein, melden Sie sich für unseren produktspezifischen Newsletter an. Weitere Informationen hierzu finden Sie unter <http://support.automation.siemens.com>.

Inhaltsverzeichnis

Gewährleistung, Haftung und Support	2
1 Vorbemerkungen zum Dokument	14
1.1 Farbige Textstellen	14
1.2 Gegenstand des Dokumentes	14
1.3 Veranlassung und Zielsetzung des Dokumentes	16
1.4 Eigenschaften und Nutzen des Dokumentes	17
1.5 Gültigkeitsbereich des Dokumentes	17
1.6 Abdeckung des Dokumentes	18
1.7 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: Fremde Controller	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 Sprungverteiler	24
3.2.3 Übersicht Schnittstellen und Kommunikationsarten	25
3.2.4 Kombinationen Controller 1 / Controller 2	26
3.2.5 Kommunikationsarten	28
3.2.6 Zusammenfassung	30
4 ***** TEIL 1: Einführung *****	31
4.1 Gliederung und Inhalt	31
5 Modelle zur CPU-CPU Kommunikation	32
5.1 Definition Controller	32
5.2 Definition CPU-CPU Kommunikation	32
5.3 CPUs in unterschiedlichen zentralen Stationen	33
5.4 CPUs in zentraler und dezentraler Station	34
5.5 CPUs innerhalb einer zentralen Station	35
6 Verbindungen bei SIMATIC	36
6.1 Verbindungen	36
6.2 Protokolle	37
6.3 Verbindungsressourcen	37
6.4 Einrichten von Verbindungen	38
6.4.1 Projektierte Verbindung	38
6.4.2 Nicht projektierte Verbindung	39
7 Datenkonsistenz bei SIMATIC	41
7.1 Definitionen	41
7.2 Systembedingte Datenkonsistenz	42
7.3 Weitergehende Datenkonsistenz	43
8 SIMATIC Controller	44
9 Medien zur SIMATIC Kommunikation	46
10 PROFINET/Industrial Ethernet (PN/IE)	47

10.1	Vorbemerkung.....	47
10.2	Ethernet.....	47
10.3	Industrial Ethernet (IE).....	49
10.4	PROFINET (PN).....	50
11	PROFIBUS (PB).....	52
12	MPI.....	53
13	SIMATIC Rückwandbus.....	54
14	Serielle Schnittstelle (PtP).....	55
15	Gegenüberstellung der Medien.....	56
16	Schnittstellen der SIMATIC Familien.....	57
17	Informationen zu Teil 1.....	58
18	***** TEIL 2: Auswahlhilfe *****	59
18.1	Gliederung und Inhalt.....	59
19	Vorbemerkungen.....	60
19.1	Prinzip der Darstellung aller Kombinationen.....	60
19.1.1	Gliederung des Dokumentes (Inhaltsverzeichnis).....	61
19.1.2	Kombinationen (Sprungverteiler).....	61
19.2	Tabelle Schnittstellen.....	62
19.2.1	Zweck der Tabelle.....	62
19.2.2	Aufbau der Tabelle.....	62
19.2.3	Abkürzungen und Indizes.....	64
19.3	Tabelle Kombinationen.....	65
19.3.1	Zweck der Tabelle.....	65
19.3.2	Aufbau der Tabelle.....	65
19.3.3	Reales Beispiel.....	68
19.3.4	Abkürzungen und Indizes.....	69
19.4	Tabelle Kommunikationsarten-Kompakt.....	70
19.4.1	Zweck der Tabelle.....	70
19.4.2	Aufbau der Tabelle.....	70
19.5	Übersicht der Indizes in den Tabellen.....	71
20	>>> SPRUNGVERTEILER >>>	72
20.1	Sprungverteiler: Netz PN/IE.....	73
20.1.1	Übersicht Schnittstellen und Kommunikationsarten.....	73
20.1.2	Kombinationen Controller 1 / Controller 2.....	73
20.1.3	Kommunikationsarten.....	73
20.2	Sprungverteiler: Netz PB.....	74
20.2.1	Übersicht Schnittstellen und Kommunikationsarten.....	74
20.2.2	Kombinationen Controller 1 / Controller 2.....	74
20.2.3	Kommunikationsarten.....	74
20.3	Sprungverteiler: Netz MPI.....	75
20.3.1	Übersicht Schnittstellen und Kommunikationsarten.....	75
20.3.2	Kombinationen Controller 1 / Controller 2.....	75
20.3.3	Kommunikationsarten.....	75
20.4	Sprungverteiler: SIMATIC Rückwandbus.....	76
20.4.1	Übersicht Schnittstellen und Kommunikationsarten.....	76
20.4.2	Kombinationen Controller 1 / Controller 2.....	76
20.4.3	Kommunikationsarten.....	76
20.5	Sprungverteiler: Serielle Schnittstellen.....	77
20.5.1	Übersicht Schnittstellen und Kommunikationsarten.....	77
20.5.2	Kommunikationsarten.....	77

21	Auswahlhilfe: PROFINET/Industrial Ethernet (PN/IE)	78
21.1	PN/IE: Inhalt des Kapitels	78
21.2	PN/IE: Schnittstellen und Kommunikationsarten	79
21.2.1	ET 200 CPU an PN/IE	79
21.2.2	S7-300 an PN/IE	80
21.2.3	S7-400 an PN/IE	81
21.2.4	S7-1200 an PN/IE	82
21.2.5	S7-1500 an PN/IE	83
21.2.6	S7-mEC an PN/IE	84
21.2.7	Box PC an PN/IE	85
21.2.8	Panel PC an PN/IE	86
21.2.9	WinAC RTX an PN/IE	87
21.3	PN/IE: Controller 1 = ET 200 CPU	88
21.3.1	ET 200 CPU / ET 200 CPU	88
21.3.2	ET 200 CPU / S7-300	89
21.3.3	ET 200 CPU / S7-400	90
21.3.4	ET 200 CPU / S7-1200	91
21.3.5	ET 200 CPU / S7-1500	92
21.3.6	ET 200 CPU / WinAC RTX	93
21.4	PN/IE: Controller 1 = S7-300	94
21.4.1	(S7-300 / ET 200 CPU)	94
21.4.2	S7-300 / S7-300	95
21.4.3	S7-300 / S7-400	96
21.4.4	S7-300 / S7-1200	97
21.4.5	S7-300 / S7-1500	98
21.4.6	S7-300 / WinAC RTX	99
21.5	PN/IE: Controller 1 = S7-400	100
21.5.1	(S7-400 / ET 200 CPU)	100
21.5.2	(S7-400 / S7-300)	100
21.5.3	S7-400 / S7-400	101
21.5.4	S7-400 / S7-1200	102
21.5.5	S7-400 / S7-1500	103
21.5.6	S7-400 / WinAC RTX	104
21.6	PN/IE: Controller 1 = S7-1200	105
21.6.1	(S7-1200 / ET 200 CPU)	105
21.6.2	(S7-1200 / S7-300)	105
21.6.3	(S7-1200 / S7-400)	105
21.6.4	S7-1200 / S7-1200	106
21.6.5	S7-1200 / S7-1500	107
21.6.6	S7-1200 / WinAC RTX	108
21.7	PN/IE: Controller 1 = S7-1500	109
21.7.1	(S7-1500 / ET 200 CPU)	109
21.7.2	(S7-1500 / S7-300)	109
21.7.3	(S7-1500 / S7-400)	109
21.7.4	(S7-1500 / S7-1200)	109
21.7.5	S7-1500 / S7-1500	110
21.7.6	S7-1500 / WinAC RTX	111
21.8	PN/IE: Controller 1 = WinAC RTX	112
21.8.1	(WinAC RTX / ET 200 CPU)	112
21.8.2	(WinAC RTX / S7-300)	112
21.8.3	(WinAC RTX / S7-400)	112
21.8.4	(WinAC RTX / S7-1200)	112
21.8.5	(WinAC RTX / S7-1500)	112
21.8.6	WinAC RTX / WinAC RTX	113
21.9	PN/IE: Übersicht Kommunikationsarten	114
22	Auswahlhilfe: PROFIBUS (PB)	115

22.1	PB: Inhalt des Kapitels	115
22.2	PB: Schnittstellen und Kommunikationsarten	116
22.2.1	ET 200 CPU an PB	116
22.2.2	S7-300 an PB	117
22.2.3	S7-400 an PB	118
22.2.4	S7-1200 an PB	119
22.2.5	S7-1500 an PB	120
22.2.6	S7-mEC an PB	121
22.2.7	Box PC an PB	122
22.2.8	Panel PC an PB	123
22.2.9	WinAC RTX an PB	124
22.3	PB: Controller 1 = ET 200 CPU	125
22.3.1	ET 200 CPU / ET 200 CPU	125
22.3.2	ET 200 CPU / S7-300	126
22.3.3	ET 200 CPU / S7-400	127
22.3.4	ET 200 CPU / S7-1200	128
22.3.5	ET 200 CPU / S7-1500	129
22.3.6	ET 200 CPU / WinAC RTX	130
22.4	PB: Controller 1 = S7-300	131
22.4.1	(S7-300 / ET 200 CPU)	131
22.4.2	S7-300 / S7-300	132
22.4.3	S7-300 / S7-400	133
22.4.4	S7-300 / S7-1200	134
22.4.5	S7-300 / S7-1500	135
22.4.6	S7-300 / WinAC RTX	136
22.5	PB: Controller 1 = S7-400	137
22.5.1	(S7-400 / ET 200 CPU)	137
22.5.2	(S7-400 / S7-300)	137
22.5.3	S7-400 / S7-400	138
22.5.4	S7-400 / S7-1200	139
22.5.5	S7-400 / S7-1500	140
22.5.6	S7-400 / WinAC RTX	141
22.6	PB: Controller 1 = S7-1200	142
22.6.1	(S7-1200 / ET 200 CPU)	142
22.6.2	(S7-1200 / S7-300)	142
22.6.3	(S7-1200 / S7-400)	142
22.6.4	S7-1200 / S7-1200	143
22.6.5	S7-1200 / S7-1500	144
22.6.6	S7-1200 / WinAC RTX	145
22.7	PB: Controller 1 = S7-1500	146
22.7.1	(S7-1500 / ET 200 CPU)	146
22.7.2	(S7-1500 / S7-300)	146
22.7.3	(S7-1500 / S7-400)	146
22.7.4	(S7-1500 / S7-1200)	146
22.7.5	S7-1500 / S7-1500	147
22.7.6	S7-1500 / WinAC RTX	148
22.8	PB: Controller 1 = WinAC RTX	149
22.8.1	(WinAC RTX / ET 200 CPU)	149
22.8.2	(WinAC RTX / S7-300)	149
22.8.3	(WinAC RTX / S7-400)	149
22.8.4	(WinAC RTX / S7-1200)	149
22.8.5	(WinAC RTX / S7-1500)	149
22.8.6	WinAC RTX / WinAC RTX	150
22.9	PB: Übersicht Kommunikationsarten	151
23	Auswahlhilfe: MPI	152
23.1	MPI: Inhalt des Kapitels	152

23.2	MPI: Schnittstellen und Kommunikationsarten.....	153
23.2.1	ET 200 CPU an MPI.....	153
23.2.2	S7-300 an MPI	154
23.2.3	S7-400 an MPI	155
23.3	MPI: Controller 1 = ET 200 CPU	156
23.3.1	ET 200 CPU / ET 200 CPU	156
23.3.2	ET 200 CPU / S7-300.....	157
23.3.3	ET 200 CPU / S7-400.....	158
23.4	MPI: Controller 1 = S7-300.....	159
23.4.1	(S7-300 / ET 200 CPU)	159
23.4.2	S7-300 / S7-300	159
23.4.3	S7-300 / S7-400	160
23.5	MPI: Controller 1 = S7-400.....	161
23.5.1	(S7-400 / ET 200 CPU)	161
23.5.2	(S7-400 / S7-300).....	161
23.5.3	S7-400 / S7-400	161
23.6	MPI: Übersicht Kommunikationsarten	162
24	Auswahlhilfe: SIMATIC Rückwandbus	163
24.1	Inhalt des Kapitels	163
24.2	SIMATIC Rückwandbus: Schnittstellen und Kommunikationsarten.....	164
24.3	Controller 1 = S7-400 / Controller 2 = S7-400	164
24.4	Übersicht Kommunikationsarten	165
25	Auswahlhilfe: Serielle Schnittstelle (PtP).....	166
25.1	Inhalt des Kapitels	166
25.2	ET 200 CPU an PtP	167
25.3	S7-300 an PtP	168
25.4	S7-400 an PtP	169
25.5	S7-1200 an PtP	170
25.6	S7-1500 an PtP	171
25.7	S7-mEC an PtP	172
25.8	Box PC an PtP	173
25.9	Panel PC an PtP	173
25.10	WinAC RTX an PtP	173
25.11	Dezentrale Station ET 200	174
26	Informationen zu Teil 2.....	175
27	***** TEIL 3: Kommunikationsarten *****	176
27.1	Gliederung und Inhalt.....	176
28	Vorbemerkungen	177
28.1	Kapitel: Merkmale.....	177
28.2	Kapitel: Details Kommunikationsart	177
28.2.1	Zweck der Tabelle	177
28.2.2	Aufbau der Tabelle	178
28.2.3	Kriterien der Tabelle	179
28.3	Kapitel: Überblick Anwenderschnittstellen	181
28.4	Kapitel: Anwenderschnittstellen	181
29	SIMATIC S7 spezifische Kommunikation.....	183
29.1	Merkmale.....	183
29.2	Übersicht	183
30	Globaldaten-Kommunikation	184

30.1	Merkmale.....	184
30.2	Details Kommunikationsart.....	185
30.3	Überblick Anwenderschnittstellen	186
30.4	Anwenderschnittstelle GD_SND, GD_RCV	187
30.4.1	Beschreibung	187
30.4.2	Parameter für GD_SND	187
30.4.3	Parameter für GD_RCV	187
31	S7-Basiskommunikation	188
31.1	Merkmale.....	188
31.2	Details Kommunikationsart.....	189
31.3	Überblick Anwenderschnittstellen	190
31.4	Anwenderschnittstelle X_SEND / X_RCV	191
31.4.1	Beschreibung	191
31.4.2	Parameter für X_SEND	191
31.4.3	Parameter für X_RCV	191
31.5	Anwenderschnittstelle X_PUT, X_GET	192
31.5.1	Beschreibung	192
31.5.2	Parameter für X_PUT	192
31.5.3	Parameter für X_GET	192
31.6	Anwenderschnittstelle I_PUT, I_GET	193
31.6.1	Beschreibung	193
31.6.2	Parameter für I_PUT	193
31.6.3	Parameter für I_GET	193
32	S7-Kommunikation	194
32.1	Merkmale.....	194
32.2	Details Kommunikationsart.....	195
32.3	Überblick Anwenderschnittstellen	197
32.4	Anwenderschnittstelle: Typ "USEND / URCV"	198
32.4.1	Vorbemerkung.....	198
32.4.2	Beschreibung	198
32.4.3	Parameter für Typ "USEND"	199
32.4.4	Parameter für Typ "URCV"	199
32.5	Anwenderschnittstelle: BSEND / BRCV	200
32.5.1	Beschreibung	200
32.5.2	Parameter für BSEND	201
32.5.3	Parameter für BRCV	201
32.6	Anwenderschnittstelle: Typ "PUT, GET"	202
32.6.1	Vorbemerkung.....	202
32.6.2	Beschreibung	202
32.6.3	Parameter für Typ "PUT"	203
32.6.4	Parameter für Typ "GET"	203
33	PROFINET/Industrial Ethernet (PN/IE)	204
33.1	Merkmale.....	204
33.2	Übersicht	204
34	PN/IE: Offene-Kommunikation mit Send/Receive-Bausteinen.....	205
34.1	Merkmale.....	205
34.2	Details Kommunikationsart.....	206
34.3	Überblick Anwenderschnittstellen	208
34.4	Anwenderschnittstelle AG_xSEND, AG_xRECV	209
34.4.1	Beschreibung	209
34.4.2	Parameter für AG_SEND, AG_LSEND, AG_SSEND	210
34.4.3	Parameter für AG_RECV, AG_LRECV, AG_SSRECV	210
34.5	Anwenderschnittstelle FETCH, WRITE (Server).....	211

35	PN/IE: Offene-Kommunikation mit „T-Bausteinen“	212
35.1	Merkmale.....	212
35.2	Details Kommunikationsart.....	213
35.3	Überblick Anwenderschnittstellen	215
35.4	Anwenderschnittstelle TSEND / TRCV	216
35.4.1	Beschreibung	216
35.4.2	Parameter für TSEND	216
35.4.3	Parameter für TRCV.....	216
35.5	Anwenderschnittstelle TUSEND / TURCV	217
35.5.1	Beschreibung	217
35.5.2	Parameter für TUSEND.....	217
35.5.3	Parameter für TURCV	217
35.6	Anwenderschnittstelle Verbindungsbausteine	218
35.6.1	Beschreibung	218
35.6.2	Parameter für TCON	218
35.6.3	Parameter für TDISCON	218
35.7	Anwenderschnittstelle TSEND_C / TRCV_C	219
35.7.1	Beschreibung	219
35.7.2	Parameter für TSEND_C.....	219
35.7.3	Parameter für TRCV_C.....	220
35.8	Anwenderschnittstelle FETCH, WRITE (Server).....	221
35.8.1	Beschreibung	221
35.8.2	Parameter für FW_TCP.....	221
35.8.3	Parameter für FW_IOT.....	221
36	PN/IE: CBA	222
36.1	Merkmale.....	222
36.2	Anwenderschnittstellen	224
37	PN/IE: PNIO	225
37.1	Merkmale.....	225
37.2	Details Kommunikationsart.....	226
37.3	Überblick Anwenderschnittstellen	227
37.4	Anwenderschnittstelle PNIO_SEND, PNIO_RECV	227
37.4.1	Beschreibung	227
37.4.2	Parameter für PNIO_SEND.....	228
37.4.3	Parameter für PNIO_RECV.....	228
37.5	Anwenderschnittstelle DPRD_DAT, DPWR_DAT	229
37.5.1	Beschreibung	229
37.5.2	Parameter für DPRD_DAT	229
37.5.3	Parameter für DPWR_DAT	229
38	PROFIBUS (PB).....	230
38.1	Merkmale.....	230
38.2	Übersicht	230
39	PB: Offene-Kommunikation mit Send/Receive-Bausteinen	231
39.1	Merkmale.....	231
39.2	Details Kommunikationsart.....	232
39.3	Überblick Anwenderschnittstellen	233
39.4	Anwenderschnittstelle: AG_xSEND, AG_xRECV	233
39.4.1	Beschreibung	233
39.4.2	Parameter für AG_SEND, AG_LSEND.....	234
39.4.3	Parameter für AG_RECV, AG_LRECV.....	234
40	PB: FMS-Kommunikation.....	235

40.1	Merkmale	235
40.2	Details Kommunikationsart	236
40.3	Überblick Anwenderschnittstellen	237
40.4	Anwenderschnittstelle: READ, WRITE, REPORT	237
40.4.1	Beschreibung	237
40.4.2	Parameter für READ	238
40.4.3	Parameter für WRITE	238
40.4.4	Parameter für REPORT	238
41	PB: DP-Kommunikation	239
41.1	Merkmale	239
41.2	Details Kommunikationsart	240
41.3	Überblick Anwenderschnittstellen	241
41.4	Anwenderschnittstelle DP_SEND, DP_RECV	241
41.4.1	Beschreibung	241
41.4.2	Parameter für DP_SEND	242
41.4.3	Parameter für DP_RECV	242
41.5	Anwenderschnittstelle DPRD_DAT, DPWR_DAT	243
41.5.1	Beschreibung	243
41.5.2	Parameter für DPRD_DAT	243
41.5.3	Parameter für DPWR_DAT	243
42	Serielle Schnittstelle	244
42.1	Merkmale	244
42.2	Vergleich Protokolle: *ASCII* / 3964(R) / RK 512	244
42.2.1	Abgrenzung	244
42.2.2	Merkmale *ASCII*	245
42.2.3	Merkmale 3964(R)	246
42.2.4	Merkmale RK 512	247
42.2.5	Details Kommunikationsart	248
43	Überblick Anwenderschnittstellen	249
43.1	Einsatz in Stationen mit CPU	249
43.2	Einsatz in dezentralen Stationen ohne CPU	250
44	ET 200S: *ASCII* und 3964(R)	251
44.1	Beschreibung	251
44.2	Parameter für S_SEND	251
44.3	Parameter für S_RCV	251
45	S7-300 CPU: *ASCII* / 3964(R)	252
45.1	Beschreibung	252
45.2	Parameter für SEND_PTP	252
45.3	Parameter für RCV_PTP	252
46	S7 300 CPU: RK 512	253
46.1	Beschreibung	253
46.2	Daten senden	253
46.3	Daten holen	253
46.4	Parameter für SEND_RK	254
46.5	Parameter für SERVE_RK	254
46.6	Parameter für FETCH_RK	255
46.7	Parameter für SERVE_RK	255
47	CP 340: *ASCII* / 3964(R)	256
47.1	Beschreibung	256
47.2	Parameter für P_SEND	256

47.3	Parameter für P_RCV	256
48	CP 341: *ASCII* / 3964(R)	257
48.1	Beschreibung	257
48.2	Parameter für P_SND_RK	257
48.3	Parameter für P_RCV_RK	257
49	CP 341: RK 512	258
49.1	Beschreibung	258
49.2	Daten senden	258
49.3	Daten holen	258
49.4	Parameter für P_SND_RK	259
49.5	Parameter für P_RCV_RK	259
49.6	Parameter für P_SND_RK	260
49.7	Parameter für P_RCV_RK	260
50	CP 440: *ASCII* / 3964(R)	261
50.1	Beschreibung	261
50.2	Parameter für SEND_440	261
50.3	Parameter für REC_440	261
51	CP 441: *ASCII* / 3964(R)	262
51.1	Beschreibung	262
51.2	Parameter für BSEND	263
51.3	Parameter für BRCV	263
52	CP 441: RK 512	264
52.1	Beschreibung	264
52.2	Daten senden	264
52.3	Daten senden	264
52.4	Parameter für BSEND	265
52.5	Parameter für BRCV	266
52.6	Parameter für BSEND	266
52.7	Parameter für PUT	267
52.8	Parameter für GET	267
53	S7-1200: Freeport	268
53.1	Beschreibung	268
53.2	Parameter für SEND_PTP	268
53.3	Parameter für RCV_PTP	268
54	S7-1500, S7-300, S7-400: Freeport / 3964(R)	269
54.1	Beschreibung	269
54.2	Parameter für Send_P2P	269
54.3	Parameter für Receive_P2P	269
55	Informationen zu Teil 3	270
56	***** TEIL 4: Fremde Controller *****	271
56.1	Gliederung und Inhalt	271
56.2	Vorbemerkung	271
57	Modbus/TCP	272
57.1	Merkmale	272
57.2	Überblick Anwenderschnittstellen	272
57.3	S7-1200: Modbus Client	273
57.3.1	Beschreibung	273

57.3.2	Parameter MB_CLIENT	273
57.4	S7-1200: Modbus Server	274
57.4.1	Beschreibung	274
57.4.2	Parameter MB_SERVER	274
58	Modbus Seriell (RTU Format)	275
58.1	Merkmale	275
58.2	Überblick Anwenderschnittstellen	276
58.2.1	Anschluss über CP bzw. CM	276
58.2.2	Modbus Master	276
58.2.3	Modbus Slave	277
58.3	1SI: Modbus Master	278
58.4	1SI: Modbus Slave	278
58.4.1	Beschreibung	278
58.4.2	Parameter S_MODB	278
58.5	CP 341: Modbus Master	279
58.5.1	Beschreibung	279
58.5.2	Parameter P_SND_RK	279
58.5.3	Parameter P_RCV_RK	279
58.6	CP 341: Modbus Slave	280
58.6.1	Beschreibung	280
58.6.2	Parameter FB80	280
58.7	CP 441-2: Modbus Master	281
58.7.1	Beschreibung	281
58.7.2	Parameter BSEND	281
58.7.3	Parameter BRCV	281
58.8	CP 441-2: Modbus Slave	282
58.8.1	Beschreibung	282
58.8.2	Parameter FB180	282
58.9	CM 1241: Modbus Master	283
58.9.1	Beschreibung	283
58.9.2	Parameter MB_MASTER	283
58.10	CM 1241: Modbus Slave	284
58.10.1	Beschreibung	284
58.10.2	Parameter MB_SLAVE	284
58.11	CM PtP / CM PtP HF: Modbus Master	285
58.11.1	Beschreibung	285
58.11.2	Parameter Modbus_Master	285
58.12	CM PtP / CM PtP HF: Modbus Slave	286
58.12.1	Beschreibung	286
58.12.2	Parameter Modbus_Slave	286
59	Informationen zu Teil 4	287
60	***** TEIL 5: Anhang *****	288
60.1	Gliederung und Inhalt	288
61	Literaturangaben	289
62	Begriffe	292
63	Abkürzungen	294
63.1	Im gesamten Dokument	294
63.2	Nur in Tabellen	295
64	Hintergrundwissen	296
64.1	ISO/OSI-Referenzmodell	296
64.2	Kommunikationsmodelle	297

64.2.1	Client und Server	297
64.2.2	Master und Slave	298
64.2.3	Consumer und Provider	298
64.3	Quittierung	299
65	Betrachtete Komponenten	300
65.1	SIMATIC CPU	300
65.2	SIMATIC CP oder CM	302
65.2.1	Einsatz in Stationen mit CPU	302
65.2.2	Einsatz in dezentralen Stationen ohne CPU	303
66	Historie	304
66.1	Versionen	304
66.2	Wesentliche Änderungen	304

1 Vorbemerkungen zum Dokument

1.1 Farbige Textstellen

Zur besseren Orientierung im Dokument ist in einigen Bereichen des Dokuments der **Text blau** dargestellt. Diese **blauen Bereiche** befassen sich entweder mit der Systematik des Dokumentes oder sie enthalten Beispiele zur Anwendung des Dokumentes. Damit soll eine Verwechslung mit den Bereichen ausgeschlossene werden, die ausschließlich die Technik (schwarzer Text) beschreiben.

1.2 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

Kommunikations-aufgabe	Kommunikations-partner	Kommunikation	Daten (Beispiele)	Netz (Beispiele)
Feld- und Prozess-Kommunikation	<ul style="list-style-type: none"> Controller dezentrale Peripherie (Aktor, Sensor) 	innerhalb eines Netzes	Endschalterpositionen Temperaturwerte	PROFINET/ Industrial Ethernet PROFIBUS
Daten-Kommunikation	<ul style="list-style-type: none"> Controller 1 Controller 2 	innerhalb eines Netzes, oder über Netzgrenzen hinweg	Sollwerte Rezepte	PROFINET/ Industrial Ethernet PROFIBUS
IT-Kommunikation	<ul style="list-style-type: none"> Controller PC 	weltweit	E-Mail Datei	PROFINET/ Industrial Ethernet Internet

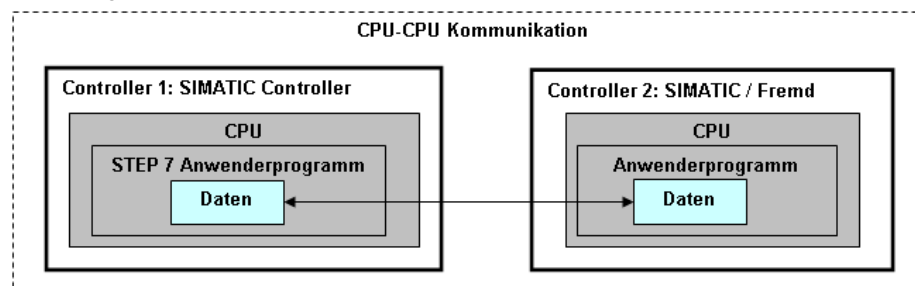
Gegenstand des vorliegenden Dokumentes ist die Daten-Kommunikation zwischen folgenden Kommunikationspartnern:

- 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



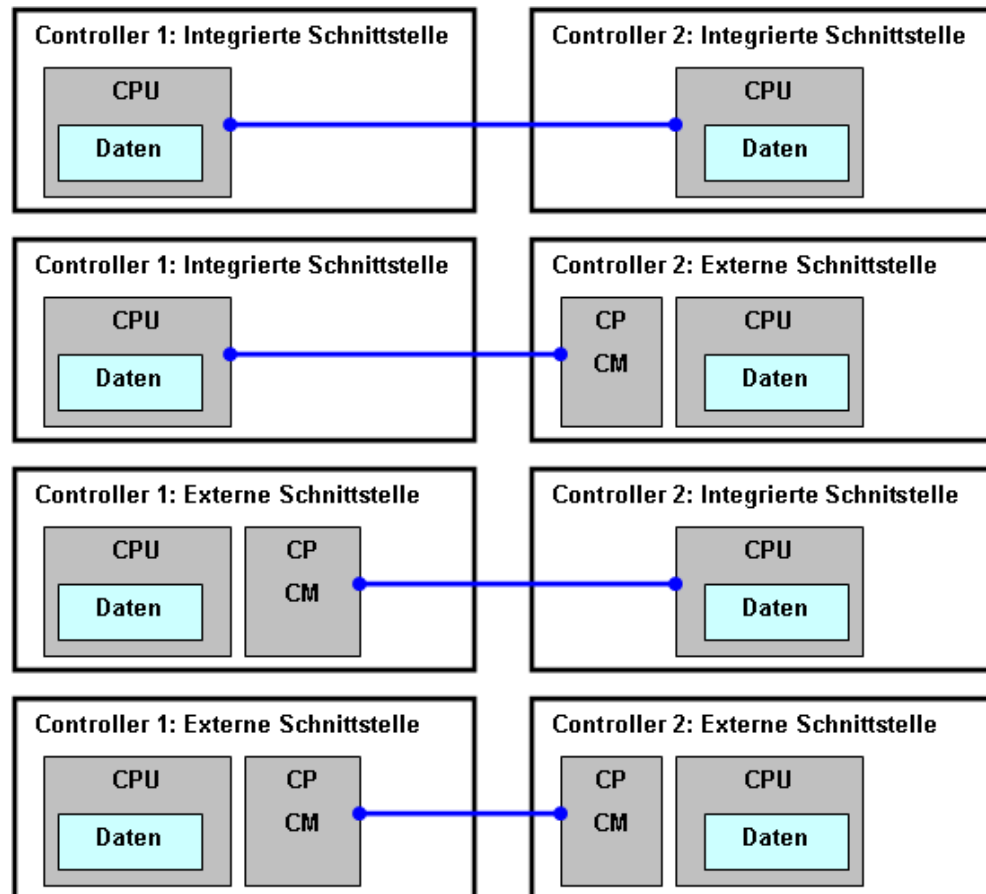
Kommunikationswege

Zur Kommunikation können folgende Schnittstellen verwendet werden:

- integrierte Schnittstelle: Schnittstelle einer SIMATIC CPU
- externe Schnittstelle: Schnittstelle eines SIMATIC CPs oder CMs

Das folgende Bild zeigt schematisch die im Dokument betrachteten Kombinationen der Schnittstellen von Controller 1 und Controller 2.

Abbildung 1-2



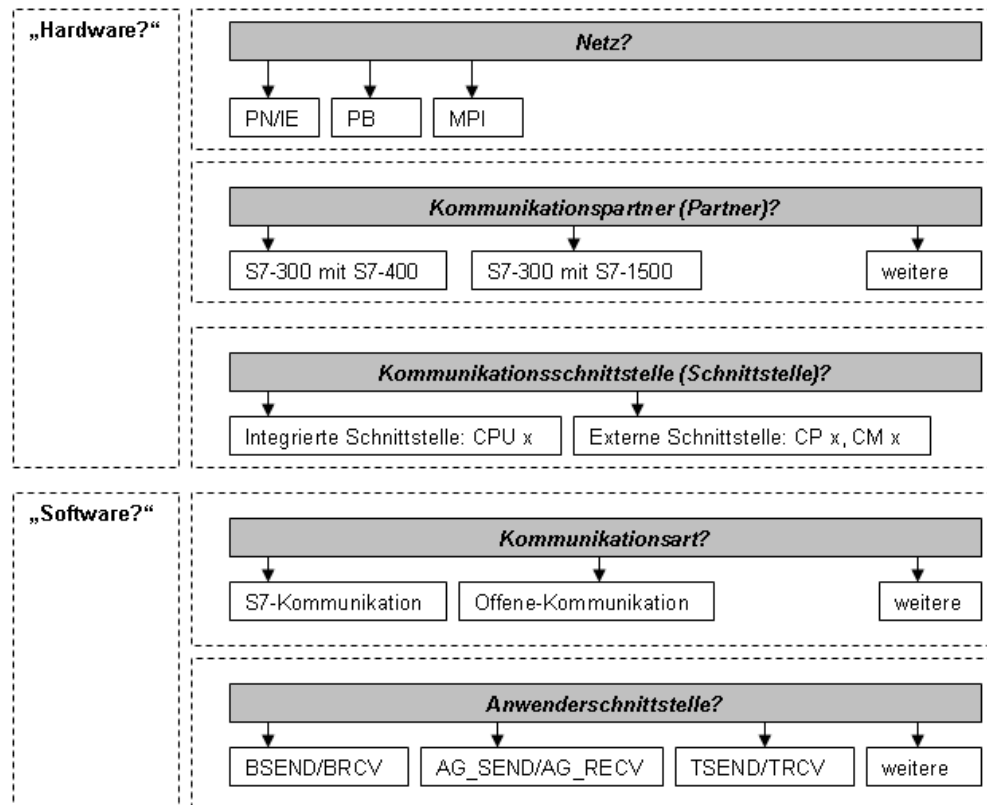
1.3 Veranlassung und Zielsetzung des Dokumentes

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?

1.4 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 der Kommunikationsbausteine (BSEND, TSEND, ...) nicht beschrieben (*1).

Zu (*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 Informationen (Handbuch, Applikationsbeispiel, FAQ, ...)

1.5 Gültigkeitsbereich des Dokumentes

Alle Aussagen im Dokument beziehen sich ausschließlich auf die aktuellsten (mit Lieferfreigabe versehenen) Controller der SIMATIC:

- **Stand März 2013**

Im Dokument werden folgende Möglichkeiten der Kommunikation nicht betrachtet:

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

Im Dokument werden folgende Komponenten nicht betrachtet:

- SIMATIC Controller LOGO!
- H-CPU's, T-CPU's
- Komponenten mit Auslaferklärung

1.6 Abdeckung des Dokumentes

Überblick

Medium:

- Netz: PN/IE, PB, MPI
- SIMATIC Rückwandbus
- Serielle Schnittstelle (PtP)

SIMATIC Familien:

- SIMATIC Modulare Controller:
ET 200 CPU, S7-300, S7-400, S7-1200, S7-1500
- SIMATIC PC-based Controller:
WinAC RTX, S7-mEC, Box PC, Panel PC

Kommunikationsarten:

- SIMATIC spezifisch
- Offener Standard

Komponenten

Im Kapitel 65 ist eine Übersicht aller Komponenten (CPU, CP, CM) zu finden, die im Dokument berücksichtigt wurden.

Berücksichtigung der Versionen von STEP 7

Wenn Aussagen im Dokument davon abhängen, welche Variante des Engineering Tools benutzt wird, dann wird dies im Dokument wie folgt gekennzeichnet:

- "STEP 7": STEP 7 bis V5.5 und/oder ab V10
- "STEP 7 (nicht TIA)": STEP 7 nur bis V5.5
- "STEP 7 (TIA)": STEP 7 nur ab V10

1.7 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, ...)	<u>/0/</u>
Applikationsbeispiele zur Kommunikation	Lösungen zu konkreten Aufgabenstellungen (Dokumentation und STEP 7 Projekt)	<u>/200//200//200 /</u>
FAQs zur Kommunikation	Antworten auf häufig gestellte Fragen	<u>/0/</u>
Systemhandbuch Kommunikation mit SIMATIC	Basiswissen zur industriellen Kommunikation	<u>/3/</u>
Katalog Produkte für Totally Integrated Automation und Micro Automation	Bestellunterlage für SIMATIC Controller	<u>/4/</u>
Katalog Industrielle Kommunikation	Bestellunterlage für SIMATIC Net Produkt	<u>/5/</u>

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 von Teil 1 steht ein Kapitel mit Hinweisen auf weitere Informationen.
Details zum Inhalt von Teil 1: Siehe Kapitel 4.1.

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 (S7-300, ...):
 - Welche Schnittstellen (CPU, CP, CM) 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 von Teil 2 steht ein Kapitel mit Hinweisen auf weitere Informationen.
Details zum Inhalt von Teil 2: Siehe Kapitel 18.1.

2.3 TEIL 3: Kommunikationsarten

Ziel

Teil 3 dient der Vertiefung:

- Detaillierte Informationen zu allen Kommunikationsarten

Inhalt

Für jede Kommunikationsart (S7-Kommunikation, ...) wird beschrieben:

- Merkmale
- Eigenschaften (Tabelle mit einheitlichen Kriterien)
- Anwenderschnittstelle (Kommunikationsbausteine, ...)

Am Ende von Teil 3 steht ein Kapitel mit Hinweisen auf weitere Informationen.
Details zum Inhalt von Teil 3: Siehe 27.1.

2.4 TEIL 4: Fremde Controller

Teil 4 beschreibt Beispiele für die Kommunikation über offengelegte Protokolle zwischen den Partnern:

- SIMATIC Controller
- Controller von Fremdherstellern (fremde Controller)

Beispiel: Kommunikation zu fremden Controllern über Modbus/TCP

Am Ende von Teil 4 steht ein Kapitel mit Hinweisen auf weitere Informationen.
Details zum Inhalt von Teil 4: Siehe Kapitel 56.1.

2.5 TEIL 5: Anhang

Inhalt von Teil 5:

- Literaturangaben
- Begriffe und Abkürzungen
- Hintergrundwissen zu ausgesuchten Themen
- Übersicht der betrachteten SIMATIC Komponenten (CPU, CP, CM)
- Historie des Dokumentes

Details zum Inhalt von Teil 5: Siehe Kapitel 60.1.

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 (Kapitel 20). 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 61 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 62 werden wichtige Begriffe erläutert. Das Kapitel 63 enthält eine Beschreibung wichtiger Abkürzungen.

Hintergrundwissen

Im Kapitel 64 werden wichtige Zusammenhänge erläutert.

3.2 Beispiel zur Anwendung des Dokumentes

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

Ziel des Kapitels ist es, das Prinzip zu zeigen. Deswegen werden im Kapitel keine Details erläutert.

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 folgenden 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

Mit Hilfe des Sprungverteilers für das Netz PN/IE (Kapitel 20.1) können die obigen Fragen leicht beantwortet werden.

Dies wird in den folgenden Kapiteln gezeigt.

3.2.2 Sprungverteiler

Für jedes Medium (PNIE, PB, MPI, ...) gibt es im Dokument einen sogenannten Sprungverteiler. Der Sprungverteiler enthält Querverweise (grau hinterlegt) zu Informationen im Dokument.

Die folgende Abbildung zeigt den Sprungverteiler für PN/IE.

Abbildung 3-1

20.1 Sprungverteiler: Netz PN/IE

1

20.1.1 Übersicht Schnittstellen und Kommunikationsarten

Tabelle 20-1 Links zu den Tabellen Schnittstellen

Familie	Kapitel
ET 200 CPU	21.2.1
S7-300	21.2.2
S7-400	21.2.3
S7-1200	21.2.4
S7-1500	21.2.5
S7-mEC	21.2.6
Box PC	21.2.7
Panel PC	21.2.8
WinAC RTX	21.2.9

2

20.1.2 Kombinationen Controller 1 / Controller 2

Tabelle 20-2 Links zu den Tabellen Kombinationen

Controller 2	Controller 1					
	ET 200 CPU	S7-300	S7-400	S7-1200	S7-1500	WinAC RTX
ET 200 CPU	21.3.1	21.3.2	21.3.3	21.3.4	21.3.5	21.3.6
S7-300	21.3.2	21.4.2	21.4.3	21.4.4	21.4.5	21.4.6
S7-400	21.3.3	21.4.3	21.5.3	21.5.4	21.5.5	21.5.6
S7-1200	21.3.4	21.4.4	21.5.4	21.6.4	21.6.5	21.6.6
S7-1500	21.3.5	21.4.5	21.5.5	21.6.5	21.7.5	21.7.6
WinAC RTX	21.3.6	21.4.6	21.5.6	21.6.6	21.7.6	21.8.6

Hinweis: Die SIMATIC Controller S7-mEC, Box PC und Panel PC verhalten sich bezüglich der Kommunikation wie WinAC RTX.

3

20.1.3 Kommunikationsarten

Tabelle 20-3 Links zu den Tabellen Kommunikationsarten

Kommunikationsart		Kapitel
Alle Kommunikationsarten (Tabelle Kompakt)		21.9
Tabelle mit Details	S7-Kommunikation	32.2
	Offene-Kommunikation mit Send/Receive-Bausteinen	34.2
	Offene-Kommunikation mit T-Bausteinen	35.2
	PNIO	37.2
Modbus/TCP (SIMATIC / fremder Controller)		59

Der Sprungverteiler besteht aus den Abschnitten:

- (1): Übersicht Schnittstellen und Kommunikationsarten
- (2): Kombinationen Controller 1 / Controller 2
- (3): Kommunikationsarten

In den folgenden Kapiteln wird die Anwendung der Abschnitte gezeigt.

3.2.3 Übersicht Schnittstellen und Kommunikationsarten

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

Abbildung 3-2

Tabelle 20-1 Links zu den Tabellen Schnittstellen

Familie	Kapitel
ET 200 CPU	21.2.1
S7-300	21.2.2
S7-400	21.2.3
S7-1200	21.2.4
S7-1500	21.2.5
S7-mEC	21.2.6
Box PC	21.2.7
Panel PC	21.2.8
WinAC RTX	21.2.9

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.

Abbildung 3-3

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

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

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

(2) Lade/Transfer Befehle, DPF
 (6) PNIO_SEND, PNIO_RECV
 (13) Server für Fetch, Write (*1)

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

- Mit dem CP 343-1 kann eine 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/URCV", BSEND/BRCV, "PUT, GET"
- Zur Verfügung stehenden Protokolle: ISO on TCP, ISO

Hinweis: An dieser Stelle wird nur das Prinzip der Tabelle gezeigt. Eine ausführliche Beschreibung des Aufbaus der Tabelle ist im Kapitel (19.2) zu finden.

3.2.4 Kombinationen Controller 1 / Controller 2

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

Abbildung 3-4

Controller 2	Controller 1					
	ET 200 CPU	S7-300	S7-400	S7-1200	S7-1500	WinAC RTX
ET 200 CPU	21.3.1	21.3.2	21.3.3	21.3.4	21.3.5	21.3.6
S7-300	21.3.2	21.4.2	21.4.3	21.4.4	21.4.5	21.4.6
S7-400	21.3.3	21.4.3	21.5.3	21.5.4	21.5.5	21.5.6
S7-1200	21.3.4	21.4.4	21.5.4	21.6.4	21.6.5	21.6.6
S7-1500	21.3.5	21.4.5	21.5.5	21.6.5	21.7.5	21.7.6
WinAC RTX	21.3.6	21.4.6	21.5.6	21.6.6	21.7.6	21.8.6

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 dort die möglichen Kommunikationsarten eingetragen. Die folgende Abbildung zeigt einen Ausschnitt dieser Tabelle für die Familien S7-300 und S7-400.

Abbildung 3-5

Controller 2: S7-400				Controller 1: S7-300 an PN/IE							
				CPU				CP			
				Schnittstelle: PN				343-1 Lean			
				IOC, IOD				IOD			
				S7	OC		PII		S7	OC	
						PNIO	CBA			PNIO	CBA
CPU	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	X: PN	IOC, IOD	(1)	(3)/(4)+(9)	(2)	x	(21)	(8)/(4)+(9)	(6)/(2)	---
		X: IE	---	(1)	(3)/(4)+(9)	---	---	(21)	(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
- (8) AG_SEND/AG_RECV
- (9) AG_SEND/AG_RECV, AG_LSEND/AG_LRCV, AG_SSEND/AG_SRCV
- (21) Controller 1 ist Server (für "PUT, GET")

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

- Alle S7-300 CPUs mit PN Schnittstelle (Controller 1) können mit dem CP 443-1 Advanced (Controller 2) kommunizieren.
- Mögliche Kommunikationsarten über die PN Schnittstelle des CPs:
 - S7-Kommunikation (S7)
 - Offene-Kommunikation (OC)
 - PN-Kommunikation (PN),
- Mögliche Kommunikationsarten über die IE Schnittstelle des CPs:
 - S7-Kommunikation (S7)
 - Offene-Kommunikation (OC)
- Mögliche Kommunikationsbausteine bei S7-Kommunikation:
Controller 1 und Controller 2: Kommunikationsbausteine (1)
- Mögliche Kommunikationsbausteine bei Offene-Kommunikation:
 - Controller 1: Kommunikationsbausteine (3)
 - Controller 2: Kommunikationsbausteine (4) und (9)

- Mögliche Kommunikationsarten bei PN-Kommunikation:
 - PNIO mit Kommunikationsbausteinen (2)
 - CBA

Hinweis: An dieser Stelle soll nur das Prinzip der Tabelle gezeigt werden. Eine ausführliche Beschreibung des Aufbaus der Tabelle ist im Kapitel (19.3) zu finden.

3.2.5 Kommunikationsarten

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

Abbildung 3-6

Tabelle 20-3 Links zu den Tabellen Kommunikationsarten

Kommunikationsart		Kapitel
Alle Kommunikationsarten (Tabelle Kompakt)		21.9
Tabelle mit Details	S7-Kommunikation	32.2
	Offene-Kommunikation mit Send/Receive-Bausteinen	34.2
	Offene-Kommunikation mit T-Bausteinen	35.2
	PNIO	37.2
Modbus/TCP (SIMATIC / fremder Controller)		59

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.

Abbildung 3-7

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

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 Kommunikationsart S7-Kommunikation. Die folgende Abbildung zeigt die Tabelle.

Abbildung 3-8

Kommunikationsart:		S7-Kommunikation		
Protokoll:		S7-Protokoll (*1)		
Allgemeines				
Medien		MPI, PB, PN/IE, Rückwandbus (*3)		
Schnittstellen		CPU, CP, CM		
Anbindung	SIMATIC S5	nein		
	Fremd (offene Standards)	nein		
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		
Anwenderschnittstelle				
Kommunikationsbausteine		BSEND / BRCV	Typ "USEND / URCV"	Typ "PUT, GET"
maximale Anzahl Daten (*2)		<= 64 KByte	>= 160 Byte	>= 160 Byte
dynamische Adressierung Daten		S7-300: ja	S7-300: ja	S7-300: ja
		sonst: nein	sonst: nein	sonst: nein
remote Quittierung		Applikation	Transport	Applikation
Modell		Client / Client	Client / Client	Client / Server

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

Beispiel: Parameter des Kommunikationsbausteines BSEND

Abbildung 3-9

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
REQ	BOOL	BOOL	Anstoß Sendeauftrag
R	BOOL	BOOL	Abbruch Sendeauftrag
ID	WORD	CONN_PRG	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektion in STEP 7)
R_ID	DWORD	CONN_R_ID	Zuordnung Sende SFB/FB und Empfangs SFB/FB. Dies ermöglicht die Kommunikation mehrerer SFB/FB Paare über dieselbe logische Verbindung.
OUTPUT			Bemerkung
DONE	BOOL	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	BOOL	Fehlerinformation
STATUS	WORD	WORD	
IN_OUT			Bemerkung
SD_1	ANY	VARIANT	Sendebereich
LEN	WORD	WORD	Länge der zu sendenden Daten

3.2.6 Zusammenfassung

Das folgende Bild zeigt, am Beispiel für das Netz PN/IE, wie der Sprungverteiler im Dokument wirkt.

Im Sprungverteiler stehen Querverweise zu allen Kapiteln, die für eine CPU-CPU Kommunikation über PN/IE von Interesse sein können:

- (1): Querverweis auf die Tabelle Schnittstellen
- (2): Querverweis auf die Tabelle Kombinationen
- (3): Querverweis auf die Tabelle Kommunikationsarten-Kompakt
- (4): Querverweis auf die Tabelle Kommunikationsarten-Detail
- (5): Querverweis auf die Beschreibung

Abbildung 3-10

TEIL 2	Sprung- verteiler für:	Netz PN/IE	Übersicht Schnittstellen und Kommunikationsarten		1
			Kombination Controller 1 / Controller 2		2
			Kommunikationsarten	Übersicht (alle)	3
				Details (Einzel)	4
				Fremde Controller	5
		Netz PB	wie oben		
		Netz MPI	wie oben		
		SIMATIC Rückwandbus	wie oben		
		Serielle Schnittstellen	wie oben		
	Auswahl- hilfen für:	PN/IE	Tabelle Schnittstellen		
			Tabelle Kombinationen		
			Tabelle Kommunikationsarten-Kompakt		
		PB	wie oben		
		MPI	wie oben		
		SIMATIC Rückwandbus	wie oben		
		Serielle Schnittstellen	wie oben		
TEIL 3	Kommuni- kationsarten:	S7-Kommunikation	Tabelle Kommunikationsarten-Detail		
			Anwenderschnittstellen (FBs, FCs, ...)		
		... weitere ...	wie oben		
TEIL 4	Fremde Controller:	Modbus/TCP	Beschreibung		
		Modbus Seriell	wie oben		

4 ***** TEIL 1: Einführung *****

4.1 Gliederung und Inhalt

Tabelle 4-1

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

5 Modelle zur CPU-CPU Kommunikation

5.1 Definition Controller

Im Dokument wird die folgende Definition verwendet:

Ein Controller ist eine zentrale oder dezentrale Automatisierungsstation (Station) mit den Komponenten CPU, CP (optional), CM (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

5.2 Definition CPU-CPU Kommunikation

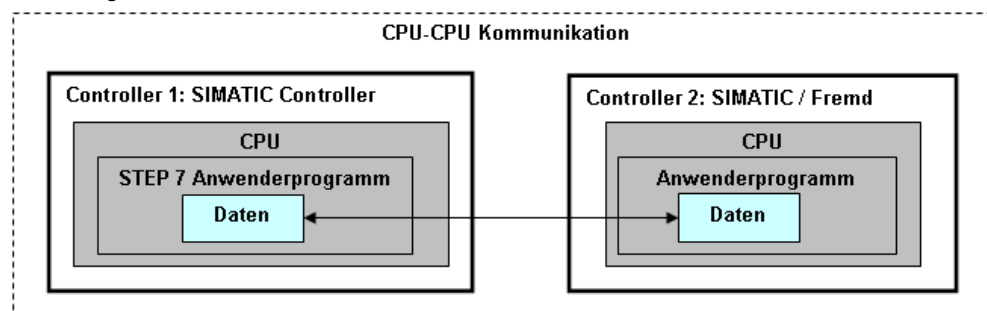
Bei der CPU-CPU Kommunikation werden Daten zwischen den CPUs zweier Controllern ausgetauscht:

- 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 5-1



Bei der CPU-CPU Kommunikation werden folgende Fälle unterschieden:

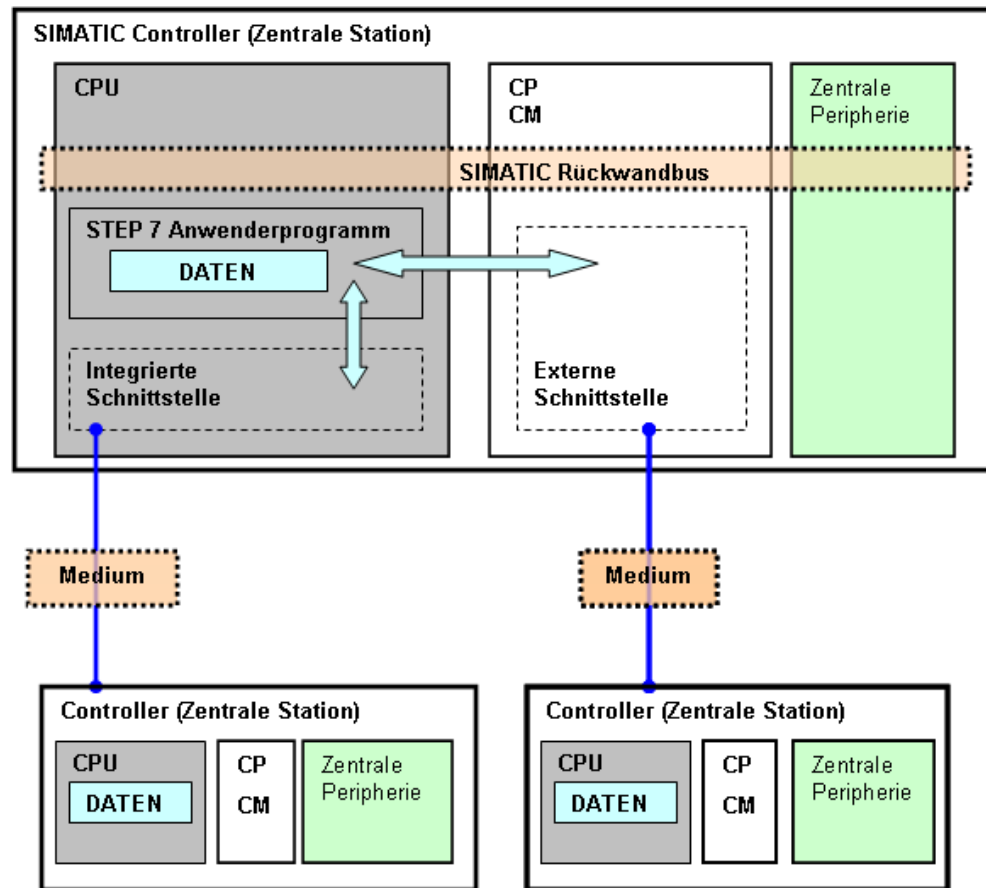
- CPUs in unterschiedlichen zentralen Stationen
- CPUs in zentraler und dezentraler Station (*1)
- CPUs innerhalb einer zentralen SIMATIC Station

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

5.3 CPUs in unterschiedlichen zentralen Stationen

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

Abbildung 5-2



Schnittstellen zur Kommunikation:

- integrierte Schnittstelle: Schnittstelle auf CPU
- externe Schnittstelle: Schnittstelle auf CP oder CM

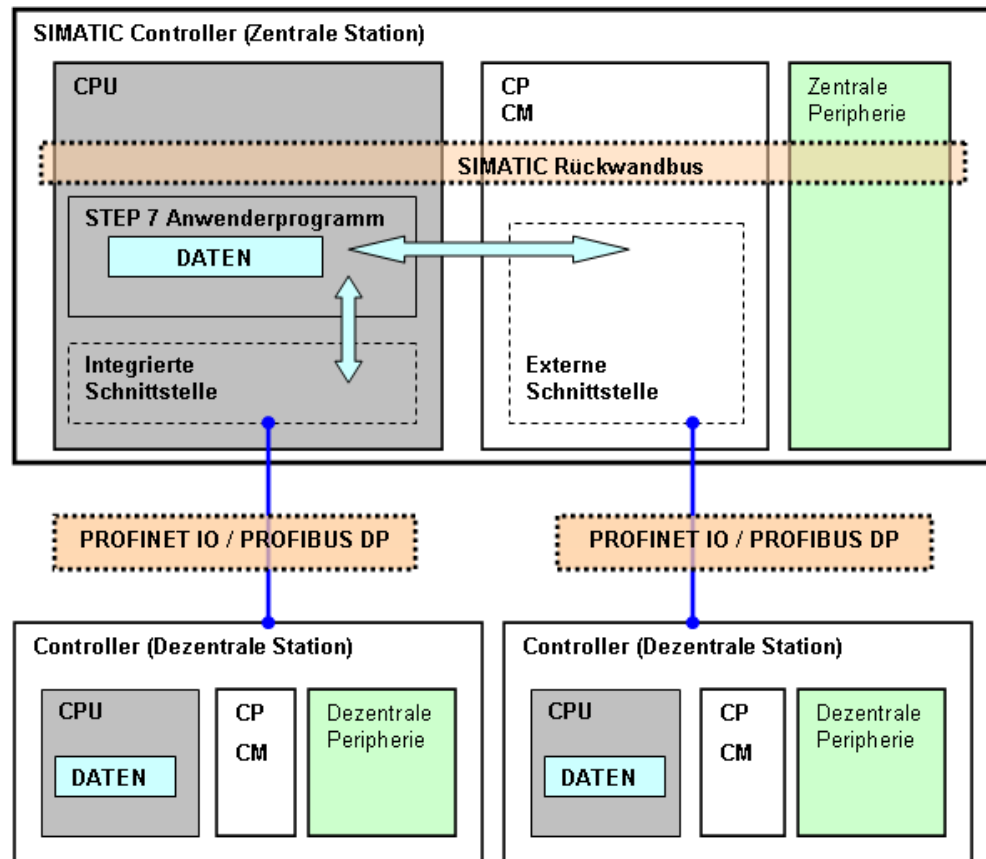
Medien zur Kommunikation:

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

5.4 CPUs in zentraler und dezentraler Station

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

Abbildung 5-3



Schnittstellen zur Kommunikation:

- integrierte Schnittstelle: Schnittstelle auf CPU
- externe Schnittstelle: Schnittstelle auf CP oder CM

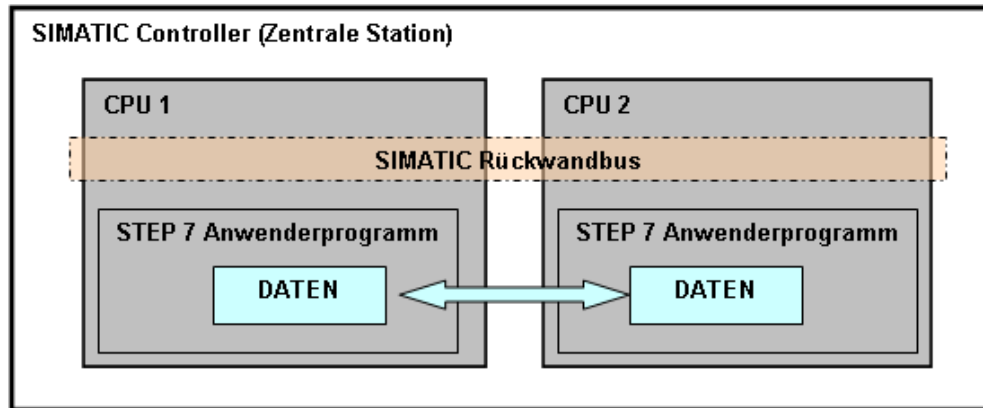
Medien zur Kommunikation:

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

5.5 CPUs innerhalb einer zentralen Station

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

Abbildung 5-4



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-CPUs in einer zentralen SIMATIC Station gleichzeitig betrieben werden.

6 Verbindungen bei SIMATIC

6.1 Verbindungen

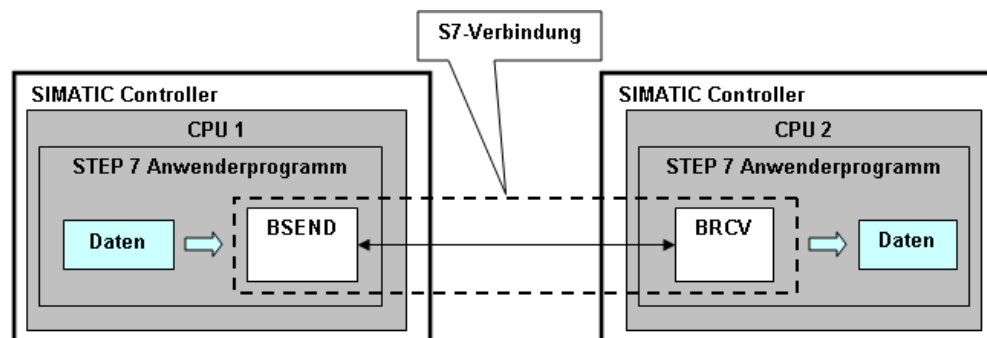
Einführung

CPU-CPU Kommunikation erfolgt bei der SIMATIC im Allgemeinen über Verbindungen. Eine Verbindung legt fest, wo die Endpunkte der Kommunikation liegen.

Das folgende Bild zeigt ein Beispiel einer Verbindung:

CPU 1 kommuniziert mit CPU 2 über eine S7-Verbindung.

Abbildung 6-1



Verbindungen müssen eingerichtet werden (Kapitel 6.4). Dies kann automatisch, durch Projektierung oder durch Programmierung erfolgen.

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 externe Schnittstelle eines CP bzw. CM) 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 bzw. CM folgende Aufgaben ausgeführt:

- Flusssteuerung (Vermeidung von Überlast in den Partnern, ...)
- Überwachung der Verbindung (Prüfung, ob Partner noch erreichbar ist, ...)
- Austausch von Quittungen (Bestätigung, dass Daten fehlerfrei angekommen sind, ...)

6.2 Protokolle

Ein Protokoll legt fest, nach welchen Regeln die Kommunikation zwischen zwei Kommunikationspartnern abläuft. Zur Einordnung eines Protokolls wird häufig das ISO/OSI-Referenzmodell verwendet (Kapitel 64.1). Bei den Transport-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 Kommunikation zwischen den Steuerungsprogrammen zweier CPUs liegt.

Verbindungslose Protokolle

Diese Protokolle bauen zwischen beiden Kommunikationspartnern keine Verbindung auf (Beispiel: UDP). Die Protokolle werden eingesetzt, wenn die Priorität bei einer schnellen Kommunikation zwischen den Steuerungsprogrammen zweier CPUs liegt.

6.3 Verbindungsressourcen

Bedeutung

Verbindungen belegen Ressourcen auf CPU, CP oder CM (Beispiel: Pufferspeicher für die Empfangsdaten). Diese Ressourcen werden bei der SIMATIC mit „Verbindungsressourcen“ bezeichnet.

Die Anzahl der maximal möglichen Verbindungen pro CPU, CP oder CM ist begrenzt. Sie hängt von den zur Verfügung stehenden Verbindungsressourcen von CPU, CP oder CM ab. In den Handbüchern der Baugruppen ist deswegen in den technischen Daten angegeben, wie viele Verbindungen pro Kommunikationsart maximal möglich sind. Das folgende Bild zeigt dazu ein Beispiel aus dem Handbuch S7-300.

Abbildung 6-2

<i>Technical specifications of CPU 31x</i>	
<i>9.7 CPU 317-2 PN/DP</i>	
Technical specifications	
TCP/IP	Yes (via integrated PROFINET interface and loadable FBs)
• Maximum number of connections	16
• Data length for connection type 01H, max.	1460 bytes
• Data length for connection type 11H, max.	32768 bytes
• Multiple passive connections per port (multiport), Supported	Yes
ISO on TCP	Yes (via integrated PROFINET interface and loadable FBs)
• Maximum number of connections	16
• Data length, max.	32768 bytes
UDP	Yes (via integrated PROFINET interface and loadable FBs)
• Maximum number of connections	16
• Data length, max.	1472 bytes

Belegung und Freigabe

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

Werden Verbindungen in STEP 7 projiziert, 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 Zahl der verwendeten Verbindungen selbst verwalten. D.h. der Anwender muss sicherstellen, dass die Anzahl maximal möglicher Verbindungen, der an der Kommunikation beteiligten Baugruppen, eingehalten wird.

6.4 Einrichten von Verbindungen

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

- projizierte Verbindung
- nicht projizierte Verbindung

6.4.1 Projektierte Verbindung

Projizierte Verbindungen sind Verbindungen, die in der Hardware-Konfiguration von STEP 7 projiziert werden. Projektierte Verbindungen werden zum Beispiel bei folgenden Kommunikationsarten verwendet:

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

Schritte zur Realisierung einer Kommunikation:

- Projektierung mit STEP 7 (Hardware-Konfiguration):
Verbindung zwischen den Kommunikationspartnern
- Programmierung in STEP 7 (Anwenderprogramm):
Aufruf Kommunikationsbaustein (zur Datenübertragung)

Aufbau, Abbau und Abbruch einer Verbindung:

- Aufbau: Automatischer Aufbau der Verbindung beim Hochlauf der kommunikationsfähigen Baugruppen (CPU, CP, CM). Mit dem Aufbau werden erforderliche Verbindungsressourcen vom Betriebssystem der Baugruppen belegt.
- Abbau: Die Verbindung wird nach der Datenübertragung nicht abgebaut. D.h. die Verbindungsressourcen bleiben dauerhaft belegt.
- Abbruch (z.B. wegen einer Störung): Nach Abbruch wird die Verbindung durch das Betriebssystem automatisch wieder aufgebaut. Eventuell freigewordene Verbindungsressourcen werden wieder belegt.

6.4.2 Nicht projektierte Verbindung

Nicht projektierte Verbindungen sind Verbindungen, die nicht in der Hardware-Konfiguration von STEP 7 projektiert werden.

Es sind die beiden Fälle zu unterscheiden:

- Automatisches Einrichten einer Verbindung (Beispiel: S7-Basiskommunikation)
- Programmiertes Einrichten einer Verbindung (Beispiel: Offene-Kommunikation)

Automatisches Einrichten einer Verbindung

Schritt zur Realisierung einer Kommunikation:

- Programmierung in STEP 7 (Anwenderprogramm):
Aufruf Kommunikationsbaustein (zur Datenübertragung)

Aufbau, Abbau und Abbruch einer Verbindung:

- Aufbau: Bei erstmaligem Aufruf des Kommunikationsbausteines erfolgt ein Verbindungsaufbau durch das Betriebssystem der Baugruppen (CPU, CP, CM). 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.
- Abbruch (z.B. wegen einer Störung): Für das Verhalten nach Abbruch einer Verbindung gibt es zwei Möglichkeiten:
 - Die Verbindung wird vom Betriebssystem so lange wieder aufgebaut, bis der Abbau der Verbindung durch das Steuerungsprogramm veranlasst wird.
 - Das Steuerungsprogramm erkennt den Abbruch, und richtet die Verbindung erneut ein.

Programmiertes Einrichten einer Verbindung

Hier gibt es zwei unterschiedliche Möglichkeiten: Verbindungsmanagement mit Verbindungsbausteinen oder mit Kommunikationsbausteinen.

Verbindungsmanagement mit **Verbindungsbausteinen**

Schritte zur Realisierung einer Kommunikation:

- Programmierung in STEP 7 (Anwenderprogramm):
Aufruf Verbindungsbaustein (Beispiel: T_CONFIG)
- Programmierung in STEP 7 (Anwenderprogramm):
Aufruf Kommunikationsbaustein (zur Datenübertragung, Beispiel TSEND)

Aufbau, Abbau und Abbruch der Verbindung:

- Aufbau: Bei Aufruf des Verbindungsbausteines erfolgt ein Verbindungsaufbau durch das Betriebssystem der Baugruppen (CPU, CP, CM).
- Abbau: Die Verbindungen können durch Aufruf eines Verbindungsbausteines abgebaut werden. Bei Verbindungsabbau werden Verbindungsressourcen wieder freigegeben.
- Abbruch (z.B. wegen einer Störung): Für das Verhalten nach Abbruch einer Verbindung gibt es zwei Möglichkeiten:
 - Die Verbindung wird vom Betriebssystem so lange wieder aufgebaut, bis der Abbau der Verbindung durch das Steuerungsprogramm veranlasst wird.
 - Das Steuerungsprogramm erkennt den Abbruch, und richtet die Verbindung erneut ein.

Verbindungsmanagement mit **Kommunikationsbausteinen**

Schritte zur Realisierung einer Kommunikation:

- Programmierung in STEP 7 (Anwenderprogramm):
Aufruf Kommunikationsbaustein (zum Verbindungsmanagement und zur Datenübertragung) (Beispiel: T_SEND_C)
- Programmierung in STEP 7 (Anwenderprogramm):
Parametrierung Kommunikationsbaustein (Festlegung Verbindungsparameter)

Aufbau, Abbau und Abbruch der Verbindung:

- Aufbau: Bei Aufruf des Kommunikationsbausteines erfolgt Verbindungsaufbau durch das Betriebssystem der Baugruppen (CPU, CP, CM).
- 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.
- Abbruch (z.B. wegen einer Störung): Für das Verhalten nach Abbruch einer Verbindung gibt es zwei Möglichkeiten:
 - Die Verbindung wird vom Betriebssystem so lange wieder aufgebaut, bis der Abbau der Verbindung durch das Steuerungsprogramm veranlasst wird.
 - Das Steuerungsprogramm erkennt den Abbruch, und richtet die Verbindung erneut ein.

7 Datenkonsistenz bei SIMATIC

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

7.1 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 dann zu einem Zeitpunkt teilweise aus neuen und teilweise aus alten 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 im Beispiel 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

Systembedingte Datenkonsistenz

Vom Betriebssystem einer CPU oder eines CP wird Datenkonsistenz für eine maximale Anzahl von Daten ("Anzahl konsistente Daten") garantiert. Für Datenbereiche, die kleiner sind als "Anzahl konsistente Daten" sind im STEP 7 Anwenderprogramm keine Maßnahmen erforderlich, um Datenkonsistenz zu gewährleisten.

Weitergehende Datenkonsistenz

Wenn Datenkonsistenz für einen Datenbereich benötigt wird, der größer ist als "Anzahl konsistente Daten", dann kann dies durch Zusatzmaßnahmen im STEP 7 Anwenderprogramm erreicht werden (Beispiel: Prozessalarm OB sperren für die Zeitdauer der Datenübertragung).

7.2 Systembedingte Datenkonsistenz

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

- Typ CPU (falls Kommunikation mit der CPU)
- Typ CP bzw. CM und Typ CPU (falls Kommunikation mit Unterstützung eines CP bzw. CM)
- Typ Kommunikationsbaustein

Es folgen Aussagen zu den verschiedenen SIMATIC Familien.

S7-300

Bei S7-300 werden die Daten in Blöcken zu x Byte (siehe Tabelle 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 7-1

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

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

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 garantiert.

S7-1200

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

S7-1500

Die CPU garantiert die Datenkonsistenz für eine Variable.

7.3 Weitergehende Datenkonsistenz

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).

Fallunterscheidung

Es sind zwei Fälle zu unterscheiden:

Client Client Kommunikation

Beispiele für Kommunikationsbausteine: BSEND / BRCV

Soll weitergehende Datenkonsistenz gewährleistet werden, dann dürfen während der Übertragung die Daten nicht verändert werden.

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.

8 SIMATIC Controller

Es folgt eine Übersicht der im Dokument betrachteten SIMATIC Familien ([1/](#)).

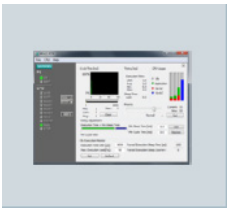
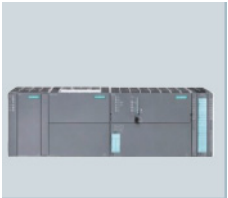

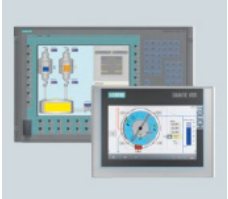
SIMATIC Modular Controller

Tabelle 8-1

SIMATIC Familie	
ET 200 CPU Modulares, dezentrales Peripheriesystem mit Intelligenz vor Ort	
S7-300 Modularer Controller in der Fertigungsindustrie	
S7-400 Modularer Controller für Fertigungs- und Prozessindustrie	
S7-1200 Modularer, kompakter Controller	
S7-1500 Modularer Controller für Fertigungs- und Prozessindustrie	

SIMATIC PC-based Controller

Tabelle 8-2

SIMATIC Familie	
WinAC RTX Software Controller	
S7-modular Embedded Controller (S7-mEC) S7-300 Aufbautechnik	
Embedded Box PC-Bundles (Box PC) Einschaltfertiger Hutschienen-PC (Bundles mit WINAC RTX)	
Embedded Panel PC-Bundles (Panel PC) Einschaltfertiger Panel PC (Bundles mit WINAC RTX)	

9 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 9-1

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

In den folgenden Kapiteln werden die Medien nur kurz charakterisiert. Details zu den Medien können in der Literatur nachgelesen werden (Kapitel 17).

Übersicht der folgenden Kapitel:

Tabelle 9-2

Medien zur SIMATIC Kommunikation		Kapitel
PROFINET/Industrial Ethernet (PN/IE)		10
	Vorbemerkung	10.1
	Ethernet	10.2
	Industrial Ethernet (IE)	10.3
	PROFINET (PN)	10.4
PROFIBUS (PB)		11
MPI		12
SIMATIC Rückwandbus		13
Serielle Schnittstelle (PtP)		14
Gegenüberstellung der Medien		15
Schnittstellen der SIMATIC Familien		16

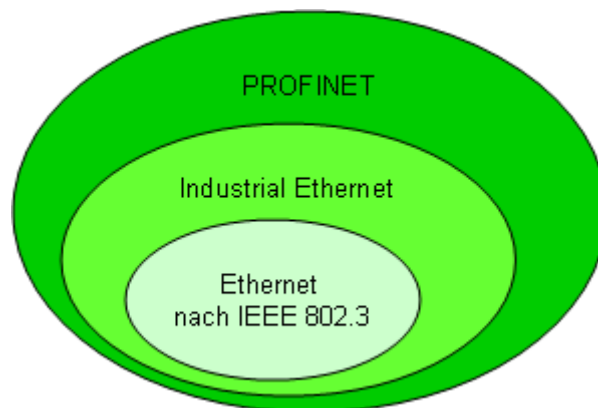
10 PROFINET/Industrial Ethernet (PN/IE)

10.1 Vorbemerkung

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

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

Abbildung 10-1



In den folgenden Kapiteln werden Ethernet, Industrial Ethernet und PROFINET kurz charakterisiert.

10.2 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)

Zu (*1): 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:

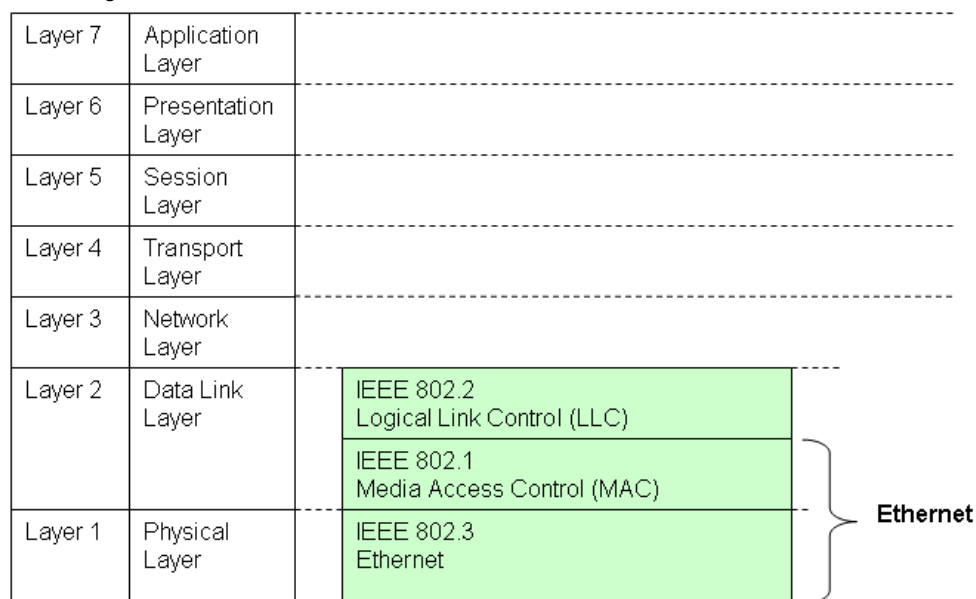
- als eigenständige Komponente (SCALANCE X)
- integriert in SIMATIC Komponenten (CPU, CP bzw. CM mit PN/IE Anschluss)

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 10-2



10.3 Industrial Ethernet (IE)

Einführung

IE ist die industrietaugliche Variante von Ethernet.

Merkmale

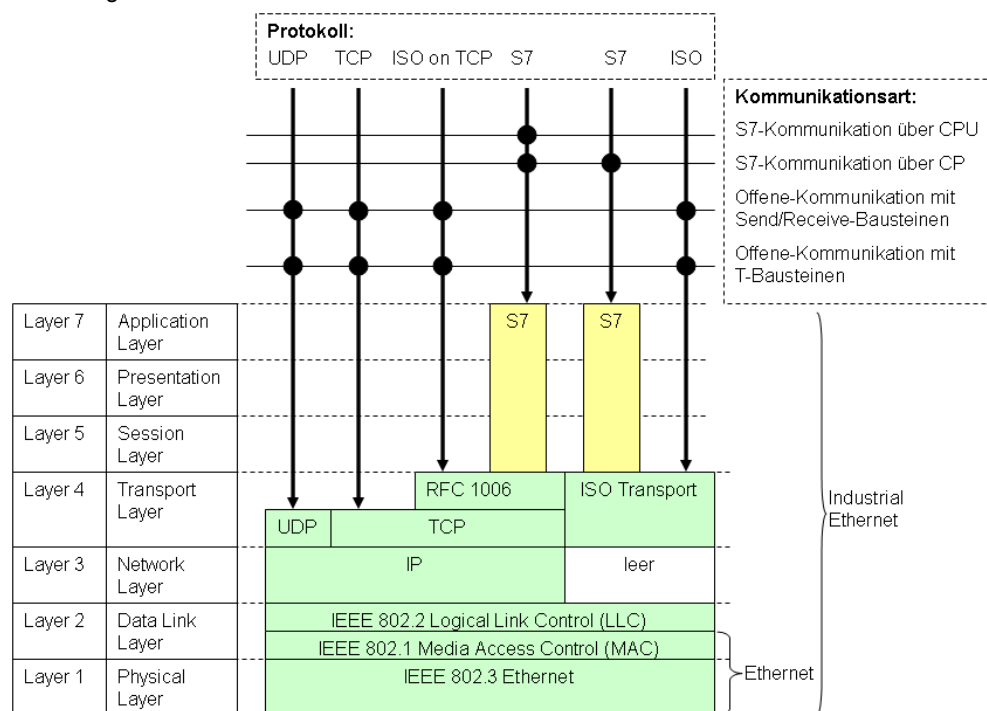
Neben den Merkmalen für Ethernet (Kapitel 10.2), gibt es für Industrial Ethernet folgende zusätzliche Merkmale:

- 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 10-3



10.4 PROFINET (PN)

Einführung

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

Ausprägungen

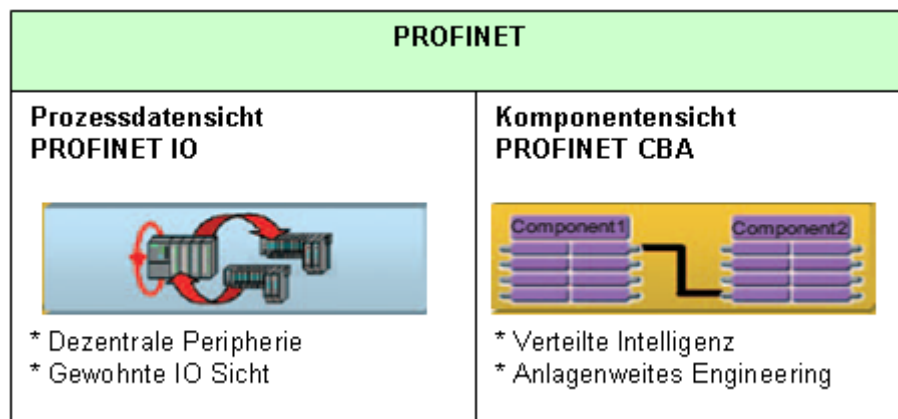
Im Rahmen von PROFINET gibt es zwei Ausprägungen:

- **PROFINET IO:**
Automatisierungskonzept für die Realisierung modularer Applikationen durch die Einbindung von dezentraler Peripherie mit Echtzeitkommunikation.
- **PROFINET CBA:**
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 10-4



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

PROFINET CBA gliedert eine komplette Automatisierungsanlage in verschiedene Funktionen auf. Diese Funktionen werden projiziert 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 projektiert (mit Tool iMap), nicht programmiert

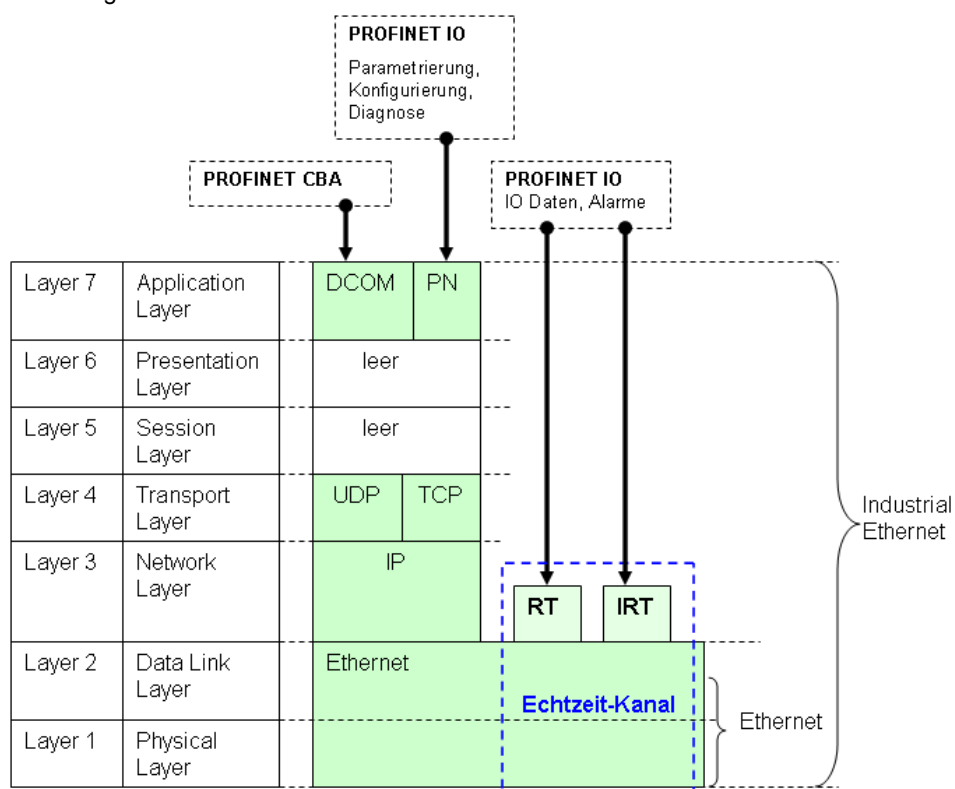
ISO/OSI-Referenzmodell

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

Tabelle 10-1

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

Abbildung 10-5



11 PROFIBUS (PB)

Einführung

PROFIBUS ist ein international standardisiertes, elektrisches Feldbussystem.

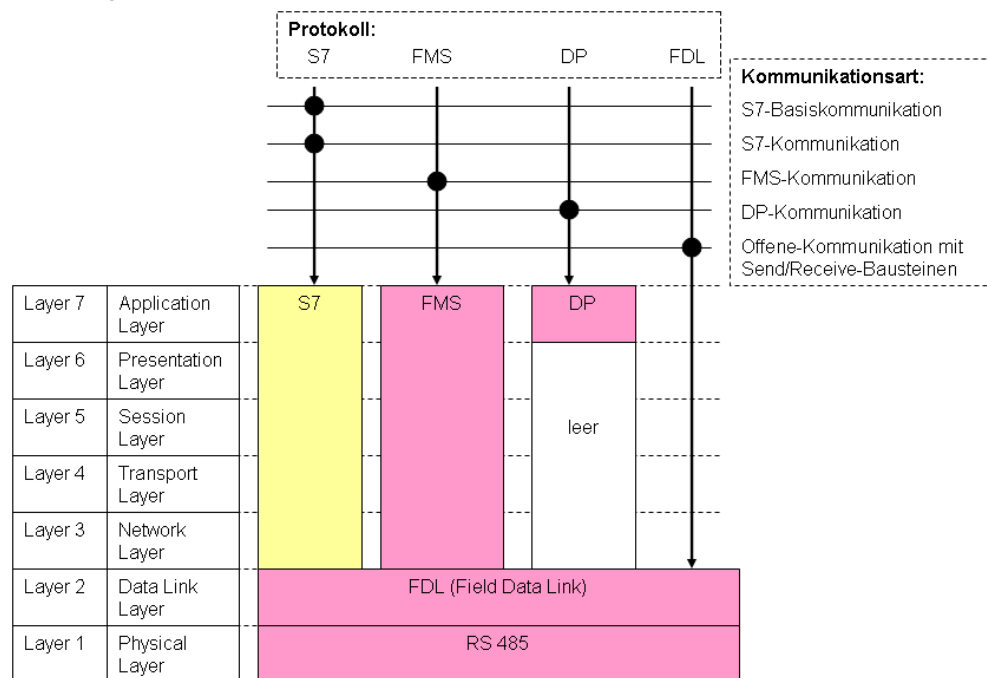
Merkmale

- 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 11-1



12 MPI

Einführung

MPI ist das Netz der SIMATIC für die Kommunikation mit PG/OP und die CPU-CPU Kommunikation. Eine MPI Schnittstelle ist auf CPUs der modularen SIMATIC Controller integriert.

Folgende Controller besitzen keine MPI Schnittstelle: S7-1200 und S7-1500.

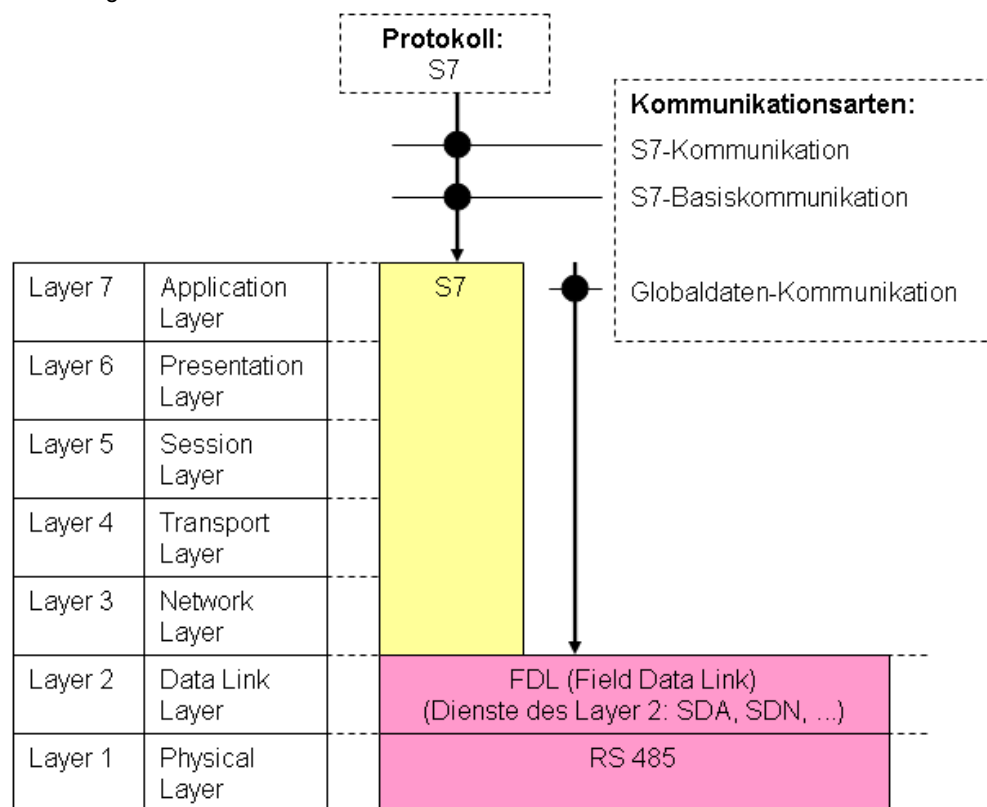
Merkmale

- Herstellerspezifische Schnittstelle
- 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 12-1



13 SIMATIC Rückwandbus

Einführung

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

Hinweis

- Nur möglich bei SIMATIC S7-400 (Multicomputing, Kapitel 5.5)
- Wird von STEP 7 (TIA) nicht unterstützt.

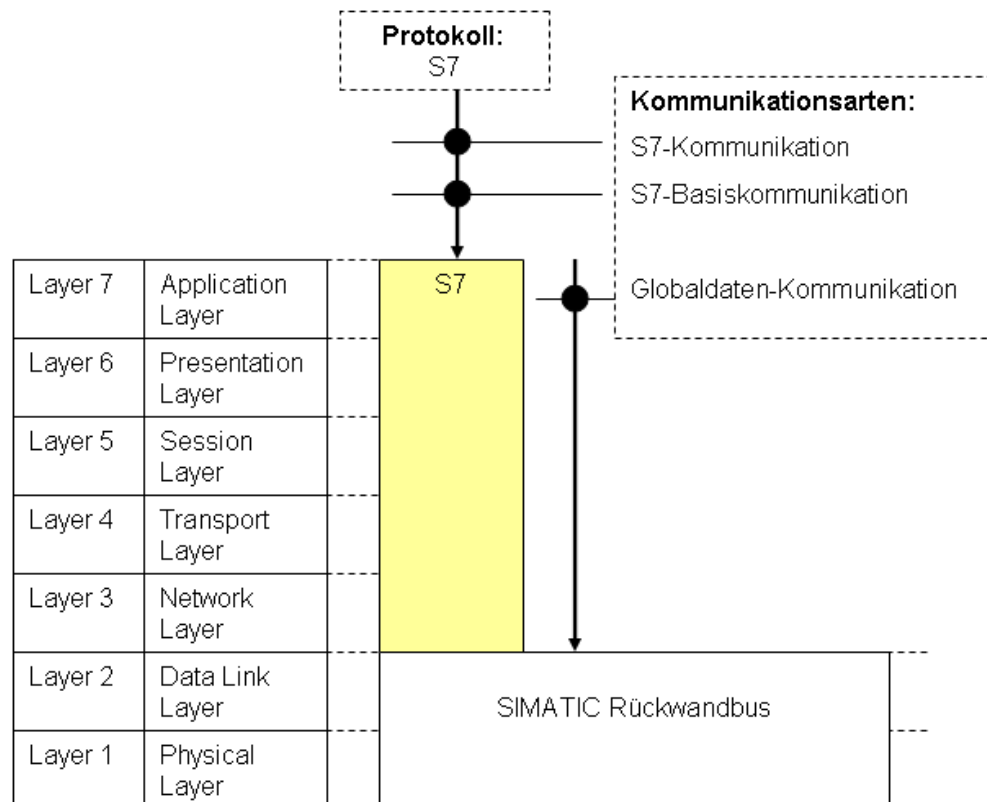
Merkmale

- Herstellerspezifische Schnittstelle
- 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 13-1



14 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 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).

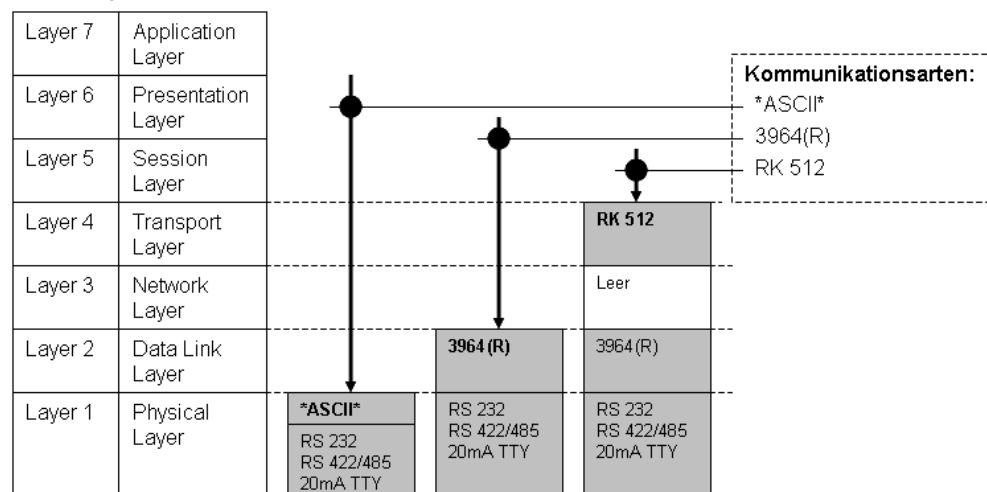
Merkmale

- 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 14-1



15 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 15-1

Medium		Übertragungsrate (von ...bis)	maximale Anzahl Partner	maximale Abstand zweier Partner	maximale Anzahl Daten pro Auftrag	Redundanter Aufbau mög- lich?
Netz	PROFINET/ Industrial Ethernet	PN: 10/100 MBit/s IE: 1/10 GBit/s	über 1000	elektrisch: 100 m optisch: 5 km (Multimode) bzw. 120 km (Singlemode)	64 KByte	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	ja
	MPI	187,5 KBit/s bis 12 MBit/s	126	elektrisch: 50 m	64 KByte	nein
SIMATIC Rückwandbus		10 MBit/s	4 CPUs in der SIMATIC Station	---	64 KByte	---
Serielle Schnittstelle		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	---

16 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 16-1

SIMATIC Controller	SIMATIC Familie	MPI	PROFIBUS			PN/IE	PN/IE			Serielle Schnitt- stelle	
				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	
	S7-400	X	X	X	X	X	X	X	X	X	
	S7-1200	---	X	X	X	X	X	---	---	X	
	S7-1500	---	X	X	X	X	X	X	---	X	
PC-based Controller	WinAC RTX	---	X	X	---	X	X	---	X	X	
	S7-mEC (mit WinAC RTX)										
	Box PC (mit WinAC RTX)										
	Panel PC (mit WinAC RTX)										

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.

17 Informationen zu Teil 1

In der Tabelle sind Verweise auf Informationen zu Themen aus Teil 1 (Einführung) zu finden. Alle Verweise /x/ sind zentral im Kapitel 61 hinterlegt. Dort sind auch die entsprechenden Links ins Internet zu finden.

Tabelle 17-1

/x/	Titel	Informationen zu
/0/	Siemens Industry Online Support: FAQs, Handbücher	Verbindungen Verbindungsressourcen Kommunikationsarten Datenkonsistenz
/6/	SIMATIC System- und Standardfunktionen für S7-300/400, Referenzhandbuch	
/15/	SIMATIC / Hardware konfigurieren und Verbindungen projektieren mit STEP 7 V5.5, Handbuch	
/33/	SIMATIC STEP 7 V5.5, Handbuch	
/32/	SIMATIC STEP 7 Professional V12.0, Systemhandbuch	SIMATIC Controller
/1/	SIMATIC Controller / Die innovative Lösung für alle Automatisierungsaufgaben, Broschüre	
/4/	Katalog ST 70, Produkte für Totally Integrated Automation und Micro Automation	
/2/	SIMATIC NET, Industrielle Kommunikation, Broschüre	Medien
/3/	SIMATIC, Kommunikation mit SIMATIC, Systemhandbuch	
/5/	Katalog IK PI, Industrielle Kommunikation	

18 ***** TEIL 2: Auswahlhilfe *****

18.1 Gliederung und Inhalt

Tabelle 18-1

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

Erläuterungen von Begriffen in obiger Tabelle

Medium:

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

SIMATIC Familien:

- Modulare Controller:
ET 200 CPU, S7-300, S7-400, S7-1200, S7-1500
- SIMATIC PC-based Controller:
WinAC RTX, S7-mEC, Box PC, Panel PC

Kommunikationsarten:

- SIMATIC spezifisch
- Offener Standard

19 Vorbemerkungen

Im Folgenden wird erläutert:

- Prinzip der Darstellung aller Kombinationen von SIMATIC Familien
- Erläuterung der im Teil 2 verwendeten Tabellen:
 - Tabelle Schnittstellen
 - Tabelle Kombinationen
 - Tabelle Kommunikationsarten-Kompakt

19.1 Prinzip der Darstellung aller Kombinationen

Im Dokument werden die Möglichkeiten der Kommunikation zwischen zwei SIMATIC Familien (x, y) dargestellt.

Die beiden SIMATIC Familien (x, y) werden als Paar bezeichnet.

Bei einem Paar ergeben sich zwei Kombinationen:

- x/y (x kommuniziert mit y)
- y/x (y kommuniziert mit x)

Für jedes Paar (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 einzigen Kapitel (Kapitel x/y). In dem anderen Kapitel (Kapitel y/x) steht lediglich ein Verweis auf dieses Kapitel (Kapitel x/y).

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

Beispiel

Paar:

- ET 200 CPU, S7-300

Kombinationen:

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

19.1.1 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. Zur Verdeutlichung wird folgendes Beispiel betrachtet:

- Medium PN/IE.
- Paar: ET 200 CPU, S7-300

Die folgende Abbildung zeigt einen Auszug aus dem Inhaltsverzeichnis.

Abbildung 19-1

21.3	PN/IE: Controller 1 = ET 200 CPU
21.3.1	ET 200 CPU / ET 200 CPU
21.3.2	ET 200 CPU / S7-300
21.3.3	ET 200 CPU / S7-400
21.3.4	ET 200 CPU / S7-1200
21.3.5	ET 200 CPU / S7-1500
21.3.6	ET 200 CPU / WinAC RTX
21.4	PN/IE: Controller 1 = S7-300
21.4.1	(S7-300 / ET 200 CPU)
21.4.2	S7-300 / S7-300
21.4.3	S7-300 / S7-400
21.4.4	S7-300 / S7-1200
21.4.5	S7-300 / S7-1500
21.4.6	S7-300 / WinAC RTX

Das Paar wird hier beschrieben.

Im Kapitel 21.4.1 steht ein **Verweis** auf das Kapitel 21.3.2

19.1.2 Kombinationen (Sprungverteiler)

Im Dokument werden die Kombinationen in Tabellen (Sprungverteiler) dargestellt. Für jede Kombination wird das Kapitel eingetragen, in dem diese Kombination beschrieben wird. Zur Verdeutlichung wird folgendes Beispiel betrachtet:

- Medium PN/IE
- Paar: ET 200 CPU, S7-300

Die folgende Abbildung zeigt den Sprungverteiler für PN/IE.

Abbildung 19-2

Controller 2	Controller 1					
	ET 200 CPU	S7-300	S7-400	S7-1200	S7-1500	WinAC RTX
ET 200 CPU	21.3.1	21.3.2	21.3.3	21.3.4	21.3.5	21.3.6
S7-300	21.3.2	21.4.2	21.4.3	21.4.4	21.4.5	21.4.6
S7-400	21.3.3	21.4.3	21.5.3	21.5.4	21.5.5	21.5.6
S7-1200	21.3.4	21.4.4	21.5.4	21.6.4	21.6.5	21.6.6
S7-1500	21.3.5	21.4.5	21.5.5	21.6.5	21.7.5	21.7.6
WinAC RTX	21.3.6	21.4.6	21.5.6	21.6.6	21.7.6	21.8.6

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

- Controller 1: ET 200 CPU / Controller 2: S7-300 (grün umrandet)
- Controller 1: S7-300 / Controller 2: ET 200 CPU (rot umrandet)

Beide Einträge enthalten die identische Kapitelnummer.

19.2 Tabelle Schnittstellen

19.2.1 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 19.3).

Die Tabelle beantwortet folgende Fragen:

- Welche Schnittstellen (CPU, CP, CM) stehen zur Verfügung?
- Welche Kommunikationsarten sind möglich?

19.2.2 Aufbau der Tabelle

Im Folgenden wird der Aufbau der Tabellen für die Medien PN/IE und PB 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 21-2).

Abbildung 19-3

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

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

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

(3) TSEND/TRCV, TUSEND/TURCV

(8) AG_SEND/AG_RECV

(13) Server für Fetch, Write (Im Server erforderliche Kommunikationsbausteine: FW_TCP bzw. FW_IOT)

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DA1

(6) PNIO_SEND, PNIO_RECV

3

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

Bereich 1: Schnittstellen Controller

Hier werden die Eigenschaften der SIMATIC Familie beschrieben:

Tabelle 19-1

	Bedeutung	Beispiel
1a	Bezeichnung des Mediums und der SIMATIC Familie	S7-300 an PN/IE
1b	Art der Schnittstelle: Integrierte Schnittstelle einer CPU, oder externe Schnittstelle eines CP bzw. CM	CPU, CP
1c	Bezeichnung der Baugruppen	CPU mit PN Schnittstelle CP 343-1 Lean CP 343-1 Advanced CP 343-1 ERPC

Bereich 2: Kommunikationsarten

Hier werden die Kommunikationsmöglichkeiten beschrieben:

Tabelle 19-2

	Bedeutung	Beispiel
2a	Bezeichnung der Kommunikationsarten, unterteilt in: <ul style="list-style-type: none"> SIMATIC spezifisch 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 dabei möglichen Kommunikationsbausteine werden entweder direkt eingetragen, oder über einen Index (x) repräsentiert. Der Index (x) wird im Bereich 3 erläutert.	direkter Eintrag (*1): "PUT, GET" Server (IoT) Index: (1), (2), (3), (6), (8), (13)
	Die möglichen Protokolle werden in Klammern angegeben.	IoT (*2), ISO, TCP, UDP
	Einträge in der Tabelle: --- Kommunikation nicht möglich x Kommunikation möglich	

Zu (*1):

Wenn nur die Funktionalität Server möglich ist, dann wird dies mit einem entsprechenden Zusatz beschrieben.

Beispiel: "PUT, GET", Server

Wenn sowohl Client als auch Server möglich sind, erfolgt kein Zusatz.

Beispiel: "PUT, GET"

Zu (*2):

Mit IoT ist ISO-on-TCP gemeint. Die Bezeichnung IoT wurde im Dokument aus Platzgründen gewählt.

Bereich 3: Kommunikationsbausteine zum Index (x)

Hier werden die zu einem Index möglichen Kommunikationsbausteine 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 22-2).

Abbildung 19-4

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

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

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

(2) Lade/Transfer Befehle, DPRD_DAT, DPWF

(7) DP_SEND, DP_RECV

(8) AG_SEND/AG_RECV

(10) READ, WRITE, REPORT

Der Aufbau der Tabelle ist genau so wie beim Medium PN/IE (siehe vorher), mit Ausnahme von Bereich 2b.

Bereich 2b: Funktionalität der Schnittstelle

Hier werden die möglichen Funktionalitäten der Schnittstelle angegeben:

- DP-Master oder DP-Slave
- DP-Slave: aktiv oder passiv

Je nach Funktionalität ergeben sich unterschiedliche Kommunikationsarten.

19.2.3 Abkürzungen und Indizes

Eine Übersicht aller Abkürzungen ist im Kapitel 63 zu finden. Eine Übersicht aller Indizes (x) der Tabelle Schnittstellen ist im Kapitel 19.5 zu finden.

19.3 Tabelle Kombinationen

19.3.1 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, CM) stehen zur Verfügung?
- Welche Kommunikationsarten sind möglich?

19.3.2 Aufbau der Tabelle

Der Aufbau der Tabelle wird an einem vereinfachten (nicht realen) Beispiel beschrieben. Für das Beispiel gelten die Randbedingungen:

- Controller 1 ist eine SIMATIC Familie mit einer CPU Schnittstelle (CPU301) und zwei CP Schnittstellen (CP302, CP303)
- Controller 2 ist eine SIMATIC Familie mit einer CPU Schnittstelle (CPU401) und zwei CP Schnittstellen (CP402, CP403)
- Es wird angenommen, dass zwei Kommunikationsarten möglich sind: S7-Kommunikation (S7) und Offene-Kommunikation (OC)

Unter obigen Randbedingungen sieht die Tabelle Kombinationen so aus:

Abbildung 19-5

Controller 2: S7-400				Controller 1: S7-300 an PN/IE			
CPU		CP		S7		OC	
CP	CP401	IOC	(1)	(3)	(21)	(8)/(3)	(1)
CP	CP402	IOD	(1)	(3)/(8)	(21)	(8)	(1)
	CP403	IOD	(1)	(3)/(8)	(21)	(8)	(1)

Kommunikationsbausteine

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

Die Tabelle ist als Matrix aufgebaut. Sie zeigt alle Kombinationen für alle Schnittstellen der beiden Controller. Es folgt eine Beschreibung der einzelnen Bereiche.

Bereich 1

Der Bereich zeigt alle kommunikationsfähigen Baugruppen von Controller 1.

Eine Spalte repräsentiert eine Schnittstelle:

- Zeile 1: CPU, CP oder CM
- Zeile 2: Bezeichnung der Baugruppe
- Zeile 3: Funktionalität der Schnittstelle auf der Baugruppe

Beispiel (roter Rahmen)

Der CP302 hat eine Schnittstelle mit der Funktionalität PN IO-Device (IOD)

Bereich 2

Der Bereich zeigt alle kommunikationsfähigen Baugruppen von Controller 2.

Eine Zeile repräsentiert eine Schnittstelle:

- Spalte 1: CPU, CP oder CM
- Spalte 2: Bezeichnung der Baugruppe
- Spalte 3: Funktionalität der Schnittstelle auf der Baugruppe

Beispiel (roter Rahmen)

Die CPU401 hat eine Schnittstelle mit der Funktionalität PN IO-Controller (IOC)

Bereich 3

Im Bereich 3 werden die Kommunikationsarten aufgeführt, die für beide Controller möglich sind. Sie werden für jede Schnittstelle (Spalte) von Controller 1 (Bereich 1) wiederholt.

Im Beispiel sind dies:

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

Somit wird in der Tabelle jede gemeinsam mögliche Kommunikationsart zwischen Controller 1 und Controller 2 berücksichtigt (siehe Erläuterungen zu Bereich 4).

Bereich 4

Jede Zelle im Bereich repräsentiert eine Kombination von zwei Schnittstellen.

Der Bereich liefert pro Kombination folgende Informationen:

- Gemeinsam mögliche Kommunikationsarten (abzulesen in Bereich 3)
- Zugehörige Kommunikationsbausteine (repräsentiert über einen Index (x))

Die Schreibweise der Indizes (x) in den Zellen liefert folgende Informationen:

Tabelle 19-3

Schreibweise in Zelle	Ist die Kommunikationsart entsprechend Bereich 3 möglich?	Welche Kommunikationsbausteine können eingesetzt werden?
(1)	ja	Siehe Bereich 5: Index (1)
(1)+(2)	ja	Siehe Bereich 5: (1) oder (2)
(1) / (2)	ja	Controller 1 mit (1) / Controller 2 mit (2)
(1) / (2)+(3)	ja	Hier sind zwei Fälle 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	Es sind <u>keine</u> Kommunikationsbausteine erforderlich
---	nein	Entfällt, da die Kommunikationsart nicht möglich ist.

Bereich 5

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 im Bereich 4 der Abbildung 19-5 bedeutet:

Kombination:

- CPU301 (Funktionalität der Schnittstelle: IOC, IOD)
- CP402 (Funktionalität der Schnittstelle: IOD)

Kommunikationsart:

- Offene-Kommunikation (OC)

Kommunikationsbausteine:

- Für Controller 1: TSEND/TRCV, TUSEND/TURCV
- Für Controller 2: AG_SEND/AG_RECV

Weitere Beispiele zum Bereich 4:

(1) bedeutet:

Kommunikationsbausteine:

- Beide Controller: "USEND/URCV", BSEND/BRCV, "PUT, GET"
- Beide Controller: Client oder Server für "PUT, GET"

(21) bedeutet:

Kommunikationsbausteine:

- Für Controller 1: Kann nur Server für "PUT, GET" sein

19.3.3 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 folgende Bild zeigt die zugehörige Tabelle Kombinationen:

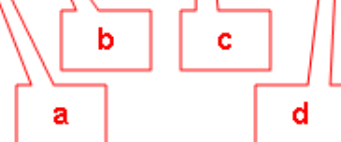
Abbildung 19-6

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	
					PNIO	CBA
CPU	Schnittstelle: PN	IOC, IOD	(1)	(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
 (22) Controller 2 ist Server (für "PUT, GET")



In der folgenden Tabelle werden Beispiele zum obigen Bild beschrieben. Erläuterungen zur Syntax sind im Kapitel 19.3.2 (Absatz "Bereich 4") zu finden.

Tabelle 19-4

im Bild	Controller 1		Controller 2		Kommunikationsart	Kommunikationsbausteine
	Schnittstelle	Funktionalität	Schnittstelle	Funktionalität		
a	IM151-8, IM154-8	IOC, IOD	343- 1	IOC, IOD	S7	Beide Controller: (1)
b	IM151-8, IM154-8	IOC, IOD	343-1 ERPC	---	OC	Controller 1: (3) Controller 2: (8)
c	IM151-8, IM154-8	IOC, IOD	343-1 Lean	IOD	PNIO	Controller 1: (2) Controller 2: (6)
d	IM151-8, IM154-8	IOC, IOD	S7-300 CPU PN	IOC, IOD	CBA	keine

19.3.4 Abkürzungen und Indizes

Eine Übersicht aller Abkürzungen ist im Kapitel 63 zu finden. Eine Übersicht aller Indizes (x) der Tabelle Kombinationen ist im Kapitel 19.5 zu finden.

19.4 Tabelle Kommunikationsarten-Kompakt

19.4.1 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?

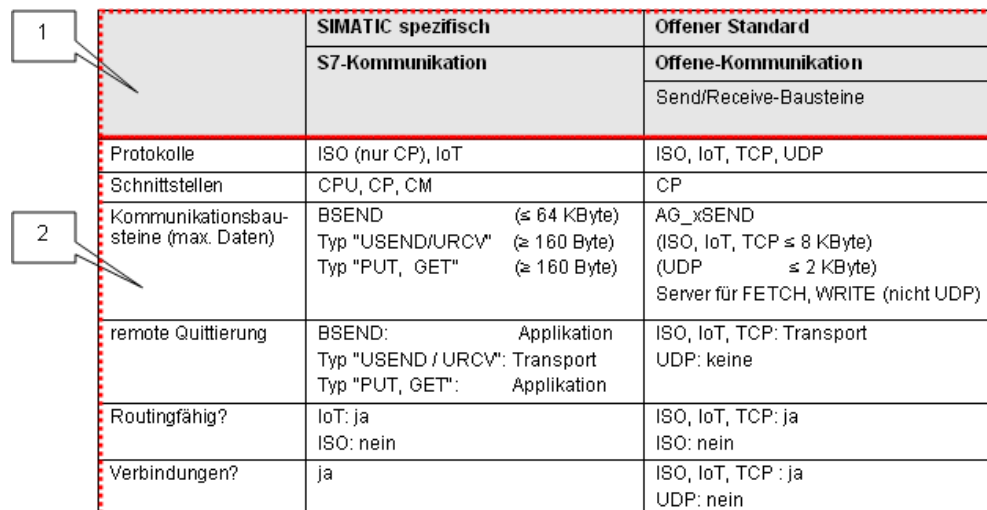
Die Tabelle Kommunikationsarten-Kompakt ist eine Zusammenfassung (Verdichtung) der Tabellen Kommunikationsarten-Detail.

Nicht jede SIMATIC Familie bzw. Baugruppe erfüllt alle in der Tabelle Kommunikationsarten-Kompakt angegebenen Werte. Die Werte sind als Obergrenzen zu betrachten.

19.4.2 Aufbau der Tabelle

Beispiel: Ausschnitt aus der Tabelle für PN/IE (Tabelle 21-34).

Abbildung 19-7



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

Die Tabelle besteht aus zwei Bereichen.

Bereich 1

In diesem Bereich stehen die Bezeichnungen für die Kommunikationsarten, unterteilt in "SIMATIC spezifisch" und "Offener Standard".

Bereich 2

Hier sind die wichtigsten Eigenschaften der Kommunikationsarten eingetragen.

Die Kriterien (Protokolle, Schnittstellen, ...) werden im Kapitel 28.2.3 beschrieben.

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

19.5 Übersicht der Indizes in den Tabellen

Tabelle Schnittstellen und Tabelle Kombinationen

In den Tabellen Schnittstellen und den Tabellen Kombinationen werden folgende Indizes verwendet:

Tabelle 19-5

(x)	Anwenderschnittstellen (Kommunikationsbausteine)
(1)	"USEND/URCV", BSEND/BRCV, "PUT, GET" (*1)
(2)	Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT
(3)	TSEND/TRCV, TUSEND/TURCV (*2)
(4)	TSEND/TRCV (*2)
(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

Tabelle Kombinationen

Folgende Indizes werden nur in den Tabellen Kombinationen verwendet:

Tabelle 19-6

(x)	Anwenderschnittstellen (Kommunikationsbausteine)
Kommunikationsart: S7-Kommunikation	
(21)	Controller 1 ist Server (für "PUT, GET") (*1)
(22)	Controller 2 ist Server (für "PUT, GET") (*1)
(40)	"PUT, GET" (beide Controller können Client oder Server sein) (*1)
Kommunikationsart: S7-Basiskommunikation	
(61)	Controller 1 ist Server (für I_PUT, I_GET)
(62)	Controller 2 ist Server (für I_PUT, I_GET)

Erläuterungen

Zu (*1):

Die Bezeichnung "USEND/URCV" steht für alle Varianten der Kommunikationsbausteine (USEND, USEND_E, USEND_S, ...). Siehe Kapitel 32.4.

Die Bezeichnung "PUT, GET" steht für alle Varianten der Kommunikationsbausteine (PUT, PUT_E, PUT_S, ...). Siehe Kapitel 32.6.

Ein Überblick zu allen Varianten ist in Kapitel 32.3 zu finden.

Zu (*2):

Bei (3) sind TSEND/TRCV und TUSEND/TURCV möglich.

Bei (4) ist TUSEND/TURCV nicht möglich.

20 >>> SPRUNGVERTEILER >>>

Das Kapitel enthält Querverweise zu allen wichtigen Informationen im Dokument.

Hinweise zum Umgang mit dem Sprungverteiler: Siehe Kapitel 3.

20.1 Sprungverteiler: Netz PN/IE

20.1.1 Übersicht Schnittstellen und Kommunikationsarten

Tabelle 20-1 Links zu den Tabellen Schnittstellen

Familie	Kapitel
ET 200 CPU	21.2.1
S7-300	21.2.2
S7-400	21.2.3
S7-1200	21.2.4
S7-1500	21.2.5
S7-mEC	21.2.6
Box PC	21.2.7
Panel PC	21.2.8
WinAC RTX	21.2.9

20.1.2 Kombinationen Controller 1 / Controller 2

Tabelle 20-2 Links zu den Tabellen Kombinationen

Controller 2	Controller 1					
	ET 200 CPU	S7-300	S7-400	S7-1200	S7-1500	WinAC RTX
ET 200 CPU	21.3.1	21.3.2	21.3.3	21.3.4	21.3.5	21.3.6
S7-300	21.3.2	21.4.2	21.4.3	21.4.4	21.4.5	21.4.6
S7-400	21.3.3	21.4.3	21.5.3	21.5.4	21.5.5	21.5.6
S7-1200	21.3.4	21.4.4	21.5.4	21.6.4	21.6.5	21.6.6
S7-1500	21.3.5	21.4.5	21.5.5	21.6.5	21.7.5	21.7.6
WinAC RTX	21.3.6	21.4.6	21.5.6	21.6.6	21.7.6	21.8.6

Hinweis: Die SIMATIC Controller S7-mEC, Box PC und Panel PC verhalten sich bezüglich der CPU-CPU Kommunikation wie WinAC RTX.

20.1.3 Kommunikationsarten

Tabelle 20-3 Links zu den Tabellen Kommunikationsarten

Kommunikationsart		Kapitel
Alle Kommunikationsarten (Tabelle Kompakt)		21.9
Tabelle mit Details	S7-Kommunikation	32.2
	Offene-Kommunikation mit Send/Receive-Bausteinen	34.2
	Offene-Kommunikation mit T-Bausteinen	35.2
	PNIO	37.2
Modbus/TCP (SIMATIC / fremder Controller)		57

20.2 Sprungverteiler: Netz PB

20.2.1 Übersicht Schnittstellen und Kommunikationsarten

Tabelle 20-4 Links zu den Tabellen Schnittstellen

Familie	Kapitel
ET 200 CPU	22.2.1
S7-300	22.2.2
S7-400	22.2.3
S7-1200	22.2.4
S7-1500	22.2.5
S7-mEC	22.2.6
Box PC	22.2.7
Panel PC	22.2.8
WinAC RTX	22.2.9

20.2.2 Kombinationen Controller 1 / Controller 2

Tabelle 20-5 Links zu den Tabellen Kombinationen

Controller 2	Controller 1					
	ET 200 CPU	S7-300	S7-400	S7-1200	S7-1500	WinAC RTX
ET 200 CPU	22.3.1	22.3.2	22.3.3	22.3.4	22.3.5	22.3.6
S7-300	22.3.2	22.4.2	22.4.3	22.4.4	22.4.5	22.4.6
S7-400	22.3.3	22.4.3	22.5.3	22.5.4	22.5.5	22.5.6
S7-1200	22.3.4	22.4.4	22.5.4	22.6.4	22.6.5	22.6.6
S7-1500	22.3.5	22.4.5	22.5.5	22.6.5	22.7.5	22.7.6
WinAC RTX	22.3.6	22.4.6	22.5.6	22.6.6	22.7.6	22.8.6

Hinweis: Die SIMATIC Controller S7-mEC, Box PC und Panel PC verhalten sich bezüglich der CPU-CPU Kommunikation wie WinAC RTX.

20.2.3 Kommunikationsarten

Tabelle 20-6 Links zu den Tabellen Kommunikationsarten

Kommunikationsart		Kapitel
Alle Kommunikationsarten (Tabelle Kompakt)		22.9
Tabelle mit Details	S7-Basiskommunikation	31.2
	S7-Kommunikation	32.2
	Offene-Kommunikation mit Send/Receive-Bausteinen	39.2
	FMS-Kommunikation	40.2
	DP-Kommunikation	41.2

20.3 Sprungverteiler: Netz MPI

20.3.1 Übersicht Schnittstellen und Kommunikationsarten

Tabelle 20-7 Links zu den Tabellen Schnittstellen

Familie	Kapitel
ET 200 CPU	23.2.1
S7-300	23.2.2
S7-400	23.2.3

20.3.2 Kombinationen Controller 1 / Controller 2

Tabelle 20-8 Links zu den Tabellen Kombinationen

Controller 2	Controller 1		
	ET 200 CPU	S7-300	S7-400
ET 200 CPU	23.3.1	23.3.2	23.3.3
S7-300	23.3.2	23.4.2	23.4.3
S7-400	23.3.3	23.4.3	23.5.3

20.3.3 Kommunikationsarten

Tabelle 20-9 Links zu den Tabellen Kommunikationsarten

Kommunikationsart		Kapitel
Alle Kommunikationsarten (Tabelle Kompakt)		23.6
Tabelle mit Details	Globaldaten	30.2
	S7-Basiskommunikation	31.2
	S7-Kommunikation	32.2

20.4 Sprungverteiler: SIMATIC Rückwandbus

20.4.1 Übersicht Schnittstellen und Kommunikationsarten

Tabelle 20-10 Links zu den Tabellen Schnittstellen

Familie	Kapitel
S7-400	24.2

20.4.2 Kombinationen Controller 1 / Controller 2

Tabelle 20-11 Links zu den Tabellen Kombinationen

Controller 2	Controller 1
	S7-400
S7-400	24.3

20.4.3 Kommunikationsarten

Tabelle 20-12 Links zu den Tabellen Kommunikationsarten

Kommunikationsart		Kapitel
Alle Kommunikationsarten (Tabelle Kompakt)		24.4
Tabelle mit Details	Globaldaten	30.2
	S7-Basiskommunikation	31.2
	S7-Kommunikation	32.2

20.5 Sprungverteiler: Serielle Schnittstellen

20.5.1 Übersicht Schnittstellen und Kommunikationsarten

Tabelle 20-13 Links zu den Tabellen Schnittstellen

Familie	Kapitel
ET 200 CPU	25.2
S7-300	25.3
S7-400	25.4
S7-1200	25.5
S7-1500	25.6
S7-mEC	25.7
Box PC	25.8
Panel PC	25.9
WinAC RTX	25.10
Dezentrale Station ET 200	25.11

20.5.2 Kommunikationsarten

Tabelle 20-14 Links zu den Tabellen Kommunikationsarten

Kommunikationsart	Kapitel
ASCII, 3964(R), RK 512	42.2
Überblick Anwenderschnittstellen	43
Modbus Seriell (RTU Format)	58

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

21.1 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)

21.2 PN/IE: Schnittstellen und Kommunikationsarten

21.2.1 ET 200 CPU an PN/IE

Tabelle 21-1

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)	(3) (IoT, TCP, UDP)	(2)	(2)	x
ET 200Pro	CPU	IM 154-8(F) PN/DP CPU	(1)	(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

21.2.2 S7-300 an PN/IE

Tabelle 21-2

Controller an PN/IE: S7-300			Kommunikationsart					
			SIMATIC spezifisch	Offener Standard				
				S7-Kommunikation	Offene-Kommunikation	PN-Kommunikation		
						IOC	IOD	CBA
CPU	Schnittstelle: PN (2 Ports)		(1)	(3) (IoT, TCP, UDP) (13) (IoT, TCP) (*1)	(2)	(2)	x	
CP	343-1 Lean	1 x PN (2 Ports)	"PUT, GET" Server	(8) (IoT, TCP, UDP) (13) (IoT, TCP)	---	(6)	---	
	343-1	1 x PN (2 Ports)	(1)	(8) (ISO, IoT, TCP, UDP) (13) (ISO, IoT, TCP)	(6)	(6)	---	
	343-1 Advanced	1 x PN (2 Ports)	(1)	(8) (ISO, IoT, TCP, UDP) (13) (IoT, TCP, ISO)	(6)	(6)	x	
		1 x IE (1 Port)	(1)	(8) (ISO, IoT, TCP, UDP) (13) (IoT, TCP, ISO)	---	---	---	
	343-1 ERPC	1 x IE (1 Port)	(1)	(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 (Im Server erforderliche Kommunikationsbausteine: FW_TCP bzw. FW_IOT)

21.2.3 S7-400 an PN/IE

Tabelle 21-3

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

(4) TSEND/TRCV

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

(13) Server für Fetch, Write (Im Server erforderliche Kommunikationsbausteine: FW_TCP bzw. FW_IOT)

21.2.4 S7-1200 an PN/IE

Tabelle 21-4

Controller an PN/IE: S7-1200		Kommunikationsart				
		SIMATIC spezifisch	Offener Standard			
			Offene-Kommunikation	PN-Kommunikation		
		S7-Kommunikation		IOC	IOD	CBA
CPU	Schnittstelle: PN	"PUT, GET"	(3) (IoT, TCP, UDP) (5) (IoT, TCP)	(2)	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(3) TSEND/TRCV, TUSEND/TURCV

(5) TSEND_C/TRCV_C

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

21.2.5 S7-1500 an PN/IE

Tabelle 21-5

Controller an PN/IE: S7-1500			Kommunikationsart				
			SIMATIC spezifisch		Offener Standard		
			S7-Kommunikation	Offene-Kommunikation	PN-Kommunikation		
					IOC	IOD	CBA
CPU	1511-1 PN	1 x PN (2 Ports)	(1)	(3) (IoT, TCP, UDP)	(2)	(2)	---
	1513-1 PN			(5) (IoT, TCP, UDP)			
	1516-3 PN/DP	1 x PN (2 Ports)	(1)	(3) (IoT, TCP, UDP)	(2)	(2)	---
		1 x IE (1 Port)	(1)	(5) (IoT, TCP, UDP)	---	---	---
CP	1543-1	1 x IE (1 Port)	(1)	(3) (IoT, TCP, UDP, ISO)	---	---	---
				(5) (IoT, TCP, UDP, ISO)			
				(13) (ISO, IoT, 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
- (5) TSEND_C/TRCV_C
- (13) Server für Fetch, Write (Im Server sind keine Kommunikationsbausteine erforderlich.)

21.2.6 S7-mEC an PN/IE

Tabelle 21-6

Controller an PN/IE: S7-mEC mit WinAC RTX (F) 2010			Kommunikationsart				
			SIMATIC spezifisch		Offener Standard		
			S7-Kommunikation	Offene-Kommunikation	PN-Kommunikation		
					IOC	IOD	CBA
CPU	EC31-RTX (F)	X1: unter Kontrolle von RTX (Submodule)	(1)	(3) (IoT, TCP, UDP)	(2)	---	x
		X2: unter Kontrolle von Windows	(1)	---	---	---	---
CP	EM PC	X1: unter Kontrolle von Windows	(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

21.2.7 Box PC an PN/IE

Tabelle 21-7

Controller an PN/IE: Box PC SIMATIC Embedded Bundles: IPC227D, IPC427C mit WinAC RTX (F) 2010		Kommunikationsart				
		SIMATIC spezifisch		Offener Standard		
		S7-Kommunikation	Offene-Kommunikation	PN-Kommunikation		
				IOC	IOD	CBA
CPU / CP (*1)	unter Kontrolle von RTX (Submodule): • „CP1616/CP1604“	(1)	(3) (IoT, TCP, UDP)	(2)	---	x
	unter Kontrolle von Windows: • „IE Allgemein“	(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

Erläuterungen zur Tabelle

Zu (*1): 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 21-8

Sammelbezeichnung	PC Karte (entspricht CP Schnittstelle)	Onboard Schnittstelle des PC (entspricht CPU Schnittstelle)
„CP1616/CP1604“	CP 1616, CP 1604	Onboard PN-Schnittstelle der SIMATIC IPC und S7-mEC: CP 1616 integriert
„IE Allgemein“	Standard Ethernet Karte (Intel Chipsatz, non shared IRQ)	Onboard Ethernet-Schnittstelle der SIMATIC IPC

21.2.8 Panel PC an PN/IE

Tabelle 21-9

Controller an PN/IE: Box PC SIMATIC Embedded Bundles: IPC277D, IPC477C mit WinAC RTX (F) 2010		Kommunikationsart				
		SIMATIC spezifisch		Offener Standard		
		S7-Kommunikation	Offene-Kommunikation	PN-Kommunikation		
				IOC	IOD	CBA
CPU / CP (*1)	unter Kontrolle von RTX (Submodule): • „CP1616/CP1604“	(1)	(3) (IoT, TCP, UDP)	(2)	---	x
	unter Kontrolle von Windows: • „IE Allgemein“	(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

Erläuterungen zur Tabelle

Zu (*1): 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 21-10

Sammelbezeichnung	PC Karte (entspricht CP Schnittstelle)	Onboard Schnittstelle des PC (entspricht CPU Schnittstelle)
„CP1616/CP1604“	CP 1616, CP 1604	Onboard PN-Schnittstelle der SIMATIC IPC und S7-mEC: CP 1616 integriert
„IE Allgemein“	Standard Ethernet Karte (Intel Chipsatz, non shared IRQ)	Onboard Ethernet-Schnittstelle der SIMATIC IPC

21.2.9 WinAC RTX an PN/IE

Tabelle 21-11

Controller an PN/IE: WinAC RTX WinAC RTX (F) 2010		Kommunikationsart				
		SIMATIC spezifisch	Offener Standard			
		S7-Kommunikation	Offene-Kommunikation	PN-Kommunikation		
				IOC	IOD	CBA
CPU / CP (*1)	unter Kontrolle von RTX (Submodule): • „CP1616/CP1604“ • „IE-Allgemein“	(1)	(3) (IoT, TCP, UDP)	(2)	---	x
	unter Kontrolle von Windows: • „CP1616/CP1604“ • „IE-Allgemein“	(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

Erläuterungen zur Tabelle

Zu (*1): 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 21-12

Sammelbezeichnung	PC Karte (entspricht CP Schnittstelle)	Onboard Schnittstelle des PC (entspricht CPU Schnittstelle)
„CP1616/CP1604“	CP 1616, CP 1604	Onboard PN-Schnittstelle der SIMATIC IPC und S7-mEC: CP 1616 integriert
„IE-Allgemein“	Standard Ethernet Karte (Intel Chipsatz, non shared IRQ)	Onboard Ethernet-Schnittstelle der SIMATIC IPC

21.3 PN/IE: Controller 1 = ET 200 CPU

21.3.1 ET 200 CPU / ET 200 CPU

Tabelle 21-13

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	
					PNIO	CBA
CPU	IM 151-8(F) PN/DP CPU IM 154-8(F) PN/DP CPU	IOC, IOD	(1)	(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

21.3.2 ET 200 CPU / S7-300

Tabelle 21-14

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	
					PNIO	CBA
CPU	Schnittstelle: PN	IOC, IOD	(1)	(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	Schnittstelle: PN	(1)	(3) / (8)	(2) / (6)	x
		Schnittstelle: IE	(1)	(3) / (8)	---	---
	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
- (22) Controller 2 ist Server (für "PUT, GET")

21.3.3 ET 200 CPU / S7-400

Tabelle 21-15

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	
PNIO	CBA						
CPU	Schnittstelle: PN		IOC, IOD	(1)	(3)	(2)	x
CP	443-1		IOC, IOD	(1)	(3) / (4)+(9)	(2)	---
	443-1 Advanced	Schnittstelle: PN	IOC, IOD	(1)	(3) / (4)+(9)	(2)	x
		Schnittstelle: IE	---	(1)	(3) / (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
- (9) AG_SEND/AG_RECV, AG_LSEND/AG_LRECV, AG_SSEND/AG_SRECV

21.3.4 ET 200 CPU / S7-1200

Tabelle 21-16

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	Schnittstelle: PN	IOC	(40)	(3) / (3)+(5)	(2)	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(3) TSEND/TRCV, TUSEND/TURCV

(5) TSEND_C/TRCV_C

(40) "PUT, GET" (beide Controller können Client oder Server sein)

21.3.5 ET 200 CPU / S7-1500

Tabelle 21-17

Controller 2: S7-1500			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	
PNIO	CBA					
CPU	Schnittstelle: PN	IOC, IOD	(1)	(3) / (3)+(5)	(2)	---
	Schnittstelle: IE	---	(1)	(3) / (3)+(5)	---	---
CP	CP 1543-1 (Schnittstelle IE)	---	(1)	(3) / (3)+(5)	---	---

[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
- (5) TSEND_C/TRCV_C

21.3.6 ET 200 CPU / WinAC RTX

Tabelle 21-18

Controller 2: WinAC RTX auf PC WinAC RTX (F) 2010			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/CP	Kontrolle von RTX (Submodule): • „CP1616/CP1604“ • „IE-Allgemein“	IOC	(1)	(3)	(2)	x
	Kontrolle von Windows: • „CP1616/CP1604“ • „IE-Allgemein“	---	(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

21.4 PN/IE: 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 21-19

Controller 2: S7-300			Controller 1: S7-300 an PN/IE																				
			CPU				CP																
			Schnittstelle: PN				343-1 Lean				343-1				343-1 Advanced						343-1 ERPC		
															Schnittstelle: PN				Schnittstelle: IE				
			IOC, IOD				IOD				IOC, IOD				IOC, IOD				---		---		
S7	OC	PN		S7	OC	PN		S7	OC	PN		S7	OC	PN		S7	OC	S7	OC				
		PNIO	CBA			PNIO	CBA			PNIO	CBA			PNIO	CBA								
CPU	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)	(1)	(8)/(3)
CP	343-1 Lean		IOD	(22)	(3)/(8)	(2)/(6)	---	---	(8)	---	---	(22)	(8)	(6)	---	(22)	(8)	(6)	---	(22)	(8)	(22)	(8)
	343-1		IOC, IOD	(1)	(3)/(8)	(2)/(6)	---	(21)	(8)	(6)	---	(1)	(8)	(6)	---	(1)	(8)	(6)	---	(1)	(8)	(1)	(8)
	343-1 Advanced	X: PN	IOC, IOD	(1)	(3)/(8)	(2)/(6)	x	(21)	(8)	(6)	---	(1)	(8)	(6)	---	(1)	(8)	(6)	x	(1)	(8)	(1)	(8)
		X: IE	---	(1)	(3)/(8)	---	---	(21)	(8)	---	---	(1)	(8)	---	---	(1)	(8)	---	---	(1)	(8)	(1)	(8)
343-1 ERPC			---	(1)	(3)/(8)	---	---	(21)	(8)	---	---	(1)	(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
- (21) Controller 1 ist Server (für "PUT, GET")
- (22) Controller 2 ist Server (für "PUT, GET")

21.4.3 S7-300 / S7-400

Tabelle 21-20

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

[Zurück zum Sprungverteiler PN/IE](#)Kommunikationsbausteine

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

21.4.4 S7-300 / S7-1200

Tabelle 21-21

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	
															Schnittstelle: PN				X: IE	
			IOC, IOD				IOD				IOC, IOD				IOC, IOD				---	
			S7	OC	PN		S7	OC	PN		S7	OC	PN		S7	OC	PN		S7	OC
					PNIO	CBA			PNIO	CBA			PNIO	CBA			PNIO	CBA		
CPU	Schnittstelle: PN	IOC	(40)	(3)/(3)+(5)	(2)	---	(21)	(8)/(3)+(5)	(2)	---	(40)	(8)/(3)+(5)	(2)	---	(40)	(8)/(3)+(5)	(2)	---	(40)	(8)/(3)+(5)

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(3) TSEND/TRCV, TUSEND/TURCV

(5) TSEND_C/TRCV_C

(8) AG_SEND/AG_RECV

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

(40) "PUT, GET" (beide Controller können Client oder Server sein)

21.4.5 S7-300 / S7-1500

Tabelle 21-22

Controller 2: S7-1500			Controller 1: S7-300 an PN/IE																			
			CPU				CP															
			Schnittstelle: PN				343-1 Lean				343-1				343-1 Advanced				343-1 ERPC			
															Schnittstelle: PN						X: IE	
			IOC, IOD				IOD				IOC, IOD				IOC, IOD				---		---	
			S7	OC		PN		S7	OC		PN		S7	OC		PN		S7	OC		S7	OC
			PNIO	CBA				PNIO	CBA				PNIO	CBA				PNIO	CBA			
CPU	Schnittstelle: PN	IOC, IOD	(1)	(3)/(3)+(5)	(2)	---	(21)	(8)/(3)+(5)	(6)/(2)	---	(1)	(8)/(3)+(5)	(6)/(2)	---	(1)	(8)/(3)+(5)	(6)/(2)	---	(1)	(8)/(3)+(5)	(1)	(8)/(3)+(5)
	Schnittstelle: IE	---	(1)	(3)/(3)+(5)	---	---	(21)	(8)/(3)+(5)	---	---	(1)	(8)/(3)+(5)	---	---	(1)	(8)/(3)+(5)	---	---	(1)	(8)/(3)+(5)	(1)	(8)/(3)+(5)
CP	CP 1543-1	---	(1)	(3)/(3)+(5)	---	---	(21)	(8)/(3)+(5)	---	---	(1)	(8)/(3)+(5)	---	---	(1)	(8)/(3)+(5)	---	---	(1)	(8)/(3)+(5)	(1)	(8)/(3)+(5)

[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
- (21) Controller 1 ist Server (für "PUT, GET")

21.4.6 S7-300 / WinAC RTX

Tabelle 21-23

Controller 2: WinAC RTX auf PC WinAC RTX (F) 2010			Controller 1: S7-300 an PN/IE																			
			CPU				CP															
			Schnittstelle: PN				343-1 Lean				343-1				343-1 Advanced				343-1 ERPC			
															Schnittstelle: PN						X: IE	
			IOC, IOD				IOD				IOC, IOD				IOC, IOD				---		---	
S7	OC	PN		S7	OC	PN		S7	OC	PN		S7	OC	PN		S7	OC	S7	OC	S7	OC	
		PNIO	CBA			PNIO	CBA			PNIO	CBA			PNIO	CBA							
CPU/CP	Kontrolle von RTX (Submodule): • „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)	(1)	(8)/(3)
	Kontrolle von Windows: • „CP1616-CP1604“ • „IE-Allgemein“	---	(1)	---	---	---	(21)	---	---	---	(1)	---	---	---	(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
- (6) PNIO_SEND, PNIO_RECV
- (8) AG_SEND/AG_RECV
- (21) Controller 1 ist Server (für "PUT, GET")

21.5 PN/IE: 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 21-24

Controller 2: S7-400				Controller 1: S7-400 an PN/IE											
				CPU				CP							
				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	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	Schnittstelle: PN	IOC, IOD	(1)	(3)/(4)+(9)	(2)	x	(1)	(4)+(9)	(2)	---	(1)	(4)+(9)	(2)	x
		Schnittstelle: IE	---	(1)	(3)/(4)+(9)	---	---	(1)	(4)+(9)	---	---	(1)	(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
- (9) AG_SEND/AG_RECV, AG_LSEND/AG_LRECV, AG_SSEND/AG_SRECV

Controller 2: S7-1200			Controller 1: S7-400 an PN/IE													
			CPU				CP									
			Schnittstelle: PN				443-1				443-1 Advanced					
											Schnittstelle: PN				X: IE	
			IOC, IOD				IOC, IOD				IOC, IOD				---	
			S7	OC	PN		S7	OC	PN		S7	OC	PN		S7	OC
PNIO	CBA	PNIO			CBA	PNIO			CBA							
CPU	Schnittstelle: PN	IOC	(40)	(3)/(3)+(5)	(2)	---	(40)	(4)+(9)/(3)+(5)	(2)	---	(40)	(4)+(9)/(3)+(5)	(2)	---	(40)	(4)+(9)/(3)+(5)

Kommunikationsbausteine

(3) TSEND/TRCV, TUSEND/TURCV

(5) TSEND_C/TRCV_C

(40) "PUT, GET" (beide Controller können Client oder Server sein)

21.5.5 S7-400 / S7-1500

Tabelle 21-26

Controller 2: S7-1500			Controller 1: S7-400 an PN/IE													
			CPU				CP									
			Schnittstelle: PN				443-1				443-1 Advanced					
											Schnittstelle: PN				X: IE	
			IOC, IOD				IOC, IOD				IOC, IOD				---	
			S7	OC	PN		S7	OC	PN		S7	OC	PN		S7	OC
PNIO	CBA	PNIO			CBA	PNIO			CBA							
CPU	Schnittstelle: PN	IOC, IOD	(1)	(3)/(3)+(5)	(2)	---	(1)	(4)+(9)/(3)+(5)	(2)	---	(1)	(4)+(9)/(3)+(5)	(2)	---	(1)	(4)+(9)/(3)+(5)
	Schnittstelle: IE	---	(1)	(3)/(3)+(5)	---	---	(1)	(4)+(9)/(3)+(5)	---	---	(1)	(4)+(9)/(3)+(5)	---	---	(1)	(4)+(9)/(3)+(5)
CP	CP 1543-1	---	(1)	(3)/(3)+(5)	---	---	(1)	(4)+(9)/(3)+(5)	---	---	(1)	(4)+(9)/(3)+(5)	---	---	(1)	(4)+(9)/(3)+(5)

[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

21.5.6 S7-400 / WinAC RTX

Tabelle 21-27

Controller 2: WinAC RTX auf PC WinAC RTX (F) 2010			Controller 1: S7-400 an PN/IE													
			CPU				CP									
			Schnittstelle: PN				443-1				443-1 Advanced					
											Schnittstelle: PN				X: IE	
			IOC, IOD				IOC, IOD				IOC, IOD				---	
S7	OC	PN		S7	OC	PN		S7	OC	PN		S7	OC			
		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	(1)	(4)+(9)/(3)
	Kontrolle von Windows: • „CP1616-CP1604“ • „IE-Allgemein“	---	(1)	---	---	---	(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

21.6 PN/IE: Controller 1 = S7-1200

21.6.1 (S7-1200 / ET 200 CPU)

Siehe ET 200 CPU / S7-1200: 21.3.4

21.6.2 (S7-1200 / S7-300)

Siehe S7-300 / S7-1200: 21.4.4

21.6.3 (S7-1200 / S7-400)

Siehe S7-400 / S7-1200: 21.5.4

21.6.4 S7-1200 / S7-1200

Tabelle 21-28

Controller 2: S7-1200			Controller 1: S7-1200 an PN/IE			
			CPU			
			Schnittstelle: PN			
			IOC			
			S7	OC	PN	
					PNIO	CBA
CPU	Schnittstelle: PN	IOC	(40)	(3)+(5)	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(3) TSEND/TRCV, TUSEND/TURCV

(5) TSEND_C/TRCV_C

(40) "PUT, GET" (beide Controller können Client oder Server sein)

21.6.5 S7-1200 / S7-1500

Tabelle 21-29

Controller 2: S7-1500			Controller 1: S7-1200 an PN/IE			
			CPU			
			Schnittstelle: PN			
			IOC			
			S7	OC	PN	
CPU	Schnittstelle: PN	IOC, IOD	(40)	(3)+(5)	(2)	---
	Schnittstelle: IE	---	(40)	(3)+(5)		---
CP	1543-1	---	(40)	(3)+(5)		---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(3) TSEND/TRCV, TUSEND/TURCV

(5) TSEND_C/TRCV_C

(40) "PUT, GET" (beide Controller können Client oder Server sein)

21.6.6 S7-1200 / WinAC RTX

Tabelle 21-30

Controller 2: WinAC RTX auf PC WinAC RTX (F) 2010			Controller 1: S7-1200 an PN/IE			
			CPU			
			Schnittstelle: PN			
			IOC			
			S7	OC	PN	
					PNIO	CBA
CPU/CP	Kontrolle von RTX (Submodule): • „CP1616/CP1604“ • „IE-Allgemein“	IOC	(40)	(3)+(5)/(3)	---	---
	Kontrolle von Windows: • „CP1616/CP1604“ • „IE-Allgemein“	---	(40)	---	---	---

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

(3) TSEND/TRCV, TUSEND/TURCV

(5) TSEND_C/TRCV_C

(40) "PUT, GET" (beide Controller können Client oder Server sein)

21.7 PN/IE: Controller 1 = S7-1500

21.7.1 (S7-1500 / ET 200 CPU)

Siehe ET 200 CPU / S7-1500: 21.3.5

21.7.2 (S7-1500 / S7-300)

Siehe S7-300 / S7-1500: 21.4.5

21.7.3 (S7-1500 / S7-400)

Siehe S7-400 / S7-1500: 21.5.5

21.7.4 (S7-1500 / S7-1200)

Siehe S7-1200 / S7-1500: 21.6.5

21.7.5 S7-1500 / S7-1500

Tabelle 21-31

Controller 2: S7-1500			Controller 1: S7-1500 an PN/IE											
			CPU								CP			
			Schnittstelle: PN				Schnittstelle: IE				1543-1			
			IOC, IOD				---				---			
			S7	OC		PN		S7	OC		PN		S7	OC
					PNIO	CBA			PNIO	CBA			PNIO	CBA
CPU	Schnittstelle: PN	IOC, IOD	(1)	(3)+(5)	(2)	---	(1)	(3)+(5)	---	---	(1)	(3)+(5)	---	---
	Schnittstelle: IE	---	(1)	(3)+(5)	---	---	(1)	(3)+(5)	---	---	(1)	(3)+(5)	---	---
CP	CP 1543-1	---	(1)	(3)+(5)	---	---	(1)	(3)+(5)	---	---	(1)	(3)+(5)	---	---

[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
- (5) TSEND_C/TRCV_C

21.7.6 S7-1500 / WinAC RTX

Tabelle 21-32

Controller 2: WinAC RTX			Controller 1: S7-1500 an PN/IE											
			CPU								CP			
			Schnittstelle: PN				Schnittstelle: IE				1543-1			
			IOC, IOD				---				---			
			S7	OC	PN		S7	OC	PN		S7	OC	PN	
					PNIO	CBA			PNIO	CBA			PNIO	CBA
CPU	RTX	IOC	(1)	(3)+(5)/(3)	(2)	---	(1)	(3)+(5)/(3)	---	---	(1)	(3)+(5)/(3)	---	---
	Windows	---	(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
- (5) TSEND_C/TRCV_C

21.8 PN/IE: Controller 1 = WinAC RTX

21.8.1 (WinAC RTX / ET 200 CPU)

Siehe ET 200 CPU / WinAC RTX: 21.3.6

21.8.2 (WinAC RTX / S7-300)

Siehe S7-300 / WinAC RTX: 21.4.6

21.8.3 (WinAC RTX / S7-400)

Siehe S7-400 / WinAC RTX: 21.5.6

21.8.4 (WinAC RTX / S7-1200)

Siehe S7-1200 / WinAC RTX: 21.6.6

21.8.5 (WinAC RTX / S7-1500)

Siehe S7-1500 / WinAC RTX: 21.7.6

21.8.6 WinAC RTX / WinAC RTX

Tabelle 21-33

Controller 2: WinAC RTX auf PC WinAC RTX (F) 2010			Controller 1: WinAC RTX an PN/IE WinAC RTX (F) 2010							
			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				---			
			S7	OC	PN		S7	OC	PN	
					PNIO	CBA			PNIO	CBA
CPU/CP	Kontrolle von RTX (Submodule):	IOC	(1)	(3)	---	x	(1)	---	---	---
	<ul style="list-style-type: none"> „CP1616/CP1604“ „IE-Allgemein“ 									
	Kontrolle von Windows:	---	(1)	---	---	---	(1)	---	---	---
	<ul style="list-style-type: none"> „CP1616/CP1604“ „IE-Allgemein“ 									

[Zurück zum Sprungverteiler PN/IE](#)

Kommunikationsbausteine

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

(3) TSEND/TRCV, TUSEND/TURCV

21.9 PN/IE: Übersicht Kommunikationsarten

Gegenüberstellung aller über PN/IE verfügbaren Kommunikationsarten.

Tabelle 21-34 Kommunikationsarten-Kompakt

	SIMATIC spezifisch	Offener Standard		
	S7-Kommunikation	Offene-Kommunikation		PN-Kommunikation
		Send/Receive-Bausteine	T-Bausteine T-Compact-Bausteine	PNIO
Protokolle	ISO (nur CP), IoT	ISO, IoT, TCP, UDP	IoT, TCP, UDP	PN
Schnittstellen	CPU, CP, CM	CP	CPU, CP, CM	CPU, CP, CM
Kommunikationsbausteine (max. Daten)	BSEND (≤ 64 KByte) Typ "USEND/URCV" (≥ 160 Byte) Typ "PUT, GET" (≥ 160 Byte)	AG_xSEND (ISO, IoT, TCP ≤ 8 KByte) (UDP ≤ 2 KByte) Server für FETCH, WRITE (nicht UDP)	TSEND, TSEND_C, ... (IoT ≤ 32 KByte) (TCP ≤ 64 KByte) (UDP = 1472 Byte)	Ladebefehle/Transferbefehle DPR_DAT, DPWR_DAT PNIO_SEND, PNIO_RECV
remote Quittierung	BSEND: Applikation Typ "USEND / URCV": Transport Typ "PUT, GET": Applikation	ISO, IoT, TCP: Transport UDP: keine	IoT, TCP: Transport UDP: keine	Applikation
Verbindungen?	ja	ISO, IoT, TCP : ja UDP: nein	IoT, TCP: ja UDP: nein	nein

[Zurück zum Sprungverteiler PN/IE](#)

Die Daten sind den Tabellen Kommunikationsarten-Detail entnommen:

- S7-Kommunikation (Tabelle 32-1), PN-Kommunikation (Tabelle 37-1)
- Offene Kommunikation mit Send/Receive-Bausteinen (Tabelle 34-3), Offene Kommunikation mit T-Bausteinen (Tabelle 35-3)

22 Auswahlhilfe: PROFIBUS (PB)

22.1 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)

22.2 PB: Schnittstellen und Kommunikationsarten

22.2.1 ET 200 CPU an PB

Tabelle 22-1

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(F) CPU	DP-Slave (*1)	aktiv	I_PUT, I_GET, Server	"PUT, GET", Server	(2)
				passiv	I_PUT, I_GET, Server	---	(2)
	CP	DP-Mastermodul (*2)	DP-Master		I_PUT, I_GET	"PUT, GET", Server	(2)
ET 200 Pro	CPU	IM154-8(F) PN/DP CPU	DP-Master		I_PUT, I_GET	"PUT, GET", Server	(2)
			DP-Slave (*1)	aktiv	I_PUT, I_GET, Server	"PUT, GET", Server	(2)
				passiv	I_PUT, I_GET, Server	---	(2)
	CP	DP-Mastermodul	DP-Master		I_PUT, I_GET	"PUT, GET", Server	(2)

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

Erläuterungen zur Tabelle

Zu (*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 DP-Schnittstelle in STEP 7.

Zu (*2): für IM151-7(F) CPU oder IM151-8(F) PN/DP CPU

22.2.2 S7-300 an PB

Tabelle 22-2

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

Zu (*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 DP-Schnittstelle in STEP 7.

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

Zu (*3): 342-5 steht für die Varianten: CP 342-5, CP 342-5 FO

22.2.3 S7-400 an PB

Tabelle 22-3

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

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

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

(8) AG_SEND/AG_RECV

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(10) READ, WRITE, REPORT

Erläuterungen zur Tabelle

Zu (*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 DP-Schnittstelle in STEP 7.

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

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

22.2.4 S7-1200 an PB

Tabelle 22-4

Controller an PB: S7-1200		Funktionalität der Schnittstelle	Kommunikationsart				
			SIMATIC spezifisch		Offener Standard		
			S7-Basiskommunikation	S7-Kommunikation	Offene-Kommunikation	FMS-Kommunikation	DP-Kommunikation
CM	CM 1242-5	DP-Slave	---	---	---	---	(2)
	CM 1243-5	DP-Master	---	"PUT, GET"	---	---	(2)

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

22.2.5 S7-1500 an PB

Tabelle 22-5

Controller an PB: S7-1500			Funktionalität der Schnittstelle		Kommunikationsart				
					SIMATIC spezifisch		Offener Standard		
					S7-Basiskommunikation	S7-Kommunikation	Offene-Kommunikation	FMS-Kommunikation	DP-Kommunikation
CPU	1516-3 PN/DP	Schnittstelle: DP	DP-Master		---	(1)	---	---	(2)
CM	CM 1542-5		DP-Master		---	(1)	---	---	(2)
			DP-Slave (*1)	aktiv	---	(1)	---	---	(2)
				passiv	---	"PUT, GET", Server	---	---	(2)

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

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

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

Erläuterungen zur Tabelle

Zu (*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 DP-Schnittstelle in STEP 7.

22.2.6 S7-mEC an PB

Tabelle 22-6

Controller an PB: S7-mEC mit WinAC RTX (F) 2010			Funktionalität der Schnittstelle	Kommunikationsart	
				SIMATIC spezifisch	Offener Standard
				S7-Kommunikation	DP-Kommunikation
CP	EM PCI-104	unter Kontrolle von RTX (Submodule): CP 5603	DP-Master	(1)	(2)

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

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

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

22.2.7 Box PC an PB

Tabelle 22-7

Controller an PB: Box PC SIMATIC Embedded Bundles: : HMI IPC427C mit WinAC RTX (F) 2010		Funktionalität der Schnittstelle	Kommunikationsart	
			SIMATIC spezifisch	Offener Standard
			S7-Kommunikation	DP-Kommunikation
CPU/CP (*1)	unter Kontrolle von RTX (Submodule): • „CP5611/CP5621“ • „CP5613/CP5603/CP5623“	DP-Master	(1)	(2)

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

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

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

Erläuterungen zur Tabelle

Zu (*1): 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 22-8

Schnittstelle	CP (steckbar in PC)	CPU (Onboard Schnittstelle des PC)
„CP5611/CP5621“	CP 5611, CP 5611-A2, CP5621	Onboard PB-Schnittstelle der SIMATIC IPC: CP 5611 integriert
„CP5613/CP5603/CP5623“	CP 5613, CP 5613-A2, CP 5603, CP 5623	---

22.2.8 Panel PC an PB

Tabelle 22-9

Controller an PB: Box PC SIMATIC Embedded Bundles: HMI IPC477C mit WinAC RTX (F) 2010		Funktionalität der Schnittstelle	Kommunikationsart	
			SIMATIC spezifisch	Offener Standard
			S7-Kommunikation	DP-Kommunikation
CPU/CP (*1)	unter Kontrolle von RTX (Submodule): • „CP5611/CP5621“ • „CP5613/CP5603/CP5623“	DP-Master	(1)	(2)

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

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

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

Erläuterungen zur Tabelle

Zu (*1): 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 22-10

Schnittstelle	CP (steckbar in PC)	CPU (Onboard Schnittstelle des PC)
„CP5611/CP5621“	CP 5611, CP 5611-A2, CP5621	Onboard PB-Schnittstelle der SIMATIC IPC: CP 5611 integriert
„CP5613/CP5603/CP5623“	CP 5613, CP 5613-A2, CP 5603, CP 5623	---

22.2.9 WinAC RTX an PB

Tabelle 22-11

Controller an PB: WinAC RTX WinAC RTX (F) 2010		Funktionalität der Schnittstelle	Kommunikationsart	
			SIMATIC spezifisch	Offener Standard
			S7-Kommunikation	DP-Kommunikation
CPU / CP (*1)	unter Kontrolle von RTX (Submodule): <ul style="list-style-type: none"> „CP5611/CP5621“ „CP5613/CP5603/CP5623“ 	DP-Master	(1)	(2)

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

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

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

Erläuterungen zur Tabelle

Zu (*1): 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 22-12

Sammelbezeichnung	PC Karte (entspricht CP Schnittstelle)	Onboard Schnittstelle des PC (entspricht CPU Schnittstelle)
„CP5611/CP5621“	CP 5611, CP 5611-A2, CP5621	Onboard PB-Schnittstelle der SIMATIC IPC: CP 5611 integriert
„CP5613/CP5603/CP5623“	CP 5613, CP 5613-A2, CP 5603, CP 5623	---

22.3 PB: Controller 1 = ET 200 CPU

22.3.1 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 22.2).

Tabelle 22-13

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

22.3.2 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 22.2).

Tabelle 22-14

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

22.3.3 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 22.2).

Tabelle 22-15

Controller 2: S7-400			Controller 1: ET 200 CPU an PB														
			ET200 S						ET 200 Pro								
			CPU			CP			CPU						CP		
			IM151-7 (F) CPU			DP-Mastermodul			IM154-8 (F) PN/DP CPU						DP-Mastermodul		
			DP-Slave passiv			DP-Master			DP-Master			DP-Slave passiv			DP-Master		
			S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7-B	S7	DP
CPU	Schnittstelle: DP, MPI/DP	DP-Master	(61)	---	(2)	---	(21)	---	---	(21)	---	(61)	---	(2)	---	(21)	---
		DP-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)	---
		DP-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)

22.3.4 ET 200 CPU / S7-1200

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 22.2).

Tabelle 22-16

Controller 2: S7-1200			Controller 1: ET 200 CPU an PB														
			ET200 S						ET 200 Pro								
			CPU			CP			CPU						CP		
			IM151-7 (F) CPU			DP-Mastermodul			IM154-8 (F) PN/DP CPU						DP-Mastermodul		
			DP-Slave passiv			DP-Master			DP-Master			DP-Slave passiv			DP-Master		
CM			S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7-B	S7	DP
	CM 1242-5	DP-Slave	---	---	---	---	---	(2)	---	---	(2)	---	---	---	---	---	(2)
	CM 1243-5	DP-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")

22.3.5 ET 200 CPU / S7-1500

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 22.2).

Tabelle 22-17

Controller 2: S7-1500			Controller 1: ET 200 CPU an PB														
			ET200 S						ET 200 Pro								
			CPU			CP			CPU						CP		
			IM151-7 (F) CPU			DP-Mastermodul			IM154-8 (F) PN/DP CPU						DP-Mastermodul		
			DP-Slave passiv			DP-Master			DP-Master			DP-Slave passiv			DP-Master		
			S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7 B	S7	DP	S7-B	S7	DP
CPU	Schnittstelle: DP	DP-Master	---	---	(2)	---	(21)	---	---	(21)	---	---	---	(2)	---	(21)	---
CM	1542-5 (DP)	DP-Master	---	---	(2)	---	(21)	---	---	(21)	---	---	---	(2)	---	(21)	---
		DP-Slave passiv	---	---	---	---	---	(2)	---	---	(2)	---	---	---	---	---	(2)

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

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

22.3.6 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 22.2).

Tabelle 22-18

Controller 2: WinAC RTX auf PC			Controller 1: ET 200 CPU an PB														
WinAC RTX (F) 2010			ET200 S						ET 200 Pro								
			CPU			CP			CPU						CP		
			IM151-7 (F) CPU			DP-Mastermodul			IM154-8 (F) PN/DP CPU						DP-Mastermodul		
			DP-Slave passiv			DP-Master			DP-Master			DP-Slave passiv			DP-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):	DP-Master	---	---	(2)	---	(21)	---	---	(21)	---	---	---	(2)	---	(21)	---
	<ul style="list-style-type: none"> „CP5611/CP5621“ „CP5613CP5603/CP5623“ 																

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

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

22.4 PB: Controller 1 = S7-300

22.4.1 (S7-300 / ET 200 CPU)

Siehe ET 200 CPU / S7-300: 22.3.2

22.4.2 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 22.2).

Tabelle 22-19

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

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(7) DP_SEND, DP_RECV

(8) AG_SEND/AG_RECV

(10) READ, WRITE, REPORT

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

(22) Controller 2 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)

22.4.3 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 22.2).

Tabelle 22-20

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

[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
- (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)

22.4.4 S7-300 / S7-1200

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 22.2).

Tabelle 22-21

Controller 2: S7-1200			Controller 1: S7-300 an PB																
			CPU							CP									
			Schnittstelle: DP, MPI/DP							342-5							343-5		
			DP-Master			DP-Slave passiv				kein DP		DP-Master			DP-Slave passiv			kein DP	
			S7 B	S7	DP	S7 B	S7	DP	S7	OC	S7	OC	DP	S7	OC	DP	S7	OC	FMS
CM	CM 1242-5	DP-Slave	---	---	(2)	---	---	---	---	---	---	---	(7)/(2)	---	---	---	---	---	---
	CM 1243-5	DP-Master	---	(21)	---	---	---	(2)	(40)	---	(40)	---	---	---	---	(7)/(2)	(21)	---	---

[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")

(40) "PUT, GET" (beide Controller können Client oder Server sein)

22.4.5 S7-300 / S7-1500

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 22.2).

Tabelle 22-22

Controller 2: S7-1500			Controller 1: S7-300 an PB																
			CPU							CP									
			Schnittstelle: DP, MPI/DP							342-5							343-5		
			DP-Master			DP-Slave passiv				kein DP		DP-Master			DP-Slave passiv			kein DP	
			S7 B	S7	DP	S7 B	S7	DP	S7	OC	S7	OC	DP	S7	OC	DP	S7	OC	FMS
CPU	Schnittstelle: DP	DP-Master	---	(21)	---	---	---	(2)	(1)	---	(1)	---	---	---	---	(7)/(2)	(21)	---	---
CM	1542-5	DP-Master	---	(21)	---	---	---	(2)	(1)	---	(1)	---	---	---	---	(7)/(2)	(21)	---	---
		DP-Slave passiv	---	---	(2)	---	---	---	---	---	(22)	---	(7)/(2)	---	---	---	---	---	---

[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
- (21) Controller 1 ist Server (für "PUT, GET")
- (22) Controller 2 ist Server (für "PUT, GET")

22.4.6 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 22.2).

Tabelle 22-23

Controller 2: WinAC RTX WinAC RTX (F) 2010			Controller 1: S7-300 an PB																
			CPU							CP									
			Schnittstelle: DP, MPI/DP							342-5									
			DP-Master			DP-Slave passiv				kein DP		DP-Master			DP-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):	DP-Master	---	(21)	---	---	---	(2)	(1)	---	(1)	---	---	---	---	(7)/(2)	(21)	---	---
	<ul style="list-style-type: none"> „CP5611/CP5621“ „CP5613CP5603/CP5623“ 																		

[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

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

22.5 PB: Controller 1 = S7-400

22.5.1 (S7-400 / ET 200 CPU)

Siehe ET 200 CPU / S7-400: 22.3.3

22.5.2 (S7-400 / S7-300)

Siehe S7-300 / S7-400: 22.4.3

22.5.3 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 22.2).

Tabelle 22-24

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

[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
- (21) Controller 1 ist Server (für "PUT, GET")
- (22) Controller 2 ist Server (für "PUT, GET")

22.5.4 S7-400 / S7-1200

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 22.2).

Tabelle 22-25

Controller 2: S7-1200			Controller 1: S7-400 an PB													
			CPU						CP							
			Schnittstelle: DP, MPI/DP						443-5 Basic			443-5 Ext.				
			DP-Master			DP-Slave passiv			kein DP			kein DP		DP-Master		
			S7 B	S7	DP	S7 B	S7	DP	S7	OC	FMS	S7	OC	S7	OC	DP
CM	CM 1242-5	DP-Slave	---	---	(2)	---	---	---	---	---	---	---	---	---	(2)	
	CM 1243-5	DP-Master	---	(40)	---	---	---	(2)	(40)	---	---	(40)	---	(40)	---	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(40) "PUT, GET" (beide Controller können Client oder Server sein)

22.5.5 S7-400 / S7-1500

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 22.2).

Tabelle 22-26

Controller 2: S7-1500			Controller 1: S7-400 an PB													
			CPU						CP							
			Schnittstelle: DP, MPI/DP						443-5 Basic			443-5 Ext.				
			DP-Master			DP-Slave passiv			kein DP			kein DP		DP-Master		
			S7 B	S7	DP	S7 B	S7	DP	S7	OC	FMS	S7	OC	S7	OC	DP
CPU	Schnittstelle: DP	DP-Master	---	(1)	---	---	---	(2)	(1)	---	---	(1)	---	(1)	---	---
CP	1542-5	DP-Master	---	(1)	---	---	---	(2)	(1)	---	---	(1)	---	(1)	---	---
		DP-Slave passiv	---	(22)	(2)	---	---	---	(22)	---	---	(22)	---	(22)	---	(2)

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

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

22.5.6 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 22.2).

Tabelle 22-27

Controller 2: WinAC RTX WinAC RTX (F) 2010			Controller 1: S7-400 an PB													
			CPU						CP							
			Schnittstelle: DP, MPI/DP						443-5 Basic			443-5 Ext.				
			DP-Master			DP-Slave passiv			kein DP			kein DP		DP-Master		
			S7 B	S7	DP	S7 B	S7	DP	S7	OC	FMS	S7	OC	S7	OC	DP
CPU/CP	Kontrolle von RTX (Submodule): <ul style="list-style-type: none">„CP5611/CP5621“„CP5613CP5603/CP5623“	DP-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

22.6 PB: Controller 1 = S7-1200

22.6.1 (S7-1200 / ET 200 CPU)

Siehe ET 200 C PU / S7-1200: 22.3.4

22.6.2 (S7-1200 / S7-300)

Siehe S7-300 / S7-1200: 22.4.4

22.6.3 (S7-1200 / S7-400)

Siehe S7-400 / S7-1200: 22.5.4

22.6.4 S7-1200 / S7-1200

Tabelle 22-28

Controller 2: S7-1200			Controller 1: S7-1200 an PB			
			CM			
			CM 1243-5		CM 1242-5	
			DP-Master		DP-Slave	
			S7	DP	S7	DP
CM	CM 1242-5	DP-Slave	---	(2)	---	---
	CM 1243-5	DP-Master	(40)	---	---	(2)

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(40) "PUT, GET" (beide Controller können Client oder Server sein)

22.6.5 S7-1200 / S7-1500

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 22.2).

Tabelle 22-29

Controller 2: S7-1500			Controller 1: S7-1200 an PB			
			CM			
			CM 1243-5		CM 1242-5	
			DP-Master		DP-Slave	
			S7	DP	S7	DP
CPU	Schnittstelle: DP	DP-Master	(40)	---	---	(2)
CM	1542-5	DP-Master	(40)	---	---	(2)
		DP-Slave passiv	(22)	(2)	---	---

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

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

(40) "PUT, GET" (beide Controller können Client oder Server sein)

22.6.6 S7-1200 / WinAC RTX

Tabelle 22-30

Controller 2: WinAC RTX			Controller 1: S7-1200 an PB			
WinAC RTX (F) 2010			CM			
			CM 1243-5		CM 1242-5	
			DP-Master		DP-Slave	
			S7	DP	S7	DP
CPU/CP	Kontrolle von RTX (Submodule):	DP-Master	(40)	---	---	(2)
	<ul style="list-style-type: none"> „CP5611/CP5621“ „CP5613CP5603/CP5623“ 					

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

(2) Lade/Transfer Befehle, DPRD_DAT, DPWR_DAT

(40) "PUT, GET" (beide Controller können Client oder Server sein)

22.7 PB: Controller 1 = S7-1500

22.7.1 (S7-1500 / ET 200 CPU)

Siehe ET 200 CPU / S7-1500: 22.3.5

22.7.2 (S7-1500 / S7-300)

Siehe S7-300 / S7-1500: 22.4.5

22.7.3 (S7-1500 / S7-400)

Siehe S7-400 / S7-1500: 22.5.5

22.7.4 (S7-1500 / S7-1200)

Siehe S7-1200 / S7-1500: 22.6.5

22.7.5 S7-1500 / S7-1500

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 22.2).

Tabelle 22-31

Controller 2: S7-1500			Controller 1: S7-1500 an PB								
			CPU			CM					
			Schnittstelle: DP			1542-5					
			DP-Master			DP-Master			DP-Slave passiv		
			S7 B	S7	DP	S7	OC	DP	S7	OC	DP
CPU	Schnittstelle: DP	DP-Master	---	(1)	---	(1)	---	---	(21)	---	(2)
CP	1542-5	DP-Master	---	(1)	---	(1)	---	---	(21)	---	(2)
		DP-Slave passiv	---	(22)	(2)	(22)	---	(2)	---	---	---

[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")
- (22) Controller 2 ist Server (für "PUT, GET")

22.7.6 S7-1500 / 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 22.2).

Tabelle 22-32

Controller 2: WinAC RTX			Controller 1: S7-1500 an PB								
WinAC RTX (F) 2010			CPU			CM					
			Schnittstelle: DP			1542-5					
			DP-Master			DP-Master			DP-Slave passiv		
			S7 B	S7	DP	S7	OC	DP	S7	OC	DP
CPU/CP	Kontrolle von RTX (Submodule): • „CP5611/CP5621“ • „CP5613CP5603/CP5623“	DP-Master	---	(1)	---	(1)	---	---	(21)	---	(2)

[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")

22.8 PB: Controller 1 = WinAC RTX

22.8.1 (WinAC RTX / ET 200 CPU)

Siehe ET 200 CPU / WinAC RTX: 22.3.6

22.8.2 (WinAC RTX / S7-300)

Siehe S7-300 / WinAC RTX: 22.4.6

22.8.3 (WinAC RTX / S7-400)

Siehe S7-400 / WinAC RTX: 22.5.6

22.8.4 (WinAC RTX / S7-1200)

Siehe S7-1200 / WinAC RTX: 22.6.6

22.8.5 (WinAC RTX / S7-1500)

Siehe S7-1500 / WinAC RTX: 22.7.6

22.8.6 WinAC RTX / WinAC RTX

Tabelle 22-33

Controller 2: WinAC RTX WinAC RTX (F) 2010			Controller 1: WinAC RTX an PB WinAC RTX (F) 2010	
			CPU/CP	
			Kontrolle von RTX (Submodule):	
			<ul style="list-style-type: none"> „CP5611/CP5621“ „CP5613CP5603/CP5623“ 	
			DP-Master	
			S7	DP
CPU/CP	Kontrolle von RTX (Submodule):	DP-Master	(1)	---
	<ul style="list-style-type: none"> „CP5611/CP5621“ „CP5613CP5603/CP5623“ 			

[Zurück zum Sprungverteiler PB](#)

Kommunikationsbausteine

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

22.9 PB: Übersicht Kommunikationsarten

Gegenüberstellung aller über PB verfügbaren Kommunikationsarten.

Tabelle 22-34 Kommunikationsarten-Kompakt

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

[Zurück zum Sprungverteiler PB](#)

Die Daten sind den Tabellen Kommunikationsarten-Detail entnommen:

- S7-Basiskommunikation (Tabelle 31-1), S7-Kommunikation (Tabelle 32-1)
- Offene Kommunikation mit Send/Receive-Bausteinen (Tabelle 39-1)
- FMS-Kommunikation (Tabelle 40-1), DP-Kommunikation Tabelle 41-1)

23 Auswahlhilfe: MPI

23.1 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)

23.2 MPI: Schnittstellen und Kommunikationsarten

23.2.1 ET 200 CPU an MPI

Tabelle 23-1

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

[Zurück zum Sprungverteiler MPI](#)

Kommunikationsbausteine

(11) X_SEND/X_RCV, X_PUT, X_GET

23.2.2 S7-300 an MPI

Tabelle 23-2

Controller an MPI: S7-300		Kommunikationsart		
		SIMATIC spezifisch		
		Globaldaten	S7-Basiskommunikation	S7-Kommunikation
CPU	Schnittstelle: MPI, MPI/DP (*1)	zyklisch	(11)	"PUT, GET", Server

[Zurück zum Sprungverteiler MPI](#)

Kommunikationsbausteine

(11) X_SEND/X_RCV, X_PUT, X_GET

Erläuterungen zur Tabelle

Zu (*1): MPI/DP Schnittstelle in der Betriebsart MPI

23.2.3 S7-400 an MPI

Tabelle 23-3

Controller an MPI: S7-400		Kommunikationsart		
		SIMATIC spezifisch		
		Globaldaten	S7-Basiskommunikation	S7-Kommunikation
CPU	Schnittstelle: MPI/DP (*1)	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

Zu (*1): MPI/DP Schnittstelle in der Betriebsart MPI

23.3 MPI: Controller 1 = ET 200 CPU

23.3.1 ET 200 CPU / ET 200 CPU

Tabelle 23-4

Controller 2: ET 200 CPU		Controller 1: ET 200 CPU an MPI		
		CPU		
		IM151-7 (F) 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

23.3.2 ET 200 CPU / S7-300

Tabelle 23-5

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

[Zurück zum Sprungverteiler MPI](#)

Kommunikationsbausteine

(11) X_SEND/X_RCV, X_PUT, X_GET

23.3.3 ET 200 CPU / S7-400

Tabelle 23-6

Controller 2: S7-400		Controller 1: ET 200 CPU an MPI		
		CPU		
		IM151-7 (F) CPU, IM154-8 (F) PN/DP CPU		
		GD	S7 Basis	S7
CPU	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")

23.4 MPI: Controller 1 = S7-300

23.4.1 (S7-300 / ET 200 CPU)

Siehe ET 200 CPU / S7-300: 23.3.2

23.4.2 S7-300 / S7-300

Tabelle 23-7

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

[Zurück zum Sprungverteiler MPI](#)

Kommunikationsbausteine

(11) X_SEND/X_RCV, X_PUT, X_GET

23.4.3 S7-300 / S7-400

Tabelle 23-8

Controller 2: S7-400		Controller 1: S7-300 an MPI		
		CPU		
		alle mit Schnittstelle: MPI, MPI/DP		
		GD	S7 Basis	S7
CPU	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")

23.5 MPI: Controller 1 = S7-400

23.5.1 (S7-400 / ET 200 CPU)

Siehe ET 200 CPU / S7-400: 23.3.3

23.5.2 (S7-400 / S7-300)

Siehe S7-300 / S7-400: 23.4.3

23.5.3 S7-400 / S7-400

Tabelle 23-9

Controller 2: S7-400		Controller 1: S7-400 an MPI		
		CPU		
		alle mit Schnittstelle: MPI/DP		
		GD	S7 Basis	S7
CPU	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

23.6 MPI: Übersicht Kommunikationsarten

Gegenüberstellung aller über MPI verfügbaren Kommunikationsarten.

Tabelle 23-10 Kommunikationsarten-Kompakt

	SIMATIC spezifisch		
	Globaldaten-Kommunikation	S7-Basiskommunikation	S7-Kommunikation
Protokolle	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) Typ "USEND/URCV" (≥ 160 Byte) Typ "PUT, GET" (≥ 160 Byte)
remote Quittierung	nein	Applikation	BSEND: Applikation Typ "USEND/URCV": Transport Typ "PUT, GET": Applikation
Verbindungen?	nein	ja	ja

[Zurück zum Sprungverteiler MPI](#)

Die Daten sind den Tabellen Kommunikationsarten-Detail entnommen:

- Globaldaten-Kommunikation (Tabelle 30-1)
- S7-Basiskommunikation (Tabelle 31-1, S7-Kommunikation (Tabelle 32-1)

24 Auswahlhilfe: SIMATIC Rückwandbus

24.1 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

Zu (*1)

Betrachtete Kombinationen Controller 1 / Controller 2:

Bei der S7-400 können bis zu 4 CPUs in einem einzigen Controller betrieben werden (Multicomputing, Kapitel 5.5). D.h. in den folgenden Tabellen gibt es keine Unterscheidung zwischen Controller 1 und Controller 2. Die CPUs stecken im gleichen Controller.

24.2 SIMATIC Rückwandbus: Schnittstellen und Kommunikationsarten

Tabelle 24-1

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

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

Tabelle 24-2

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

24.4 Übersicht Kommunikationsarten

Gegenüberstellung aller über den SIMATIC Rückwandbus verfügbaren Kommunikationsarten.

Tabelle 24-3 Kommunikationsarten-Kompakt

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

[Zurück zum Sprungverteiler Rückwandbus](#)

Die Daten sind den Tabellen Kommunikationsarten-Detail entnommen:

- Globaldaten-Kommunikation (Tabelle 30-1)
- S7-Basiskommunikation (Tabelle 31-1)

25 Auswahlhilfe: Serielle Schnittstelle (PtP)

25.1 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.

25.2 ET 200 CPU an PtP

Tabelle 25-1

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

Zusätzliche Möglichkeit: In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden (Kapitel 25.11).

[Zurück zum Sprungverteiler SERIELL](#)

Erläuterungen zur Tabelle

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

25.3 S7-300 an PtP

Tabelle 25-2

Controller an PtP: S7-300		Protokoll		maximale Anzahl Daten	Physik (max. Abstand Kommunikationspartner)	Übertragungsrate
		Typ	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 Funktionscode (*1)		bei 20mA: bis 19,2 kBit/s

Zusätzliche Möglichkeit: In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden (Kapitel 25.11).

[Zurück zum Sprungverteiler SERIELL](#)

Erläuterungen zur Tabelle

Zu (*1): Siehe dazu die entsprechenden Handbücher.

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

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

25.4 S7-400 an PtP

Tabelle 25-3

Controller an PtP: S7-400		Protokolle		max. Anzahl Daten	Physik (max. Abstand Kommunikationspartner)	Übertragungsrate
		Typ	integriert / ladbar			
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	ASCII, 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), Modbus Master und Modbus Slave	ladbar	abhängig vom Funktionscode (*3)		

Zusätzliche Möglichkeit: In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden (Kapitel 25.11).

[Zurück zum Sprungverteiler SERIELL](#)

Erläuterungen zur Tabelle

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

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

Zu (*3): Siehe dazu die entsprechenden Handbücher.

25.5 S7-1200 an PtP

Tabelle 25-4

Controller an PtP: S7-1200		Protokoll		maximale Anzahl Daten	Physik (max. Abstand Kommunikationspartner)	Übertragungsrate
		Typ	integriert / ladbar			
CPU	Schnittstelle: PN	Modbus/TCP, Modbus Client und Server	integriert			
	CB 1241 (*3)	Modbus Seriell (RTU Format), Modbus Master und Slave	integriert		1000 m	300 Bit/s bis 115,2 kBit/s
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 115,2 kBit/s

Zusätzliche Möglichkeit: In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden (Kapitel 25.11).

[Zurück zum Sprungverteiler SERIELL](#)

Erläuterungen zur Tabelle

Zu (*1): Auswahl der Physik: Unterschiedliche Baugruppenvarianten

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

Zu (*3) Die Baugruppe wird in die CPU gesteckt.

25.6 S7-1500 an PtP

Tabelle 25-5

Controller an PtP: S7-1500			Protokoll		maximale Anzahl Daten	Physik (max. Abstand Kommunikationspartner)	Übertragungsrate
			Typ	integriert / ladbar			
CP	CM PtP RS232	BA	3964(R), Freeport, USS (*1)	integriert	1024 Byte	RS 232 inkl. Begleitsignale (15 m)	300 bis 19200 bit/s
		HF	3964(R), Freeport, USS (*1) Modbus RTU	integriert	4096 Byte	RS 232 inkl. Begleitsignale (15 m)	300 bis 115200 bit/s
	CM PtP RS422/485	BA	3964(R), Freeport, USS (*1)	integriert	1024 Byte	RS 422/485 (1200 m)	300 bis 19200 bit/s
		HF	3964(R), Freeport, USS (*1) Modbus RTU	integriert	4096 Byte	RS 422/485 (1200 m)	300 bis 115200 bit/s

Zusätzliche Möglichkeit: In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden (Kapitel 25.11).

[Zurück zum Sprungverteiler SERIELL](#)

Erläuterungen zur Tabelle

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

25.7 S7-mEC an PtP

Tabelle 25-6

Controller an PtP: S7-mEC		Eigenschaften
CP	CP 340	Siehe Kapitel 25.3: S7-300 / CP 340

Zusätzliche Möglichkeit: In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden (Kapitel 25.11).

[Zurück zum Sprungverteiler SERIELL](#)

25.8 Box PC an PtP

In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden (Kapitel 25.11).

[Zurück zum Sprungverteiler SERIELL](#)

25.9 Panel PC an PtP

In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden (Kapitel 25.11).

[Zurück zum Sprungverteiler SERIELL](#)

25.10 WinAC RTX an PtP

In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden (Kapitel 25.11).

[Zurück zum Sprungverteiler SERIELL](#)

25.11 Dezentrale Station ET 200

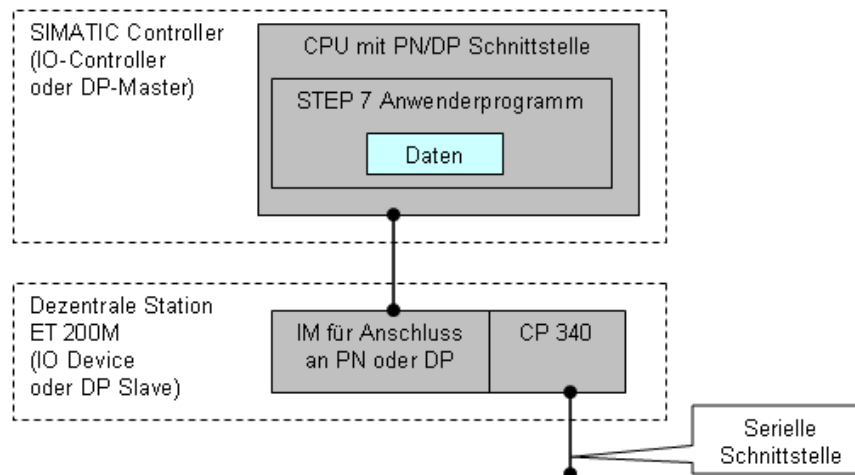
In dezentralen ET 200 Stationen können serielle Schnittstellenbaugruppen eingesetzt werden.

Tabelle 25-7

Serielle Schnittstellenbaugruppe	dezentrale Station	Medium
1-SI Modul	ET 200S	PROFINET, PROFIBUS
CP340, CP341	ET 200M	PROFINET, PROFIBUS
CM PtP RS232 BA CM PtP RS232 HF CM PtP RS422/485 BA CM PtP RS422/485 HF	ET 200MP	PROFINET
ET 200SP CM PtP	ET 200SP	PROFINET

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 folgende Abbildung zeigt ein Beispiel.

Abbildung 25-1



[Zurück zum Sprungverteiler SERIELL](#)

26 Informationen zu Teil 2

In den Tabellen sind Verweise auf Informationen zu den Themen aus Teil 2 (Auswahlhilfe) zu finden. Alle Verweise /x/ sind zentral im Kapitel 61 hinterlegt. Dort sind auch die entsprechenden Links ins Internet zu finden.

Kommunikation

Die folgende Tabelle enthält Verweise zu Handbüchern mit Details zu den Schnittstellen der Kommunikationspartner.

Tabelle 26-1

/x/	Titel	Informationen zu
/7/	CPU 31xC und CPU 31xTechnische Daten, Gerätehandbuch	Kommunikation über: MPI, PB, IE
/8/	Automatisierungssystem S7-400 CPU-Daten, Gerätehandbuch	
/34/	SIMATIC S7 S7-1200 Automatisierungssystem, Systemhandbuch	
/35/	SIMATIC S7 S7-1500 Automatisierungssystem, Systemhandbuch	
/36/	SIMATIC WinAC RTX (F) 2010, Betriebsanleitung	
/37/	PROFINET Systemanschluss für SIMATIC S7, Handbücher	
/38/	PROFIBUS Systemanschluss für SIMATIC S7, Handbücher	Kommunikation über: Serielle Schnittstelle
/39/	Serielle Kommunikation	
/21/	S7-300 CPU 31xC Technologische Funktionen, Handbuch	

Applikationsbeispiele

Die folgende Tabelle enthält Verweise zu Applikationsbeispielen.

Tabelle 26-2

/x/	Titel / Inhalt
/200/	Applikationsbeispiele zum Thema Kommunikation

Leistungsdaten

Die folgende Tabelle enthält einen Verweis zu den Ergebnissen von Messungen:

Tabelle 26-3

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

Beispiele für Messungen:

- Übertragungszeit für typische Konfigurationen am Industrial Ethernet
- PN-Reaktionszeit für typische Konfigurationen am PROFINET IO

27 ***** TEIL 3: Kommunikationsarten *****

27.1 Gliederung und Inhalt

Tabelle 27-1

Kapitel	Gliederung		Inhalt
29	SIMATIC S7 spezifische Kommunikation		Übersicht
30		Globaldaten-Kommunikation	Beschreibung
31		S7-Basiskommunikation	
32		S7-Kommunikation	
33	PROFINET/Industrial Ethernet		Übersicht
34		Offene-Kommunikation mit Send/Receive-Bausteinen	Beschreibung
35		Offene-Kommunikation mit T-Bausteinen	
36		PN-Kommunikation: CBA	
37		PNIO	
38	PROFIBUS		Übersicht
39		Offene-Kommunikation mit Send/Receive-Bausteinen	Beschreibung
40		FMS-Kommunikation	
41		DP-Kommunikation	
42	Serielle Schnittstelle		Übersicht
43		Überblick Anwenderschnittstellen	Tabelle mit allen Schnittstellen
43 bis 54		ET 200S, S7-300, S7-400, S7-1200, S7-1500	Beschreibung der Schnittstellen
55	Informationen		Handbücher, FAQs, Applikationen

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

Tabelle 27-2

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

28 Vorbemerkungen

Die Kommunikationsarten werden nach einem einheitlichen Schema beschrieben. Für jede Kommunikationsart gibt es die folgenden Kapitel:

- Merkmale
- Details Kommunikationsart
- Überblick Anwenderschnittstellen
- Anwenderschnittstellen

Im Folgenden wird beschrieben, welche Inhalte diese Kapitel haben.

28.1 Kapitel: Merkmale

In diesem Kapitel wird die Kommunikationsart stichpunktartig charakterisiert.

28.2 Kapitel: Details Kommunikationsart

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

28.2.1 Zweck der Tabelle

Die Tabelle beantwortet folgende Fragen:

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

Hinweis

Die Tabellen Kommunikationsart-Detail sind Grundlage für eine weitere Tabelle mit der Bezeichnung Kommunikationsarten-Kompakt (19.4).

In der Tabelle Kommunikationsarten-Kompakt werden alle Kommunikationsarten eines Mediums (PN/IE, ...) zusammengefasst.

28.2.2 Aufbau der Tabelle

Der Aufbau der Tabelle Kommunikationsarten-Detail wird am Beispiel erläutert:

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

Das Bild zeigt einen Ausschnitt der Tabelle (Tabelle 35-3).

Abbildung 28-1

1	Kommunikationsart:		Offene-Kommunikation mit T-Bausteinen (Netz PN/IE)	
	Protokoll:	ISO on TCP		TCP
	Allgemeines			
	Schnittstellen		CPU, CP, CM	CPU, CP, CM
2	Anbindung	SIMATIC S5	ja	ja
		Fremd (offene Standards)	ja	ja
	Protokoll			
	dynamische Datenlänge		ja	nein
	Multicast / Broadcast		nein	nein
	Verbindungen	zum remote Partner?	ja	ja
		dynamisch / statisch	TSEND/TRCV: dynamisch + statisch TSEND_C/TRCV_C: dynamisch	TSEND/TRCV: dynamisch TSEND_C/TRCV_C: dynamisch
	Routingfähig		ja	ja
	Anwenderschnittstelle			
	Kommunikationsbausteine		TSEND / TRCV TSEND_C / TRCV_C	TSEND / TRCV TSEND_C / TRCV_C
	maximale Anzahl Daten (*1)		<= 64 KByte	<= 64 KByte
	dynamische Adressierung Daten		ja	ja
	remote Quittierung		Transport	Transport
	Modell		Client / Client	Client / Client

Die Tabelle besteht aus zwei Bereichen.

Bereich 1

Der Bereich enthält die Bezeichnung der Kommunikationsart und der möglichen Protokolle.

Bereich 2

Der Bereich enthält die Kriterien, mit denen die Kommunikationsart beschrieben wird. Die Kriterien sind unter folgenden Überschriften zusammengefasst:

- Allgemeines
- Protokoll
- Anwenderschnittstelle

Die einzelnen Kriterien werden im folgenden Kapitel beschrieben. Eine Untermenü dieser Kriterien erscheint in den Tabellen Kommunikationsarten-Kompakt. Diese Kriterien sind mit einem Unterstrich gekennzeichnet (Beispiel: Schnittstellen).

28.2.3 Kriterien der Tabelle

Hinweis: Der Eintrag "---" im Wertebereich der Tabelle bedeutet: "Das Kriterium ist hier nicht relevant."

Tabelle 28-1

Kriterium		Bedeutung	Wertebereich
Allgemeines			
Medien		Über welche Medien können die beiden Kommunikationspartner Daten austauschen?	PN/IE, PB, MPI, Rückwandbus, Serielle Schnittstelle
<u>Schnittstellen</u>		Hier wird eingetragen, an welcher Schnittstelle die Medien angeschlossen sind. integrierte Schnittstelle: CPU externe Schnittstelle: CP, CM	CPU, CP, CM
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
Protokoll			
dynamische Datenlänge		Kann zur Laufzeit (RUN der CPU) am Kommunikationsbaustein die Datenlänge geändert werden?	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
<u>Verbindung</u>	zum remote Partner	Wird zur Kommunikation eine Verbindung zum remote Partner aufgebaut?	Ja, nein
	dynamisch / statisch	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

Kriterium	Bedeutung	Wertebereich
Anwenderschnittstelle		
<u>Kommunikationsbausteine</u>	Hier werden alle für die Datenübertragung zur Verfügung stehenden Kommunikationsbausteine (FB, SFB, FC, SFC) aufgeführt. Bausteine, die eventuell für den Verbindungsaufbau und Verbindungsabbau benötigt werden (Verbindungsbausteine), werden hier nicht erwähnt <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 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
<u>Dynamische Adressierung Daten</u>	Ist eine Änderung der Adressierung von Datenbereichen zur Laufzeit (RUN der CPU) möglich?	ja, nein
<u>remote Quittierung</u>	Hier wird angegeben, ob eine Datenübertragung vom remote Kommunikationspartner (CPU, CP, CM) quittiert wird, und was die Quittierung bedeutet. <u>nein:</u> keine Quittung von remote <ul style="list-style-type: none"> Die Daten wurden gesendet, sie haben den lokalen Partner verlassen. Der Sender erfährt <u>nicht</u>, ob die Daten im remote Anwenderdatenbereich (in der remote Applikation auf der CPU) angekommen sind. <u>Transport:</u> Transport Quittung von remote <ul style="list-style-type: none"> Die Daten wurden gesendet und vom remote Partner (CPU, CP, CM) empfangen. Der Sender erfährt <u>nicht</u>, ob die Daten im remote Anwenderdatenbereich (in der remote Applikation auf der CPU) angekommen sind. <u>Applikation:</u> Applikation Quittung von remote <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
<u>Modell</u>	Hier wird angegeben, welches Kommunikationsmodell der Datenübertragung zu Grunde liegt. Beschreibung der Kommunikationsmodelle: Siehe Kapitel 64.2.	Client / Client, Client / Server, S7 nur Server, Master / Slave, Consumer / Provider

28.3 Kapitel: 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, CM) sind die Kommunikationsbausteine geeignet?
- Wo in STEP 7 sind die Kommunikationsbausteine zu finden?

28.4 Kapitel: Anwenderschnittstellen

Die Anwenderschnittstellen der jeweiligen Kommunikationsarten werden stichpunktartig beschrieben:

- Funktionalität der Kommunikationsbausteine
- Parameter der Kommunikationsbausteine

Im vorliegenden Dokument werden zur Beschreibung der Kommunikationsbausteine einheitliche Begriffe verwendet. In diesem Zusammenhang werden die beiden Fälle unterschieden:

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

Die folgenden Bilder zeigen die in diesem Zusammenhang verwendeten Begriffe.

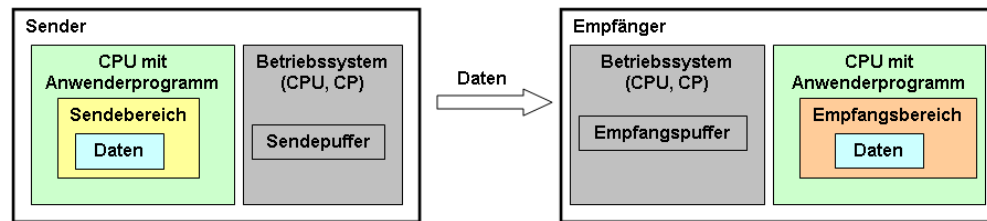
Dem remote Partner werden keine Adressinformationen der Daten mitgeteilt

Verwendete Begriffe:

- Sendebereich, Empfangsbereich
- Sendepuffer, Empfangspuffer
- Daten senden
- Daten empfangen

Beispiel: Daten senden

Abbildung 28-2

**Dem remote Partner werden Adressinformationen der Daten mitgeteilt**

Verwendete Begriffe:

- Quellbereich, Zielbereich
- Daten schreiben
- Daten lesen, holen

Daten schreiben

Abbildung 28-3

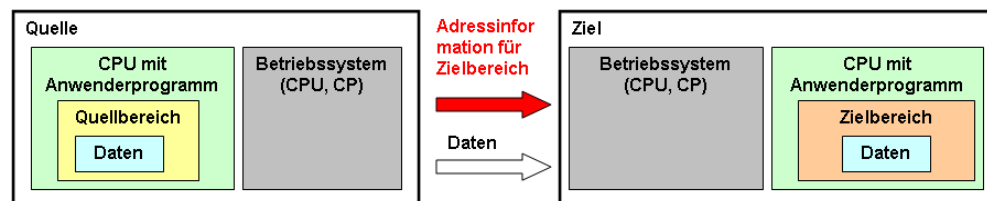
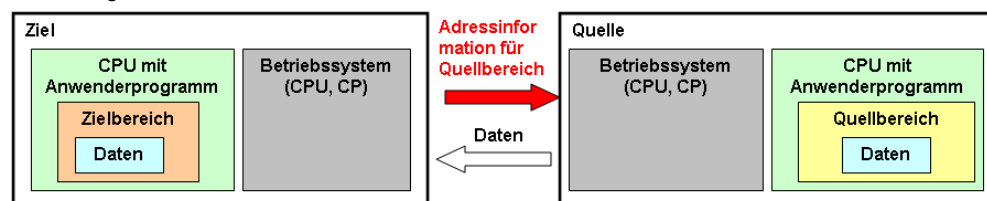
Daten lesen, holen

Abbildung 28-4



29 SIMATIC S7 spezifische Kommunikation

29.1 Merkmale

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

- Optimierte Kommunikation zwischen SIMATIC S7 Controllern
- Herstellerspezifische Kommunikation

29.2 Übersicht

Folgende Kommunikationsarten und Medien stehen zur Verfügung:

Tabelle 29-1

Kommunikationsart	Medien				
	Netz			Rückwand- bus	Serielle Schnittstelle
	PN/IE	PB	MPI		
Globaldaten-Kommunikation	---	---	x	x	---
S7-Basiskommunikation	---	x (DP)	x	---	---
S7-Kommunikation	x	x	x	x	---

In den folgenden Kapiteln werden die Kommunikationsarten beschrieben:

Tabelle 29-2

	Kapitel
Globaldaten-Kommunikation	30
S7-Basiskommunikation	31
S7-Kommunikation	32

30 Globaldaten-Kommunikation

30.1 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

30.2 Details Kommunikationsart

Tabelle 30-1 Kommunikationsarten-Detail

Kommunikationsart:		GD-Kommunikation
Protokoll:		S7-Protokoll
Allgemeines		
Medien		MPI, Rückwandbus (nur bei SIMATIC S7-400, Multicomputing)
Schnittstellen		CPU
Anbindung	SIMATIC S5	nein
	Fremd (offene Standards)	nein
Protokoll		
dynamische Datenlänge		nein
Multicast / Broadcast		Multicast
Verbindungen	zum remote Partner	nein
	dynamisch / statisch	---
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
dynamische Adressierung Daten		nein
remote Quittierung		nein
Modell		---

[Zurück zum Sprungverteiler MPI](#)

[Zurück zum Sprungverteiler Rückwandbus](#)

30.3 Überblick Anwenderschnittstellen

Zyklische Übertragung

Es sind keine Kommunikationsbausteine erforderlich.

Azyklische Übertragung

Überblick Kommunikationsbausteine:

Tabelle 30-2

Kommunikationsbau- steine	S7-400
	CPU
GD_SND	SFC 60
GD_RCV	SFC 61

Kommunikationsbausteine in STEP 7:

Tabelle 30-3

Schnittstelle		zu finden in STEP 7	
S7-400	CPU	STEP 7 (nicht TIA)	Bibliothek: Standard Library / System Function Blocks
		STEP 7 (TIA)	---

30.4 Anwenderschnittstelle GD_SND, GD_RCV

30.4.1 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

30.4.2 Parameter für GD_SND

Tabelle 30-4

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

30.4.3 Parameter für GD_RCV

Tabelle 30-5

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

31 S7-Basiskommunikation

31.1 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:
 - CPU im Zentralgerät oder Erweiterungsgerät
 - CPU dezentral

Zu (*1):

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 Kommunikation erfolgt über das Netz MPI.

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

CPU innerhalb der eigenen Station

Die Kommunikation erfolgt über das Netz PB (DP).

Die erforderlichen 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 Kommunikationsauftrag: < 94 Byte
- Das Anwenderprogramm im Sender erhält eine Information, wenn die Daten im Anwenderdatenbereich des Empfängers angekommen sind (Quittierung von der remote Applikation).

31.2 Details Kommunikationsart

Tabelle 31-1 Kommunikationsarten-Detail

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
dynamische Adressierung Daten		ja	ja	ja
remote Quittierung		Applikation	Applikation	Applikation
Modell		Client / Server	Client / Server	Client / Client
Protokoll				
dynamische Datenlänge		ja		
Multicast / Broadcast		nein		
Verbindungen	zum remote Partner	ja		
	dynamisch / statisch	dynamisch + statisch		

[Zurück zum Sprungverteiler MPI](#)

[Zurück zum Sprungverteiler Rückwandbus](#)

[Zurück zum Sprungverteiler PB](#)

31.3 Überblick Anwenderschnittstellen

Überblick Kommunikationsbausteine:

Tabelle 31-2

Kommunikationsbaustein		S7-300 CPU	S7-400 CPU
X-Bausteine (Netz MPI)	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 (Netz PB(DP))	I_PUT	SFC 73	SFC 73
	I_GET	SFC 72	SFC 72

Kommunikationsbausteine in STEP 7:

Tabelle 31-3

Schnittstelle		zu finden in STEP 7	
S7-300, S7-400	CPU	STEP 7 (nicht TIA)	Bibliothek: Standard Library / System Function Blocks
		STEP 7 (TIA)	Anweisungen: Kommunikation

31.4 Anwenderschnittstelle X_SEND / X_RCV

31.4.1 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.

31.4.2 Parameter für X_SEND

Tabelle 31-4

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
OUTPUT	Typ	Bemerkung
RET_VAL	INT	Fehlerinformation
BUSY	BOOL	Auftrag läuft / Auftrag fertig (*1)

Zu (*1): remote Quittierung: Applikation

31.4.3 Parameter für X_RCV

Tabelle 31-5

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

31.5 Anwenderschnittstelle X_PUT, X_GET

31.5.1 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.

31.5.2 Parameter für X_PUT

Tabelle 31-6

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
SD	ANY	Quellbereich
OUTPUT	Typ	Bemerkung
RET_VAL	RET_VAL	Fehlerinformation
BUSY	BUSY	Auftrag läuft / Auftrag fertig (*1)

Zu (*1): remote Quittierung: Applikation

31.5.3 Parameter für X_GET

Tabelle 31-7

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
OUTPUT	Typ	Bemerkung
RET_VAL	INT	Fehlerinformation
BUSY	BOOL	Auftrag läuft / Auftrag fertig (*1)
INOUT	Typ	Bemerkung
RD	ANY	Zielbereich (E, A, M, D)

Zu (*1): remote Quittierung: Applikation

31.6 Anwenderschnittstelle I_PUT, I_GET

31.6.1 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.

31.6.2 Parameter für I_PUT

Tabelle 31-8

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
SD	ANY	Quellbereich
OUTPUT	Typ	Bemerkung
RET_VAL	INT	Fehlerinformation
BUSY	BOOL	Auftrag läuft / Auftrag fertig (*1)

Zu (*1): remote Quittierung: Applikation

31.6.3 Parameter für I_GET

Tabelle 31-9

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
OUTPUT	Typ	Bemerkung
RET_VAL	INT	Fehlerinformation
BUSY	BOOL	Auftrag läuft / Auftrag fertig (*1)
RD	ANY	Zielbereich

Zu (*1): remote Quittierung: Applikation

32 S7-Kommunikation

32.1 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 externe Schnittstelle von CP bzw. CM..
- Kommunikation über projektierte Verbindungen (S7-Verbindung).
- Anzahl Daten pro Kommunikationsauftrag: ≤ 64 KByte
- Das Anwenderprogramm im Sender erhält eine Information, wenn die Daten im Anwenderdatenbereich des Empfängers angekommen sind (Quittierung von der remote Applikation).

32.2 Details Kommunikationsart

Tabelle 32-1 Kommunikationsarten-Detail

Kommunikationsart:		S7-Kommunikation		
Protokoll:		S7-Protokoll		
Allgemeines				
Medien		MPI, PB, PN/IE, Rückwandbus (nur bei SIMATIC S7-400, Multicomputing)		
Schnittstellen		CPU, CP, CM		
Anbindung	SIMATIC S5	nein		
	Fremd (offene Standards)	nein		
Protokoll				
dynamische Datenlänge		ja		
Multicast / Broadcast		nein		
Verbindungen	zum remote Partner?	ja		
	dynamisch / statisch	statisch		
Anwenderschnittstelle				
Kommunikationsbausteine		BSEND / BRCV	Typ "USEND / URCV"	Typ "PUT, GET"
maximale Anzahl Daten (*1)		<= 64 KByte	>= 160 Byte	>= 160 Byte
dynamische Adressierung Daten		S7-300: ja	S7-300: ja	S7-300: ja
		sonst: nein	sonst: nein	sonst: nein
remote Quittierung		Applikation	Transport	Applikation
Modell		Client / Client	Client / Client	Client / Server

[Zurück zum Sprungverteiler PN/IE](#)

[Zurück zum Sprungverteiler PB](#)

[Zurück zum Sprungverteiler MPI](#)

[Zurück zum Sprungverteiler Rückwandbus](#)

Zu (*1): Die maximale Anzahl Daten pro Kommunikationsauftrag ist abhängig von:

- Netz (MPI, PB, PN/IE)
- Schnittstelle (Kommunikation über CPU, CP oder CM)
- Kommunikationspartner (S7-300, S7-400, ...)
- Kommunikationsbaustein (BSEND, ...)

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 ermittelt werden (Online Hilfe STEP 7).

Tabelle 32-2

Kommunikationsbau- stein	S7-300		S7-400	S7-1200	S7-1500
	CPU	CP	CPU, CP	CPU, CM	CPU, CP, CM
BSEND / BRCV	= 64 KByte	≤ 32 KByte	= 64 KByte	---	= 64 kbyte
"USEND / URCV"	≥ 160 Byte	≤ 160 Byte	≥ 440 Byte	---	≥ 920 byte
"PUT, GET"	≥ 160 Byte	≤ 160 Byte	≥ 400 Byte	≥ 160 byte	≥ 880 byte

32.3 Überblick Anwenderschnittstellen

Überblick Kommunikationsbausteine

Tabelle 32-3

Typ der Anwenderschnittstelle			S7-300		S7-400, WinAC	S7-1200	S7-1500
			CPU	CP			
BSEND/BRCV	Baustein:		FB12/ FB13	FB12 / FB13	SFB12 / SFB13	---	SFB12 / SFB13
	STEP 7	TIA	BSEND /BRCV	BSEND /BRCV	BSEND /BRCV	---	BSEND /BRCV
		Nicht TIA	BSEND /BRCV	BSEND /BRCV	BSEND /BRCV	---	---
"USEND/URCV" (*1)	Baustein:		FB8 / FB9	FB8 / FB9	SFB8 / SFB9	---	SFB8 / SFB9
	STEP 7	TIA	USEND_S/ URCV_S	USEND_S/ URCV_S	USEND/ URCV	---	USEND/ URCV
		Nicht TIA	USEND /URCV	USEND /URCV	USEND /URCV	---	---
		Baustein:	FB28 / FB29	---	---	---	---
	STEP 7	TIA	USEND /URCV	---	---	---	---
		Nicht TIA	USEND_E /URCV_E	---	---	---	---
"PUT, GET" (*2)	Baustein:		FB15, FB14	FB15, FB14	SFB15, SFB14	SFB15, SFB14	SFB15, SF 14
	STEP 7	TIA	PUT_S, GET_S	PUT_S, GET_S	PUT, GET	PUT, GET	PUT, GET
		Nicht TIA	PUT, GET	PUT, GET	PUT, GET	---	---
		Baustein:	FB35, FB34	---	---	---	---
	STEP 7	TIA	PUT, GET	---	---	---	---
		Nicht TIA	PUT_E, GET_E	---	---	---	---

Zu (*1): Unter der Bezeichnung "USEND / URCV" werden alle Varianten der Bausteine (FBx, SFBx) und der Bezeichnungen in STEP 7 (USEND, USEND_E, USEND_S) zusammengefasst.

Zu (*2): Unter der Bezeichnung "PUT, GET" werden alle Varianten der Bausteine (FBx, SFBx) und der Bezeichnungen in STEP 7 (PUT, PUT_E, PUT_S) zusammengefasst.

Kommunikationsbausteine in STEP 7

Tabelle 32-4

Schnittstelle		zu finden in STEP 7	
S7-300	CPU	STEP 7 (nicht TIA)	Bibliothek: Standard Library / Communication Blocks
	CP	STEP 7 (nicht TIA)	Bibliothek: SIMATIC_NET_CP / CP300
S7-400, WinAC		STEP 7 (nicht TIA)	Bibliothek: Standard Library / System Function Blocks
S7-1200, S7-1500		STEP 7 (TIA)	Anweisung: Kommunikation

32.4 Anwenderschnittstelle: Typ "USEND / URCV"

32.4.1 Vorbemerkung

Unter der Bezeichnung "USEND / URCV" werden alle Varianten der Kommunikationsbausteine (FBx, SFBx) und der Bezeichnungen in STEP 7 (USEND, USEND_E, USEND_S) zusammengefasst.

Die Kommunikationsbausteine unterscheiden sich in der maximalen Anzahl von Sendebereichen bzw. Empfangsbereichen.

Tabelle 32-5

		S7-300		S7-400, WinAC	S7-1200, S7-1500
		CPU, CP	CPU		
maximale Anzahl Bereiche		1	4	4	4
Bezeichnung in STEP 7	TIA	USEND_S, URCV_S	USEND, URCV	USEND, URCV	USEND, URCV
	Nicht TIA	USEND, URCV	USEND_E, URCV_E	USEND, URCV	---

32.4.2 Beschreibung

Der Kommunikationsbaustein vom Typ "USEND" sendet Daten an einen Kommunikationsbaustein vom Typ "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.

Typ "USEND"

Unkoordiniertes Senden von Daten

Typ "URCV"

Unkoordiniertes Empfangen von Daten

32.4.3 Parameter für Typ "USEND"

Tabelle 32-6

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
REQ	BOOL	BOOL	Anstoß Sendeauftrag
ID	WORD	CONN_PRG	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
R_ID	DWORD	CONN_R_ID	Zuordnung Sende SFB/FB und Empfangs SFB/FB. Dies ermöglicht die Kommunikation mehrerer SFB/FB Paare über dieselbe logische Verbindung.
OUTPUT			Bemerkung
DONE	BOOL	BOOL	Auftrag fertig (*1)
ERROR	BOOL	BOOL	Fehlerinformation
STATUS	WORD	WORD	
IN_OUT			Bemerkung
SD_i	ANY	VARIANT	Sendebereich (i=1,2,3,4)

Zu (*1): remote Quittierung: Transport

32.4.4 Parameter für Typ "URCV"

Tabelle 32-7

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
EN_R	BOOL	BOOL	Anstoß Empfangsauftrag
ID	WORD	CONN_PRG	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
R_ID	DWORD	CONN_R_ID	Zuordnung Sende SFB/FB und Empfangs SFB/FB. Dies ermöglicht die Kommunikation mehrerer SFB/FB Paare über dieselbe logische Verbindung.
OUTPUT			Bemerkung
NDR	BOOL	BOOL	Daten im Eingangspuffer
ERROR	BOOL	BOOL	Fehlerinformation
STATUS	WORD	WORD	
IN_OUT			Bemerkung
RD_i	ANY	VARIANT	Empfangsbereich (i=1,2,3,4)

32.5 Anwenderschnittstelle: BSEND / BRCV

32.5.1 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 müssen zur Datenübertragung paarweise verwendet werden.

Die Datenübertragung wird wie folgt 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 werden 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

32.5.2 Parameter für BSEND

Tabelle 32-8

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
REQ	BOOL	BOOL	Anstoß Sendeauftrag
R	BOOL	BOOL	Abbruch Sendeauftrag
ID	WORD	CONN_PRG	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
R_ID	DWORD	CONN_R_ID	Zuordnung Sende SFB/FB und Empfangs SFB/FB. Dies ermöglicht die Kommunikation mehrerer SFB/FB Paare über dieselbe logische Verbindung.
OUTPUT			Bemerkung
DONE	BOOL	BOOL	Auftrag fertig (*1)
ERROR	BOOL	BOOL	Fehlerinformation
STATUS	WORD	WORD	
IN_OUT			Bemerkung
SD_1	ANY	VARIANT	Sendebereich
LEN	WORD	WORD	Länge der zu sendenden Daten

Zu (*1): remote Quittierung: Applikation

32.5.3 Parameter für BRCV

Tabelle 32-9

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
EN_R	BOOL	BOOL	Anstoß Empfangsauftrag
ID	WORD	BOOL	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
R_ID	DWORD	CONN_PRG	Zuordnung Sende SFB/FB und Empfangs SFB/FB. Dies ermöglicht die Kommunikation mehrerer SFB/FB Paare über dieselbe logische Verbindung.
OUTPUT		CONN_R_ID	Bemerkung
NDR	BOOL		Daten im Empfangsbereich
ERROR	BOOL	BOOL	Fehlerinformation
STATUS	WORD	BOOL	
IN_OUT		WORD	Bemerkung
RD_1	ANY	VARIANT	Empfangsbereich
LEN	WORD	WORD	Länge der empfangenen Daten

32.6 Anwenderschnittstelle: Typ "PUT, GET"

32.6.1 Vorbemerkung

Unter der Bezeichnung "PUT, GET" werden alle Varianten der Kommunikationsbausteine (FBx, SFBx) und der Bezeichnungen in STEP 7 (PUT, PUT_E, PUT_S) zusammengefasst.

Die Kommunikationsbausteine unterscheiden sich in der maximalen Anzahl von Sendebereichen bzw. Empfangsbereichen.

Tabelle 32-10

		S7-300		S7-400, WinAC	S7-1200, S7-1500
		CPU, CP	CPU		
maximale Anzahl Bereiche		1	4	4	4
Bezeichnung in STEP 7	TIA	PUT_S, GET_S	PUT, GET	PUT, GET	PUT, GET
	Nicht TIA	PUT, GET	PUT_E, GET_E	PUT, GET	---

32.6.2 Beschreibung

Mit den Kommunikationsbausteinen kommuniziert eine CPU (Client) mit einer anderen CPU (Server).

In der anderen CPU (Server) ist im Anwenderprogramm kein Kommunikationsbaustein erforderlich (Client / Server Kommunikation).

Typ "PUT"

Daten in CPU schreiben

Typ "GET"

Daten aus CPU lesen

32.6.3 Parameter für Typ "PUT"

Tabelle 32-11

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
REQ	BOOL	BOOL	Anstoß Schreibauftrag
ID	WORD	CONN_PRG	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
OUTPUT			Bemerkung
DONE	BOOL	BOOL	Auftrag fertig (*1)
ERROR	BOOL	BOOL	Fehlerinformation
STATUS	WORD	WORD	
IN_OUT			Bemerkung
ADDR_i	ANY	REMOTE	Zielbereich (i=1,2,3,4)
SD_i	ANY	VARIANT	Quellbereich (i=1,2,3,4)

Zu (*1): remote Quittierung: Applikation

32.6.4 Parameter für Typ "GET"

Tabelle 32-12

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
REQ	BOOL	BOOL	Anstoß Leseauftrag
ID	WORD	CONN_PRG	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
OUTPUT			Bemerkung
NDR	BOOL	BOOL	Auftrag fertig (*1)
ERROR	BOOL	BOOL	Fehlerinformation
STATUS	WORD	WORD	
IN_OUT			Bemerkung
ADDR_i	ANY	REMOTE	Quellbereich (i=1,2,3,4)
RD_i	ANY	VARIANT	Zielbereich (i=1,2,3,4)

Zu (*1): remote Quittierung: Applikation

33 PROFINET/Industrial Ethernet (PN/IE)

33.1 Merkmale

Die Kommunikation über PN/IE zeichnet sich durch folgende Merkmale aus:

- Kommunikation zwischen SIMATIC Controllern
- Kommunikation mit fremden Controllern

33.2 Übersicht

Über PN/IE sind folgende Kommunikationsarten möglich:

Tabelle 33-1

	Kommunikationsart		Kapitel
SIMATIC S7 spezifische Kommunikation	S7-Kommunikation		32
Offener Standard	Offene-Kommunikation mit Send/Receive-Bausteinen		34
	Offene-Kommunikation mit T-Bausteinen		35
	PN-Kommunikation	CBA	36
		PNIO	37

Die Kommunikationsarten zum „Offenen Standard“ werden im Folgenden beschrieben.

34 PN/IE: Offene-Kommunikation mit Send/Receive-Bausteinen

34.1 Merkmale

Die Offene-Kommunikation mit Send/Receive-Bausteinen zeichnet sich durch folgende Merkmale aus:

- Offener Standard (Kommunikation mit fremden Controllern möglich)
- Kommunikation über CP
- Kommunikation über projektierte Verbindungen
- Kommunikation über Protokolle: TCP, ISO on TCP, UDP, ISO
- Anzahl Daten pro Kommunikationsauftrag: ≤ 8 KByte
- Quittierung vom remote Transportsystem (nicht beim Protokoll UDP).
- Keine Quittierung von der remote Applikation

Projektierung von Verbindungen

Verbindungen werden in STEP 7 projektiert. Aufbau und Abbau der Verbindungen erfolgt durch das Betriebssystem. Da die Verbindungen projektiert werden, sind die Verbindungsparameter zur Laufzeit nicht änderbar.

Verbindungstypen

Tabelle 34-1

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 muss eine „UDP-Verbindung“ projektiert werden. Eine „UDP-Verbindung“ ist jedoch keine Verbindung im Sinne des Dokumentes (Kapitel 6). Bei UDP wird keine Verbindung zum remote Kommunikationspartner aufgebaut.

Merkmale Protokolle

Tabelle 34-2

Protokoll	Verbindungsaufbau zum remote Partner	Übertragung von Länge und Ende der Daten
TCP	ja	nein (*1)
ISO on TCP	ja	ja
UDP	nein	ja

Zu (*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.

34.2 Details Kommunikationsart

Tabelle 34-3 Kommunikationsarten-Detail

Kommunikationsart:		Offene-Kommunikation mit Send/Receive-Bausteinen (Netz PN/IE)						
Protokoll:		ISO		ISO on TCP		TCP		UDP
Allgemeines								
Schnittstellen		CP		CP		CP		CP
Anbindung	SIMATIC S5	ja		ja		ja		ja
	Fremd (offene Standards)	nein		ja		ja		ja
Protokoll								
dynamische Datenlänge		ja		ja		nein		ja
Multicast / Broadcast		nein		nein		nein		ja / ja
Verbindun- gen	zum remote Partner?	ja		ja		ja		nein
	dynamisch / statisch	statisch		statisch		statisch		statisch
Anwenderschnittstelle								
Kommunikationsbausteine (*1)		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 (*1)		<= 8 KByte		<= 8 KByte		<= 8 KByte		<= 2 KByte
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

[Zurück zum Sprungverteiler PN/IE](#)

Erläuterungen zur Tabelle:

Zu (*1): Die maximale Anzahl Daten ist abhängig von:

- Kommunikationspartner (S7-300, S7-400, ...)
- Kommunikationsbaustein (AG_SEND, AG_LSEND, AG_SSEND)
- Protokoll (TCP, ...)

Übersicht:

Tabelle 34-4

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

34.3 Überblick Anwenderschnittstellen

Send/Receive-Bausteine

Überblick Kommunikationsbausteine:

Tabelle 34-5

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 34-6

Schnittstelle		zu finden in STEP 7 (nicht TIA)
S7-300	CP	Bibliothek: SIMATIC_NET_CP / CP300
S7-400	CP	Bibliothek: 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.

34.4 Anwenderschnittstelle AG_xSEND, AG_xRECV

Für die 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

34.4.1 Beschreibung

Der Kommunikationsbaustein AG_xRECV sendet Daten an den Kommunikationsbaustein AG_xSEND. 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.

34.4.2 Parameter für AG_SEND, AG_LSEND, AG_SSEND

Tabelle 34-7

INPUT	Typ	Bemerkung
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
LEN	INT	Länge der zu sendenden Daten
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): Fallunterscheidung:

Unterschiedliche Bedeutung, je nach verwendetem Protokoll:

- Daten liegen auf dem remote CP: ISO, ISO on TCP, TCP
- Daten wurden vom lokalen CP gesendet: UDP

34.4.3 Parameter für AG_RECV, AG_LRECV, AG_SSRECV

Tabelle 34-8

INPUT	Typ	Bemerkung
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
OUTPUT	Typ	Bemerkung
NDR	BOOL	Daten im Empfangsbereich
LEN	INT	Länge der empfangenen Daten
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): 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.

34.5 Anwenderschnittstelle FETCH, WRITE (Server)

In der S7-CPU des Servers sind für den Datenaustausch keine Kommunikationsbausteine nötig.

FETCH

Der Verbindungspartner (fremder Controller, SIMATIC S5) kann lesend auf Daten in der S7-CPU (Server) zugreifen (Daten holen).

WRITE

Der Verbindungspartner (fremder Controller, SIMATIC S5) kann schreibend auf Daten in der SIMATIC S7 zugreifen (Daten schreiben).

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.

35 PN/IE: Offene-Kommunikation mit „T-Bausteinen“

35.1 Merkmale

Die Offene-Kommunikation mit T-Bausteinen zeichnet sich durch folgende Merkmale aus:

- Offener Standard (Kommunikation mit fremden Controllern möglich)
- Kommunikation über
 - projektierte oder programmierte Verbindungen: TCP, IoT, UDP
 - projektierte Verbindungen: ISO
- Kommunikation über Protokolle: TCP, ISO on TCP, UDP, ISO
- Anzahl Daten pro Kommunikationsauftrag: ≤ 64 KByte
- Quittierung vom remote Transportsystem (nicht beim Protokoll UDP).
- Keine Quittierung von der remote Applikation.

Programmierung von Verbindungen

Aufbau und Abbau der Verbindungen werden im STEP 7 Anwenderprogramm programmiert. Dazu gibt es zwei Möglichkeiten:

- Aufruf von Verbindungsbausteinen (TCON, ...)
- Aufruf von Kommunikationsbausteinen, in denen der Aufbau oder Abbau von Verbindungen integriert ist (TSEND_C, ...)

Durch die Programmierung der Verbindungen sind die Verbindungsparameter zur Laufzeit (im RUN der CPU) änderbar.

Verbindungstypen

Tabelle 35-1

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 parametrisiert. Bei UDP wird keine Verbindung zum remote Kommunikationspartner aufgebaut.

Merkmale Protokolle

Tabelle 35-2

Protokoll	Verbindungsaufbau zum remote Partner	Übertragung von Länge und Ende der Daten
TCP	ja	nein (*1)
ISO on TCP	ja	ja
UDP	nein	ja

Zu (*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.

35.2 Details Kommunikationsart

Tabelle 35-3 Kommunikationsarten-Detail

Kommunikationsart:		Offene-Kommunikation mit T-Bausteinen (Netz PN/IE)		
Protokoll:		ISO on TCP	TCP	UDP
Allgemeines				
Schnittstellen		CPU, CP, CM	CPU, CP, CM	CPU, CP, CM
Anbindung	SIMATIC S5	ja	ja	nein
	Fremd (offene Standards)	ja	ja	ja
Protokoll				
dynamische Datenlänge		ja	nein	ja
Multicast / Broadcast		nein	nein	nein
Verbin- dungen	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
Anwenderschnittstelle				
Kommunikationsbausteine		TSEND / TRCV TSEND_C / TRCV_C	TSEND / TRCV TSEND_C / TRCV_C	TUSEND / TURCV
maximale Anzahl Daten (*1)		<= 64 KByte	<= 64 KByte	= 1472 Byte
dynamische Adressierung Daten		ja	ja	ja
remote Quittierung		Transport	Transport	nein
Modell		Client / Client	Client / Client	Client / Client

[Zurück zum Sprungverteiler PN/IE](#)

Erläuterungen zur Tabelle:

Zu (*1): Die maximale Anzahl Daten ist abhängig von:

- Kommunikationspartner (S7-300, S7-400, ...) und verwendete Schnittstelle (CPU, CP, CM)
- Protokoll (TCP, ...) und Verbindungstyp

Übersicht:

Tabelle 35-4

Protokoll	Verbin- dungstyp	S7-1200	S7-1500		S7-300	S7-400		WinAC RTX 2010
		CPU, CM	CPU, CM	CP	CPU	CPU	CP	CPU / CP (Submodul)
TCP	B#16#01	---	---	---	1460 Byte	---	---	---
	B#16#11	8 KByte	64 KByte	64 KByte	32 KByte	32 KByte	---	64 KByte
ISO on TCP	B#16#12	8 KByte	64 KByte	64 kByte	32 KByte	32 KByte	1452 Byte	64 KByte
UDP	B#16#13	---	1472 Byte	1472 Byte	1472 Byte	1472 Byte	---	1472 Byte

35.3 Überblick Anwenderschnittstellen

T-Bausteine

Überblick Kommunikationsbausteine

Tabelle 35-5

Kommunikations- baustein	S7-300	S7-400	S7-1200	S7-1500
	CPU	CPU, CP		
TSEND / TRCV	FB 63 / FB 64	FB 63 / FB 64	SFB 100 / SFB 101	SFB 150 / SFB 151
TUSEND / TURCV	FB 67 / FB 68	FB 67 / FB 68	SFB 100 / SFB 101	SFB 150 / SFB 151
TSEND_C / TRCV_C	---	---	FB 1030 / FB 1031	FB 1030 / FB 1031

Überblick Verbindungsbausteine:

Tabelle 35-6

Verbindungsbaustein	S7-300	S7-400	S7-1200	S7-1500
	CPU	CPU, CP		
TCON	FB 65	FB 65	SFB 102	SFB 109
TDISCON	FB 66	FB 66	SFB 103	SFB 103

Kommunikationsbausteine in STEP 7:

Tabelle 35-7

Schnittstellen	zu finden in STEP 7	
S7-300, S7-400	STEP 7 (nicht TIA)	Bibliothek: Standard Library / Communication Blocks
S7-1200, S7-1500	STEP 7 (TIA)	Anweisung: Kommunikation

Server für Fetch/Write

Überblick: Kommunikationsbausteine

Tabelle 35-8

Kommunikations- baustein	S7-300	S7-400
	CPU	CPU
FW_TCP	FB 210	FB 210
FW_IOT	FB 220	FB 220

Kommunikationsbausteine in STEP 7:

Tabelle 35-9

Schnittstellen	zu finden in STEP 7	
S7-300, S7-400	CPU	STEP 7 (nicht TIA) Bibliothek: 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)

35.4 Anwenderschnittstelle TSEND / TRCV

35.4.1 Beschreibung

Der Kommunikationsbaustein **TSEND** sendet Daten an den Kommunikationsbaustein **TRCV**. Die Datenübertragung erfolgt über projektierte Verbindungen oder über programmierte Verbindungen (Kapitel 6).

35.4.2 Parameter für TSEND

Tabelle 35-10

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
REQ	BOOL	BOOL	Anstoß Sendeauftrag
ID	WORD	CONN_OUC	Referenz auf die zugehörige Verbindung
LEN	INT	UINT	Länge der zu sendenden Daten
OUTPUT			Bemerkung
DONE	BOOL	BOOL	Auftrag fertig (*1)
BUSY	BOOL	BOOL	Auftrag läuft
ERROR	BOOL	BOOL	Fehlerinformation
STATUS	WORD	WORD	
IN_OUT			Bemerkung
DATA	ANY	VARIANT	Sendebereich

Zu (*1): remote Quittierung: Transport

35.4.3 Parameter für TRCV

Tabelle 35-11

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
EN_R	BOOL	BOOL	Anstoß: Empfangsauftrag
ID	WORD	CONN_OUC	Referenz auf die zugehörige Verbindung
LEN	INT	UINT	Länge des Empfangsbereiches
OUTPUT			Bemerkung
NDR	BOOL	BOOL	Daten im Empfangsbereich
BUSY	BOOL	BOOL	Auftrag läuft
ERROR	BOOL	BOOL	Fehleranzeige
STATUS	WORD	WORD	
RCVD_LEN	INT	UINT	Anzahl der tatsächlich empfangenen Daten
IN_OUT			Bemerkung
DATA	ANY	VARIANT	Empfangsbereich

35.5 Anwenderschnittstelle TUSEND / TURCV

35.5.1 Beschreibung

Der Kommunikationsbaustein **TUSEND** sendet Daten an den Kommunikationsbaustein **TURCV**. Die Datenübertragung erfolgt über programmierte Verbindungen (Aufruf von Verbindungsbausteinen). Hier ist nur das Protokoll UDP (unquittierter Datagram-Dienst) möglich.

35.5.2 Parameter für TUSEND

Tabelle 35-12

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
REQ	BOOL	BOOL	Anstoß Sendeauftrag
ID	WORD	WORD	Referenz auf zugehörige Verbindung
LEN	INT	UINT	Länge der zu sendenden Daten
OUTPUT			Bemerkung
DONE	BOOL	BOOL	Auftrag fertig (*1)
BUSY	BOOL	BOOL	Auftrag läuft
ERROR	BOOL	BOOL	Fehlerinformation
STATUS	WORD	WORD	
IN_OUT			Bemerkung
DATA	ANY	VARIANT	Sendebereich
ADDR	ANY	TADDR_Param	Adresse des Empfänger

Zu (*1): remote Quittierung: keine

35.5.3 Parameter für TURCV

Tabelle 35-13

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
EN_R	BOOL	BOOL	Anstoß Empfangsauftrag
ID	WORD	WORD	Referenz auf zugehörige Verbindung
LEN	INT	UINT	Länge des Empfangsbereiches
OUTPUT			Bemerkung
NDR	BOOL	BOOL	Daten im Empfangsbereich
BUSY	BOOL	BOOL	Auftrag läuft
ERROR	BOOL	BOOL	Fehlerinformation
STATUS	WORD	WORD	
RCVD_LEN	INT	UINT	Anzahl der tatsächlich empfangenen Daten
IN_OUT			Bemerkung
DATA	ANY	VARIANT	Empfangsbereich
ADDR	ANY	TADDR_Param	Adresse des Senders

35.6 Anwenderschnittstelle Verbindungsbausteine

35.6.1 Beschreibung

Mit dem Verbindungsbaustein **TCON** wird eine Verbindung aufgebaut, mit dem Verbindungsbaustein **TDISCON** wird eine Verbindung abgebaut.

35.6.2 Parameter für TCON

Tabelle 35-14

INPUT	Typ			Bemerkung
	S7-300,S 7-400	S7-1200	S7-1500	
REQ	BOOL	BOOL	BOOL	Anstoß Aufbau Verbindung
ID	WORD	CONN_UOC	CONN_UOC	Referenz auf die Verbindung
OUTPUT				Bemerkung
DONE	BOOL	BOOL	BOOL	Auftrag fertig
BUSY	BOOL	BOOL	BOOL	Auftrag läuft
ERROR	BOOL	BOOL	BOOL	Fehlerinformation
STATUS	WORD	WORD	WORD	
IN_OUT				Bemerkung
CONNECT	ANY	TCON_Param	VARIANT	Verbindungsbeschreibung

35.6.3 Parameter für TDISCON

Tabelle 35-15

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
REQ	BOOL	BOOL	Anstoß Abbau Verbindung
ID	WORD	CONN_OUC	Referenz auf die Verbindung
OUTPUT			Bemerkung
DONE	BOOL	BOOL	Auftrag fertig
BUSY	BOOL	BOOL	Auftrag läuft
ERROR	BOOL	BOOL	Fehlerinformation
STATUS	WORD	WORD	

35.7 Anwenderschnittstelle TSEND_C / TRCV_C

Zur Unterscheidung von den anderen T-Bausteinen, werden diese Bausteine im Folgenden mit T-Compact-Bausteine bezeichnet.

35.7.1 Beschreibung

Der Kommunikationsbaustein TSEND_C sendet Daten an den Kommunikationsbaustein TRCV_C. Die Datenübertragung erfolgt über projektierte Verbindungen oder über programmierte Verbindungen (Kapitel 6).

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.

35.7.2 Parameter für TSEND_C

Tabelle 35-16

INPUT	Typ		Bemerkung
	S7-1200	S7-1500	
REQ	BOOL	BOOL	Anstoß Sendeauftrag
CONT	BOOL	BOOL	Verbindung aufbauen / abbauen
LEN	UINT	UINT	Länge der zu sendenden Bytes
OUTPUT			Bemerkung
DONE	BOOL	BOOL	Auftrag fertig (*1)
BUSY	BOOL	BOOL	Auftrag läuft
ERROR	BOOL	BOOL	Fehlerinformation
STATUS	WORD	WORD	
IN_OUT			Bemerkung
DATA	VARIANT	VARIANT	Sendebereich
ADDR	---	VARIANT	Optionaler Parameter (nur S7-1500): Zeigt beim Protokoll UDP auf die Adresse des Empfängers.
CONNECT	TCON_Param	VARIANT	Verbindungsbeschreibung
COM_RST	BOOL	BOOL	Neustart des Bausteines

Zu (*1): remote Quittierung: Transport

35.7.3 Parameter für TRCV_C

Tabelle 35-17

INPUT	Typ		Bemerkung
	S7-1200	S7-1500	
EN_R	BOOL	BOOL	Anstoß: Empfangauftrag
CONT	BOOL	BOOL	Verbindung aufbauen / abbauen
LEN	UINT	UINT	Länge des Empfangsbereiches
OUTPUT			Bemerkung
DONE	BOOL	BOOL	Daten im Empfangsbereich
BUSY	BOOL	BOOL	Auftrag läuft
RCVD_LEN	UINT	UINT	Anzahl empfangener Daten
ERROR	BOOL	BOOL	Fehlerinformation
STATUS	WORD	WORD	
RCVD_LEN	INT	INT	Anzahl der tatsächlich empfangenen Daten
IN_OUT			Bemerkung
DATA	VARIANT	VARIANT	Empfangsbereich
ADDR	---	VARIANT	Optionaler Parameter (nur S7-1500): Zeigt beim Protokoll UDP auf die Adresse des Senders.
CONNECT	TCON_Param	VARIANT	Verbindungsbeschreibung
COM_RST	BOOL	BOOL	Neustart des Bausteines

35.8 Anwenderschnittstelle FETCH, WRITE (Server)

35.8.1 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

35.8.2 Parameter für FW_TCP

Tabelle 35-18

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

35.8.3 Parameter für FW_IOT

Tabelle 35-19

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

36 PN/IE: CBA

Im Dokument werden folgende Abkürzungen verwendet:

- CBA für PROFINET CBA
- PNIO für PROFINET IO

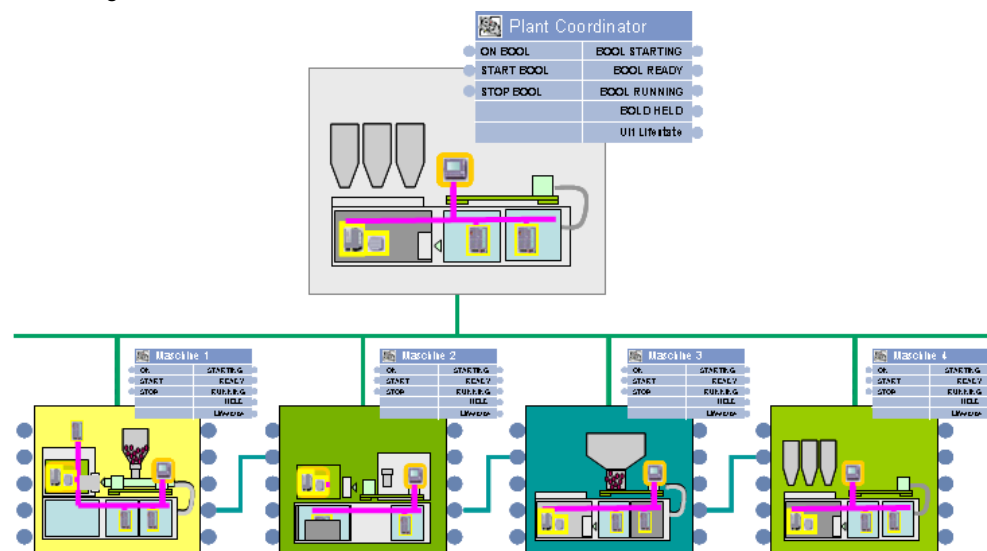
36.1 Merkmale

CBA

CBA (Component Based Automation) ist ein Automatisierungskonzept 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 36-1

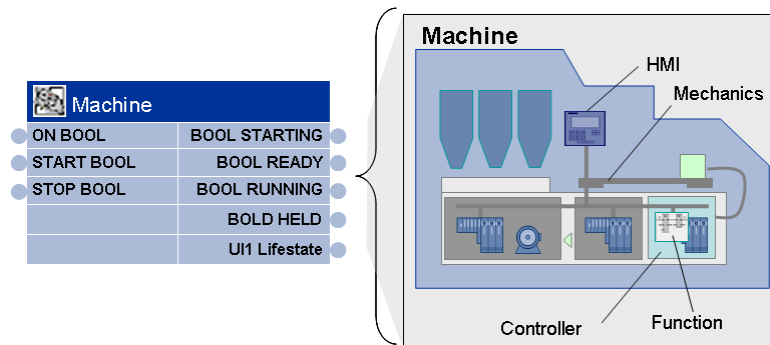


PROFINET-Komponente

Eine PROFINET-Komponente ist eine wiederverwendbare Funktionseinheit:

- Kapselung der Automatisierungsfunktionalität in einem Softwareprogramm
- Eindeutige Komponenten-Interfaces für den Datenaustausch mit anderen Komponenten.

Abbildung 36-2



Merkmale

- Grafische Konfiguration der Kommunikation (projektieren statt programmieren)
- Die Leistungsfähigkeit der Kommunikation ist offline kalkulierbar.
- Herstellerübergreifende Kommunikation

CBA und PNIO

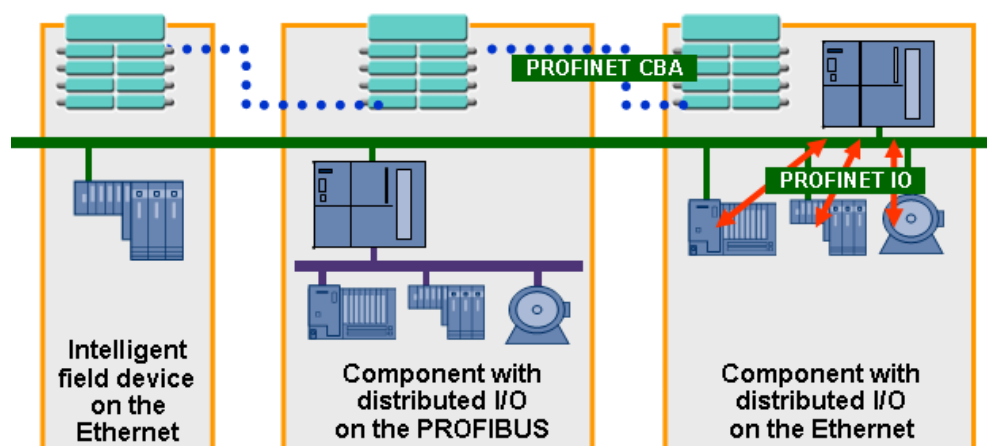
CBA (PROFINET CBA):

- Zyklischer und azyklischer Datenaustausch zwischen Controllern

PNIO (PROFINET IO):

- Zyklischer Datenaustausch zwischen einem PN Controller und den zugehörigen PN Devices.

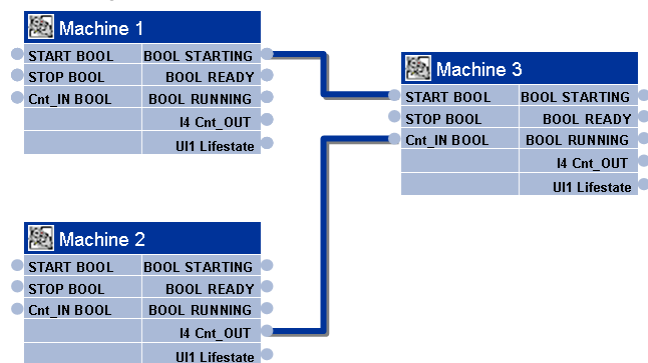
Abbildung 36-3



36.2 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 36-4



37 PN/IE: PNIO

Im Dokument werden folgende Abkürzungen verwendet:

- CBA für PROFINET CBA
- PNIO für PROFINET IO

37.1 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 in Kapitel 5.4.

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
 - IO Device sendet Daten an IO Controller
- Der Datenaustausch erfolgt konsistent über die gesamte Länge (systembedingte Datenkonsistenz)

37.2 Details Kommunikationsart

Tabelle 37-1 Kommunikationsarten-Detail

Kommunikationsart:		PNIO (Netz PN/IE)
Protokoll:		PN
Allgemeines		
Schnittstellen		CPU, CP, CM
Anbindung	SIMATIC S5	ja
	Fremd (offene Standards)	ja
Protokoll		
dynamische Datenlänge		nein
Multicast / Broadcast		nein
Verbindungen	zum remote Partner?	nein
	dynamisch / statisch	---
Anwenderschnittstelle		
Kommunikationsbausteine		Ladebefehle / Transferbefehle DPRD_DAT, DPWR_DAT PNIO_SEND, PNIO_RECV (nur S7-300 CP)
maximale Anzahl Daten		Siehe technische Daten der CPU
dynamische Adressierung Daten		nein
remote Quittierung		Applikation
Modell		Consumer / Provider

[Zurück zum Sprungverteiler PN/IE](#)

37.3 Überblick Anwenderschnittstellen

Überblick:

Tabelle 37-2

Anwender- schnittstelle	S7-300		S7-400	S7-1200	S7-1500
	CPU	CP	CPU, CP		
PNIO_SEND	---	FC 11 (*3)	---	---	---
PNIO_RECV	---	FC 12 (*3)	---	---	---
DPRD_DAT	SFC 14 (*2)	---	SFC 14 (*2)	(*1)	(*1)
DPWR_DAT	SFC 15 (*2)	---	SFC 15 (*2)	(*1)	(*1)

Die Anwenderschnittstelle ist in STEP 7 zu finden:

(*1): STEP 7 (TIA): Erweiterte Anweisung

(*2): STEP 7 (nicht TIA): Bibliothek: Standard Library / System Function Blocks

(*3): STEP 7 (nicht TIA): Bibliothek: SIMATIC_NET_CP / CP300

37.4 Anwenderschnittstelle PNIO_SEND, PNIO_RECV

37.4.1 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.

37.4.2 Parameter für PNIO_SEND

Tabelle 37-3

INPUT	Typ	Bemerkung
CPLADDR	WORD	projektierte Anfangsadresse des CP
MODE	BYTE	CP ist IO Controller oder IO Device
LEN	INT	Länge der zu sendenden Daten
OUTPUT	Typ	Bemerkung
CHECK_IOCS	BOOL	IOCS Statusbereich
IOCS	ANY	
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
DONE	BOOL	Anzeige, ob Auftrag fehlerfrei ausgeführt wurde (*1)
IN_OUT	Typ	Bemerkung
SEND	ANY	Sendebereich

Zu (*1): remote Quittierung: Applikation

37.4.3 Parameter für PNIO_RECV

Tabelle 37-4

INPUT	Typ	Bemerkung
CPLADDR	WORD	projektierte Anfangsadresse des CP
MODE	BYTE	CP ist IO Controller oder IO Device
LEN	INT	Länge der zu empfangenen Daten
OUTPUT	Typ	Bemerkung
CHECK_IOPS	BOOL	IOCS Statusbereich
IOPS	ANY	
NDR	BOOL	Anzeige, ob Auftrag fehlerfrei ausgeführt wurde (*1)
ERROR	BOOL	
STATUS	WORD	
ADD_INFO	WORD	
IN_OUT	Typ	Bemerkung
RECV	ANY	Empfangsbereich

Zu (*1): remote Quittierung: Applikation

37.5 Anwenderschnittstelle DPRD_DAT, DPWR_DAT

37.5.1 Beschreibung

Mit den Anwenderschnittstellen hat ein IO Controller Zugriff auf Daten eines IO Devices.

DPRD_DAT

Konsistente Daten eines IO Devices lesen

DPWR_DAT

Konsistente Daten in IO Device schreiben

37.5.2 Parameter für DPRD_DAT

Tabelle 37-5

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
LADDR	WORD	HW_IO	Projektierte Anfangsadresse im Eingangsbereich, aus dem gelesen werden soll
OUTPUT			Bemerkung
RET_VAL	INT	DINT, INT, LREAL, REAL	Anzeige, ob Auftrag fehlerfrei ausgeführt wurde (*1)
RECORD	ANY	VARIANT	Zielbereich

Zu (*1): remote Quittierung: Applikation

37.5.3 Parameter für DPWR_DAT

Tabelle 37-6

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
LADDR	WORD	HW_IO	Projektierte Anfangsadresse im Ausgangsbereich, auf den geschrieben werden soll
RECORD	ANY	VARIANT	Quellbereich
OUTPUT			Bemerkung
RET_VAL	INT	DINT, INT, LREAL, REAL	Anzeige, ob Auftrag fehlerfrei ausgeführt wurde (*1)

Zu (*1): remote Quittierung: Applikation

38 PROFIBUS (PB)

38.1 Merkmale

Die Kommunikation über PROFIBUS zeichnet sich durch folgende Merkmale aus:

- Kommunikation zwischen SIMATIC Controllern
- Kommunikation mit fremden Controllern

38.2 Übersicht

Über PB sind folgende Kommunikationsarten möglich:

Tabelle 38-1

	Kommunikationsart	Kapitel
SIMATIC S7 spezifische Kommunikation	S7-Basiskommunikation	31
	S7-Kommunikation	32
Offener Standard	Offene-Kommunikation mit Send/Receive-Bausteinen	39
	FMS-Kommunikation	40
	DP-Kommunikation	41

Die Kommunikationsarten zum „Offenen Standard“ werden im Folgenden beschrieben.

39 PB: Offene-Kommunikation mit Send/Receive-Bausteinen

39.1 Merkmale

Die Offene-Kommunikation mit Send/Receive-Bausteinen zeichnet sich durch folgende Merkmale aus:

- Offener Standard (Kommunikation mit fremden Controllern möglich)
- Kommunikation über CP
- Kommunikation über Protokoll: FDL
- Kommunikation über projektierte Verbindungen
- 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.

39.2 Details Kommunikationsart

Tabelle 39-1 Kommunikationsarten-Detail

Kommunikationsart:		Offene-Kommunikation mit Send/Receive-Bausteinen (Netz PB)
Protokoll:		FDL
Allgemeines		
Schnittstellen		CP
Anbindung	SIMATIC S5	ja
	Fremd (offene Standards)	ja
Protokoll		
dynamische Datenlänge		ja
Multicast / Broadcast		ja / ja
Verbindungen	zum remote Partner?	ja
	dynamisch / statisch	statisch
Anwenderschnittstelle		
Kommunikationsbausteine		AG_SEND / AG_REC, AG_LSEND / AG_LREC
maximale Anzahl Daten		= 240 Byte (Der eventuell enthaltene Auftragsheader (4 Byte) ist darin enthalten.)
dynamische Adressierung Daten		ja
remote Quittierung		Transport
Modell		Client / Client

[Zurück zum Sprungverteiler PB](#)

39.3 Überblick Anwenderschnittstellen

Überblick Kommunikationsbausteine:

Tabelle 39-2

Kommunikationsbaustein	S7-300	S7-400
	CP	CP
AG_SEND / AG_RECV	FC 5 / FC 6	FC 5 / FC 6
AG_LSEND / AG_LREC (*1)	-----	FC 50 / FC 60

Zu (*1): 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 39-3

Schnittstelle	zu finden in STEP 7	
S7-300, CP	STEP 7 (nicht TIA)	Bibliothek: SIMATIC_NET_CP / CP300
S7-400, CP	STEP 7 (nicht TIA)	Bibliothek: SIMATIC_NET_CP / CP400

39.4 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_LREC

Bedeutung von x = L

Die Kommunikationsbausteine sind optimiert für die Übertragung umfangreicher Daten (L steht für „long“).

39.4.1 Beschreibung

Der Kommunikationsbaustein AG_xRECV sendet Daten an den Kommunikationsbaustein AG_xSEND. 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.

39.4.2 Parameter für AG_SEND, AG_LSEND

Tabelle 39-4

INPUT	Typ	Bemerkung
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	Sendegebiet
LEN	INT	Länge der zu sendenden Daten
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): 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

39.4.3 Parameter für AG_RECV, AG_LRECV

Tabelle 39-5

INPUT	Typ	Bemerkung
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
OUTPUT	Typ	Bemerkung
NDR	BOOL	Daten im Empfangsbereich
LEN	INT	Länge der empfangenen Daten
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): Die Daten wurden aus dem Empfangspuffer (CP) in den Empfangsbereich (CPU) kopiert.

40 PB: FMS-Kommunikation

40.1 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

40.2 Details Kommunikationsart

Tabelle 40-1 Kommunikationsarten-Detail

Kommunikationsart:		FMS-Kommunikation (Netz PB)	
Protokoll:		FMS	
Allgemeines			
Schnittstellen		CP	
Anbindung	SIMATIC S5	ja	
	Fremd (offene Standards)	ja	
Protokoll			
dynamische Datenlänge		ja	
Multicast / Broadcast		nein / ja	
Verbindungen	zum remote Partner?	ja	
	dynamisch / statisch	statisch	
Anwenderschnittstelle			
Kommunikationsbausteine		READ, WRITE	REPORT
maximale Anzahl Daten (*1)		READ <= 237 Byte WRITE <= 233 Byte	<= 233 Byte
dynamische Adressierung Daten		ja	ja
remote Quittierung		Applikation	nein
Modell		Client / Server	Client / Server

[Zurück zum Sprungverteiler PB](#)

Erläuterungen zur Tabelle:

Zu (*1): Darin enthalten sind Informationen, welche die Daten (FMS Variable) beschreiben (/12/).

40.3 Überblick Anwenderschnittstellen

Überblick: Kommunikationsbausteine

Tabelle 40-2

Kommunikations- baustein	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 40-3

Schnittstelle	zu finden in STEP 7	
S7-300, CP	STEP 7 (nicht TIA)	Bibliothek: SIMATIC_NET_CP / CP300
S7-400, CP	STEP 7 (nicht TIA)	Bibliothek: SIMATIC_NET_CP / CP400

40.4 Anwenderschnittstelle: READ, WRITE, REPORT

40.4.1 Beschreibung

READ Variable lesen

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.

WRITE Variable schreiben

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.

REPORT Variable melden

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.

40.4.2 Parameter für READ

Tabelle 40-4

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
OUTPUT	Typ	Bemerkung
NDR	BOOL	Daten im Zielbereich (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): remote Quittierung: Applikation

40.4.3 Parameter für WRITE

Tabelle 40-5

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
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): remote Quittierung: Applikation

40.4.4 Parameter für REPORT

Tabelle 40-6

INPUT	Datentyp	Bemerkung
REQ	BOOL	Anstoß Meldeauftrag
ID	DWORD	Referenz auf die zugehörige Verbindung (aus Verbindungsprojektierung in STEP 7)
SD_1	ANY	Quellbereich lokal
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	

Zu (*1): remote Quittierung: Keine

41 PB: DP-Kommunikation

41.1 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 in Kapitel 5.4.

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
 - DP-Master schreibt Daten in DP-Slaves
- Der Datenaustausch erfolgt konsistent über die gesamte Länge (systembedingte Datenkonsistenz)

41.2 Details Kommunikationsart

Tabelle 41-1 Kommunikationsarten-Detail

Kommunikationsart:		DP-Kommunikation (Netz PB)
Protokoll:		DP
Allgemeines		
Schnittstellen		CPU, CP, CM
Anbindung	SIMATIC S5	ja
	Fremd (offene Standards)	ja
Protokoll		
dynamische Datenlänge		nein
Multicast / Broadcast		nein
Verbindungen	zum remote Partner?	nein
	dynamisch / statisch	---
Anwenderschnittstelle		
Kommunikationsbausteine		Ladebefehle / Transferbefehle DPRD_DAT, DPWR_DAT DP_SEND, DP_RECV (nur S7-300 CP)
maximale Anzahl Daten		Siehe technische Daten der CPU
dynamische Adressierung Daten		nein
remote Quittierung		Applikation
Modell		Master / Slave

[Zurück zum Sprungverteiler PB](#)

41.3 Überblick Anwenderschnittstellen

Überblick:

Tabelle 41-2

Anwender-schnittstelle	S7-300		S7-400	S7-1200	S7-1500
	CPU	CP	CPU, CP		
DP_SEND	---	FC 1 (*3)	---	---	---
DP_RECV	---	FC 2 (*3)	---	---	---
DPRD_DAT	SFC 14 (*2)	---	SFC 14 (*2)	(*1)	(*1)
DPWR_DAT	SFC 15 (*2)	---	SFC 15 (*2)	(*1)	(*1)

Die Anwenderschnittstelle ist in STEP 7 zu finden:

(*1): STEP 7 (TIA) : Erweiterte Anweisung

(*2): STEP 7 (nicht TIA): Bibliothek: Standard Library / System Function Blocks

(*3): STEP 7 (nicht TIA): Bibliothek: SIMATIC_NET_CP / CP300

41.4 Anwenderschnittstelle DP_SEND, DP_RECV

41.4.1 Beschreibung

Die Kommunikationsbausteine dienen der Kommunikation zwischen lokaler CPU und lokalem CP. 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.

41.4.2 Parameter für DP_SEND

Tabelle 41-3

INPUT	Typ	Bemerkung
CPLADDR	WORD	projektierte Anfangsadresse des CP
SEND	ANY	Sendebereich
OUTPUT	Typ	Bemerkung
DONE	BOOL	Anzeige, ob Auftrag fehlerfrei ausgeführt wurde (*1)
ERROR	BOOL	Fehleranzeige
STATUS	WORD	Statusanzeige

Zu (*1): remote Quittierung: Applikation

41.4.3 Parameter für DP_RECV

Tabelle 41-4

INPUT	Typ	Bemerkung
CPLADDR	WORD	projektierte Anfangsadresse des CP
RECV	ANY	Empfangsbereich
OUTPUT	Typ	Bemerkung
NDR	BOOL	Anzeige, ob Auftrag fehlerfrei ausgeführt wurde (*1)
ERROR	BOOL	Fehleranzeige
STATUS	WORD	Statusanzeige
DPSTATUS	BYTE	Statusanzeige

Zu (*1): remote Quittierung: Applikation

41.5 Anwenderschnittstelle DPRD_DAT, DPWR_DAT

41.5.1 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

41.5.2 Parameter für DPRD_DAT

Tabelle 41-5

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

Zu (*1): remote Quittierung: Applikation

41.5.3 Parameter für DPWR_DAT

Tabelle 41-6

INPUT	Typ	Bemerkung
LADDR	WORD	Projektierte Anfangsadresse im Ausgangsbereich, auf den geschrieben werden soll
RECORD	ANY	Quellbereich
OUTPUT	Typ	Bemerkung
RET_VAL	INT	Anzeige, ob Auftrag fehlerfrei ausgeführt wurde (*1)

Zu (*1): remote Quittierung: Applikation

42 Serielle Schnittstelle

Hinweis

Die Kopplung zu Controllern mit der Schnittstelle Modbus Seriell (RTU Format) wird in Teil 4 (Kapitel 58) beschrieben:

42.1 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)

42.2 Vergleich Protokolle: *ASCII* / 3964(R) / RK 512

42.2.1 Abgrenzung

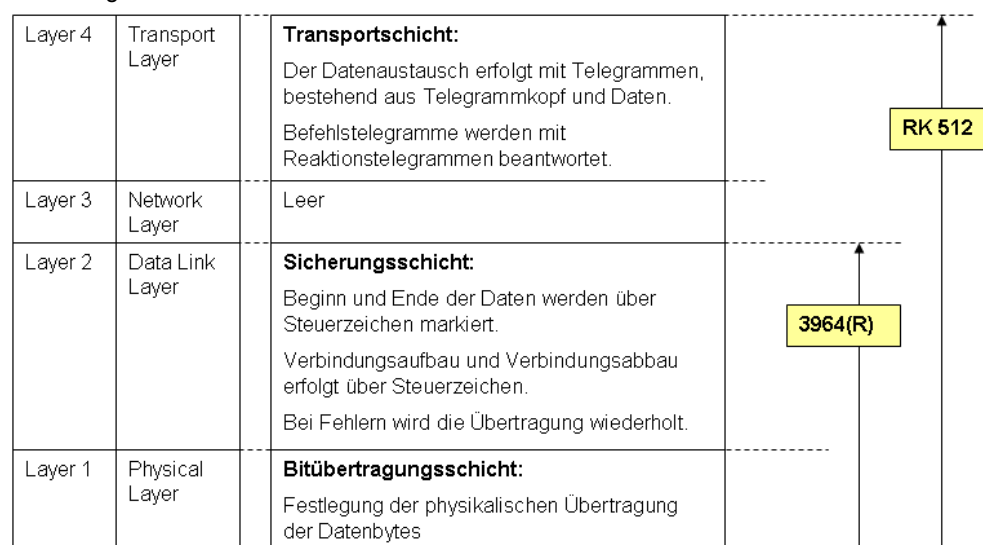
Die Kommunikationsarten lassen sich am Besten mit Hilfe des ISO/OSI-Referenzmodells abgrenzen:

Tabelle 42-1

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 42-1



42.2.2 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.

42.2.3 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.

42.2.4 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.

42.2.5 Details Kommunikationsart

Tabelle 42-2 Kommunikationsarten-Detail

Kommunikationsart:		Serielle Schnittstelle		
Protokoll:		*ASCII*	3964(R)	RK 512
Allgemeines				
Schnittstellen		CPU, CP	CPU, CP	CPU, CP
Anbindung	SIMATIC S5	ja	ja	ja
	Fremd (offene Standards)	ja	ja	nein
Anwenderschnittstelle				
Kommunikationsbaustein		siehe Kapitel 43	siehe Kapitel 43	siehe Kapitel 43
maximale Anzahl Daten		<= 4096 Byte	<= 4096 Byte	<= 4096 Byte
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
Protokoll				
dynamische Datenlänge			ja	
Verbindungen	zum remote Partner?		nein (*2)	
	dynamisch / statisch		---	

[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 STEP 7 eine „PtP-Verbindung“ projiziert. Dies ist keine Verbindung zum remote Partner.

43 Überblick Anwenderschnittstellen

43.1 Einsatz in Stationen mit CPU

Die Tabelle enthält Kombinationen für Stationen, in denen eine CPU gesteckt ist.

Die Möglichkeit, CPs bzw. CMs in dezentralen Stationen ET 200 zu betreiben, wird im folgenden Kapitel 43.2 betrachtet.

Tabelle 43-1

Schnittstelle		Protokoll	Anwenderschnittstelle		(*x)
ET200 CPU	1SI (*101)	*ASCII*, 3964R	S_SEND / S_RCV	FB3 / FB2	(*1)
S7-300	CPU	*ASCII*, 3964R	SEND_PTP / RCV_PTP	SFB60 / SFB61	(*2)
		RK 512	SEND_RK / SERVE_RK	SFB 63 / SFB 65	(*2)
			FETCH_RK / SERVE_RK	SFB 64 / SFB 65	(*2)
	CP 340 (*102)	*ASCII*, 3964R	P_SEND / P_RCV	FB3 / FB2	(*3)
	CP 341 (*102)	*ASCII*, 3964R	P_SND_RK / P_RCV_RK	FB8 / FB7	(*4)
		RK 512	P_SND_RK / P_RCV_RK	FB 8 / FB 7	(*4)
S7-400	CP 440	*ASCII*, 3964R	SEND_440 / REC_440	FB10 / FB9	(*5)
	CP 441	*ASCII*, 3964R	BSEND / BRCV	SFB12 / SFB13	(*6)
		ASCII, 3964R	BSEND / Empfangsfach	SFB12 / ---	(*6)
		RK512	BSEND / BRCV	SFB 12 / SFB 13	(*6)
			BSEND / ---	SFB 12 / ---	(*6)
			PUT	SFB 15 / ---	(*6)
			GET	SFB 14 / ---	(*6)
S7-1200	CPU	Freeport, Modbus RTU	SEND_PTP / RCV_PTP	SFB113/SFB114	(*10)
	CM 1241		MB_MASTER / MB_SLAVE		
S7-1500	CM PtP (*103)	Freeport, 3964R, Modbus RTU	Send_P2P / Receive_P2P Modbus_Master/ Modbus_Slave	FB613 / FB614 FB641 / FB642	(*10)

Hinweis

Die beiden Protokolle *ASCII* und Freeport sind vergleichbar

Erläuterungen zur Tabelle

Die Anwenderschnittstelle ist in STEP 7 zu finden:

Tabelle 43-2

(*x)	
(*1)	STEP7 (nicht TIA): Bibliothek: ET200sSI / ET200S Serial Interface
(*2)	STEP7 (nicht TIA): Bibliothek: Standard Library / System Function Blocks
(*3)	STEP 7 (nicht TIA): Bibliothek: CP PtP / CP 340
(*4)	STEP 7 (nicht TIA): Bibliothek: CP PtP / CP 341
(*5)	STEP 7 (nicht TIA): Bibliothek: CP PtP / CP 440
(*6)	STEP 7 (nicht TIA): Bibliothek: Standard Library / System Function Blocks
(*10)	STEP 7 (TIA): Anweisungen / Kommunikation
(*101) bis (*103)	siehe folgendes Kapitel

43.2 Einsatz in dezentralen Stationen ohne CPU

CPs oder CMs mit serieller Schnittstelle können auch in dezentralen Stationen betrieben werden, in denen keine CPU gesteckt ist.

CPs oder CMs aus der Tabelle im Kapitel 43.1

(*101): auch in dezentraler Station ET 200S

(*102): auch in dezentraler Station ET 200M

(*103): auch in dezentraler Station ET 200MP

Zusätzliche Möglichkeiten

Neben den oben aufgeführten CPs oder CMs gibt es noch folgende Möglichkeiten:

Tabelle 43-3

dezentrale Station	CM	Protokoll	Anwenderschnittstelle	
ET 200SP	CM PtP	Freeport, 3964R	Send_P2P / Receive_P2P	FB613 / FB614

Hinweis

Der CM PtP für ET 200SP hat die gleiche Funktionalität wie der CM PtP für S7-1500 (zentral) bzw. ET 200MP (dezentral).

44 ET 200S: *ASCII* und 3964(R)

44.1 Beschreibung

Der Kommunikationsbaustein S_SEND sendet Daten an den Kommunikationsbaustein S_RCV.

S_SEND Daten senden

S_RCV Daten empfangen

44.2 Parameter für S_SEND

Tabelle 44-1

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse ET 200S 1SI
DB_NO	INT	Sendebereich
DBB_NO	INT	
LEN	INT	Länge der zu sendenden Daten (*2)
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
COM_RST	BOOL	Neustart des FB

Zu (*1): remote Quittierung: bei *ASCII*: keine / bei 3964(R): Transport

Zu (*2): maximale Anzahl Daten: = 224 Byte

44.3 Parameter für S_RCV

Tabelle 44-2

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse ET 200S 1SI
DB_NO	INT	Empfangsbereich
DBB_NO	INT	
OUTPUT	Typ	Bemerkung
LEN	INT	Länge der empfangenen Daten (*2)
NDR	BOOL	Daten im Empfangsbereich
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
COM_RST	BOOL	Neustart des FB

Zu (*1): Daten wurden aus dem Empfangspuffer in den Empfangsbereich kopiert.

Zu (*2): maximale Anzahl Daten: = 224 Byte

45 S7-300 CPU: *ASCII* / 3964(R)

45.1 Beschreibung

Der Kommunikationsbaustein SEND_PTP sendet Daten an den Kommunikationsbaustein RCV_PTP.

SEND_PTP Daten senden

RCV_PTP Daten empfangen

45.2 Parameter für SEND_PTP

Tabelle 45-1

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Auftrag
LADDR	WORD	E/A Adresse des Submoduls
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformationen
STATUS	WORD	
IN_OUT	Typ	Bemerkung
SD_1	ANY	Sendebereich
LEN	INT	Länge der zu sendenden Daten (*2)

Zu (*1): remote Quittierung: bei *ASCII*: keine / bei 3964(R): Transport

Zu (*2): maximale Anzahl Daten: = 1024 Byte

45.3 Parameter für RCV_PTP

Tabelle 45-2

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 (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
RD_1	ANY	Empfangsbereich
LEN	INT	Länge empfangener Daten (*2)

Zu (*1): Daten wurden aus dem Empfangspuffer in den Empfangsbereich kopiert.

Zu (*2): maximale Anzahl Daten: = 1024 Byte

46 S7 300 CPU: RK 512

46.1 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

46.2 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

46.3 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

46.4 Parameter für SEND_RK

Kommunikationsbaustein in remote CPU: SERVE_RK

Tabelle 46-1

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
R_DBNO	INT	
R_OFFSET	INT	
R_CF_BYT	INT	Koppelmerker auf remote CPU
R_CF_BIT	INT	
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
SD_1	ANY	Quellbereich in lokaler CPU
LEN	INT	Länge der zu sendenden Daten (*4)

Zu (*1): remote Quittierung: Applikation

Zu (*2): maximale Anzahl Daten: 1024 Byte

46.5 Parameter für SERVE_RK

Kommunikationsbaustein in remote CPU: SEND_RK

Tabelle 46-2

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 fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
L_TYPE	CHAR	Zielbereich auf lokaler CPU
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 (*2)

Zu (*1): Daten wurden empfangen

Zu (*2): maximale Anzahl Daten: = 1024 Byte

46.6 Parameter für FETCH_RK

Kommunikationsbaustein in remote CPU: SERVE_RK

Tabelle 46-3

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
R_DBNO	INT	
R_OFFSET	INT	
R_CF_BYT	INT	Koppelmerker auf remote CPU
R_CF_BIT	INT	
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
RD_1	ANY	Zielbereich in lokaler CPU
LEN	INT	Länge der zu empfangenen Daten (*2)

Zu (*1): remote Quittierung: Applikation

Zu (*2): maximale Anzahl Daten: = 1024 Byte

46.7 Parameter für SERVE_RK

Kommunikationsbaustein in remote CPU: FETCH_RK

Tabelle 46-4

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 fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
L_TYPE	CHAR	Quellbereich auf lokaler CPU
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 (*2)

Zu (*1): Daten wurden abgeholt.

Zu (*2): maximale Anzahl Daten: = 1024 Byte

47 CP 340: *ASCII* / 3964(R)

47.1 Beschreibung

Der Kommunikationsbaustein P_SEND sendet Daten an den Kommunikationsbaustein P_RCV.

P_SEND Daten senden

P_RCV Daten empfangen

47.2 Parameter für P_SEND

Tabelle 47-1

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse CP 340
DB_NO	INT	Sendegebiet
DBB_NO	INT	
LEN	INT	Länge der zu sendenden Daten (*2)
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): remote Quittierung: bei *ASCII*: keine / bei 3964(R): Transport

Zu (*2): maximale Anzahl Daten: = 1024 Byte

47.3 Parameter für P_RCV

Tabelle 47-2

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse CP 340
DB_NO	INT	Empfangsbereich
DBB_NO	INT	
OUTPUT	Typ	Bemerkung
LEN	INT	Länge empfangener Daten (*2)
NDR	BOOL	Daten im Empfangsbereich (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): Daten wurden aus dem Empfangspuffer in den Empfangsbereich kopiert.

Zu (*2): maximale Anzahl Daten: = 1024 Byte

48 CP 341: *ASCII* / 3964(R)

48.1 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.

48.2 Parameter für P_SND_RK

Tabelle 48-1

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse CP 341
DB_NO	INT	Sendebereich
DBB_NO	INT	
LEN	INT	Länge der zu sendenden Daten (*2)
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): remote Quittierung: bei *ASCII*: keine / bei 3964(R): Transport

Zu (*2): maximale Anzahl Daten: = 1024 Byte

48.3 Parameter für P_RCV_RK

Tabelle 48-2

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse CP 341
DB_NO	INT	Empfangsbereich
DBB_NO	INT	
OUTPUT	Typ	Bemerkung
LEN	INT	Länge der empfangenen Daten (*2)
NDR	BOOL	Daten im Empfangsbereich (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): Daten wurden aus dem Empfangspuffer in den Empfangsbereich kopiert.

Zu (*2): maximale Anzahl Daten: = 1024 Byte

49 CP 341: RK 512

49.1 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.

49.2 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

49.3 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

49.4 Parameter für P_SND_RK

Kommunikationsbaustein in remote CPU: P_RCV_RK

Tabelle 49-1

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
DBB_NO	INT	
LEN	INT	Länge der zu sendenden Daten (*2)
R_CPU_NO	INT	Nummer der remote CPU
R_TYP	CHAR	Zielbereich in remote CPU
R_NO	INT	
R_OFFSET	INT	
R_CF_BYT	INT	Koppelmerker auf remote CPU
R_CF_BIT	INT	
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): remote Quittierung: Applikation

Zu (*4): maximale Anzahl Daten: 4096 Byte

49.5 Parameter für P_RCV_RK

Kommunikationsbaustein in remote CPU: P_SND_RK, mit Parameter SF = „S“

Tabelle 49-2

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 parametrier
DBB_NO	INT	
OUTPUT	Typ	Bemerkung
NDR	BOOL	Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
L_TYP	CHAR	Zielbereich auf lokaler CPU, wenn Datenziel „DB“ beim Sender parametrier
L_NO	INT	
L_OFFSET	INT	
LEN	INT	Länge der empfangenen Daten (*2)
L_CF_BYT	INT	Koppelmerker
L_CF_BIT	INT	

Zu (*1): Daten wurden empfangen

Zu (*2): maximale Anzahl Daten: = 4096 Byte

49.6 Parameter für P_SND_RK

Kommunikationsbaustein in remote CPU: P_RCV_RK

Tabelle 49-3

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
DBB_NO	INT	
LEN	INT	Datenlänge (*2)
R_CPU_NO	INT	Nummer der remote CPU
R_TYP	CHAR	Quellbereich remote CPU
R_NO	INT	
R_OFFSET	INT	
R_CF_BYT	INT	Koppelmerker auf remote CPU
R_CF_BIT	INT	
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): remote Quittierung: Applikation

Zu (*2): maximale Anzahl Daten: 1024 Byte

49.7 Parameter für P_RCV_RK

Kommunikationsbaustein in remote CPU: P_SND_RK, mit Parameter SF = „F“

Tabelle 49-4

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 fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
L_TYP	CHAR	Quellbereich auf lokaler CPU
L_NO	INT	
L_OFFSET	INT	
LEN	INT	Datenlänge (*2)
L_CF_BYT	INT	Koppelmerker
L_CF_BIT	INT	

Zu (*1) Daten wurden abgeholt

Zu (*2): maximale Anzahl Daten: = 4096 Byte

50 CP 440: *ASCII* / 3964(R)

50.1 Beschreibung

Der Kommunikationsbaustein SEND_440 sendet Daten an den Kommunikationsbaustein REC_440.

SEND_440 Daten senden

REC_440 Daten empfangen

50.2 Parameter für SEND_440

Tabelle 50-1

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse CP 440
DB_NO	INT	Sendebereich
DBB_NO	INT	
LEN	INT	Länge der zu sendenden Daten (*2)
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): remote Quittierung: bei *ASCII*: keine / 3964(R): Transport

Zu (*2): maximale Anzahl Daten: = 400 Byte

50.3 Parameter für REC_440

Tabelle 50-2

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
R	BOOL	Abbruch Auftrag
LADDR	INT	Basisadresse CP 440
DB_NO	INT	Empfangsbereich
DBB_NO	INT	
OUTPUT	Typ	Bemerkung
LEN	INT	Länge empfangener Daten (*2)
NDR	BOOL	Daten im Empfangsbereich (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): Daten wurden aus dem Empfangspuffer in den Empfangsbereich kopiert.

Zu (*2): maximale Anzahl Daten: = 400 Byte

51 CP 441: *ASCII* / 3964(R)

51.1 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 parametrier.

Nachteil durch Verwendung eines Empfangsfaches im Empfänger:

- Die Applikation (Anwenderprogramm) kann nicht erkennen, wann eine Datenübertragung stattfindet.

51.2 Parameter für BSEND

Tabelle 51-1

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 fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
SD_1	ANY	Sendebereich
LEN	WORD	Länge der zu sendenden Daten (*2)

Zu (*1): remote Quittierung: bei *ASCII*: keine / bei 3964(R): Transport

Zu (*2): maximale Anzahl Daten: = 4096 Byte

51.3 Parameter für BRCV

Tabelle 51-2

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
*ID 1	WORD	Referenz auf die lokale Verbindungsbeschreibung (wird durch Verbindungsprojektierung in STEP 7 vorgegeben)
R_ID	DWORD	Hier ohne Bedeutung
OUTPUT	Typ	Bemerkung
R e d NDR	BOOL	Daten im Empfangsbereich (*1)
e ERROR	BOOL	Fehlerinformation
e STATUS	WORD	
IN_OUT	Typ	Bemerkung
RD_1	ANY	Empfangsbereich
e LEN	WORD	Länge der empfangenen Daten (*2)

Zu (*1): Daten wurden aus dem Empfangspuffer in den Empfangsbereich kopiert.

Zu (*2): maximale Anzahl Daten: = 4096 Byte

52 CP 441: RK 512

52.1 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

52.2 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

52.3 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.

52.4 Parameter für BSEND

Kommunikationsbaustein in remote CPU: BRCV

Tabelle 52-1

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 fertig (*1)
ERROR	BOOL	Fehlerinformationen
STATUS	WORD	
IN_OUT	Typ	Bemerkung
SD_1	ANY	Quellbereich in lokaler CPU
LEN	WORD	Länge der zu sendenden Daten (*2)

Zu (*1): remote Quittierung: Applikation

Zu (*2): maximale Anzahl gesendeter Daten: 4096 Byte

52.5 Parameter für BRCV

Kommunikationsbaustein in remote CPU: BRCV

Tabelle 52-2

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 fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
RD_1	ANY	Zielbereich in lokaler CPU
LEN	WORD	Länge der empfangenen Daten (*2)

Zu (*1): Daten wurden empfangen

Zu (*2): maximale Anzahl Daten: 4096 Byte

52.6 Parameter für BSEND

Kommunikationsbaustein in remote CPU: ---

Tabelle 52-3

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
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformationen
STATUS	WORD	
IN_OUT	Typ	Bemerkung
SD_1	ANY	Quellbereich in lokaler CPU
LEN	WORD	Länge der zu sendenden Daten (*2)

Zu (*1): remote Quittierung: Applikation

Zu (*2): maximale Anzahl Daten: <= 450 Byte (abhängig von remote CPU)

52.7 Parameter für PUT

Kommunikationsbaustein in remote CPU: ---

Tabelle 52-4

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 fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT		
ADDR_i	ANY	Zielbereiche in der remote CPU (i=1, 2, 3, 4) (*2)
SD_i	ANY	Quellbereiche in der lokalen CPU (i=1, 2, 3, 4) (*2)

Zu (*1): remote Quittierung: Applikation

Zu (*2): maximale Anzahl Daten: <= 450 Byte (abhängig von remote CPU)

52.8 Parameter für GET

Kommunikationsbaustein in remote CPU: ---

Tabelle 52-5

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 fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT		
ADDR_i	ANY	Quellbereiche in der remote CPU (i=1, 2, 3, 4) (*2)
RD_i	ANY	Zielbereiche in der lokalen CPU (i=1, 2, 3, 4) (*2)

Zu (*1): remote Quittierung: Applikation

Zu (*2): maximale Anzahl Daten: <= 450 Byte (abhängig von remote CPU)

53 S7-1200: Freeport

Hinweis: Nur STEP 7 (TIA).

53.1 Beschreibung

Mit den Anweisungen kann das Anwenderprogramm Daten an die lokale Kommunikationsschnittstellen (CM, CP) senden oder von diesen Daten empfangen.

Die Kommunikationsschnittstelle (CM, CP) führt die eigentliche Datenübertragung mit dem remote Kommunikationspartner durch.

SEND_PTP Daten senden

RCV_PTP Daten empfangen

53.2 Parameter für SEND_PTP

Tabelle 53-1

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Sendeauftrag
PORT	PORT	Kennung Kommunikationsport
BUFFER	VARIANT	Sendebereich
LENGTH	UINT	Länge der zu sendenden Daten (*2)
PTRCL	BOOL	---
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): Die Daten wurden an das lokale CM übergeben

Zu (*2): maximale Anzahl Daten: =1024 Byte

53.3 Parameter für RCV_PTP

Tabelle 53-2

INPUT	Typ	Bemerkung
EN_R	BOOL	Anstoß Empfangsauftrag
PORT	PORT	Kennung Kommunikationsport
BUFFER	VARIANT	Empfangsbereich
OUTPUT	Typ	Bemerkung
NDR	BOOL	Auftrag fertig (*1)
LENGTH	UINT	Länge empfangener Daten (*2)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): Die Daten wurden vom lokalen CM abgeholt, und in den Puffer der CPU geschrieben.

Zu (*2): maximale Anzahl Daten: =1024 Byte

54 S7-1500, S7-300, S7-400: Freeport / 3964(R)

Hinweis: Nur STEP 7 (TIA).

54.1 Beschreibung

Mit den Anweisungen kann das Anwenderprogramm Daten an die lokale Kommunikationsschnittstellen (CM, CP) senden oder von diesen Daten empfangen.

Die Kommunikationsschnittstelle (CM, CP) führt die eigentliche Datenübertragung mit dem remote Kommunikationspartner durch.

Send_P2P Daten senden

Receive_P2P Daten empfangen

54.2 Parameter für Send_P2P

Tabelle 54-1

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
REQ	BOOL	BOOL	Anstoß Sendeauftrag
PORT	WORD	PORT	Kennung Kommunikationsport
BUFFER	ANY	VARIANT	Sendebereich
LENGTH	WORD	UINT	Länge der zu sendenden Daten (*2)
OUTPUT			Bemerkung
DONE	BOOL	BOOL	Auftrag fertig (*1)
ERROR	BOOL	BOOL	Fehlerinformation
STATUS	WORD	WORD	
IN_OUT			Bemerkung
COM_RST	BOOL	---	Initialisierung

Zu (*1): Die Daten wurden an das lokale CM übergeben und erfolgreich gesendet.

Zu (*2): maximale Anzahl Daten: ≤ 4096 (abhängig von CPU, CM)

54.3 Parameter für Receive_P2P

Tabelle 54-2

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
PORT	WORD	PORT	Kennung Kommunikationsport
BUFFER	ANY	VARIANT	Empfangsbereich
OUTPUT			Bemerkung
NDR	BOOL	BOOL	Auftrag fertig (*1)
LENGTH	WORD	UINT	Länge empfangener Daten (*2)
ERROR	BOOL	BOOL	Fehlerinformation
STATUS	BOOL	WORD	
IN_OUT			Bemerkung
COM_RST	BOOL	---	Initialisierung

Zu (*1): Die Daten wurden vom lokalen CM abgeholt, und in den Puffer der CPU geschrieben.

Zu (*2): maximale Anzahl Daten: ≤ 4096 (abhängig von CPU, CM)

55 Informationen zu Teil 3

In den Tabellen sind Verweise auf Informationen zu den Themen aus Teil 3 (Kommunikationsarten) zu finden. Alle Verweise /x/ sind zentral im Kapitel 61 hinterlegt. Dort sind auch die entsprechenden Links ins Internet zu finden.

Tabelle 55-1

/x/	Titel	Informationen zu
---	STEP 7 Online Hilfe	Kommunikationsarten, Kommunikationsbausteine
/6/	System- und Standardfunktionen für S7-300/400	
/13/	SIMATIC NET Funktionen (FC) und Funktionsbausteine (FB) für SIMATIC NET S7-CPs	
/40/	SIMATIC S7-1500 ET 200MP, ET 200SP Kommunikation	
/100/	FAQs für S7-300 CPUs	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	Verbindungen projektieren
/106/	FAQs für PROFIBUS S7-300/400 CPs	
/119/	FAQs zu WinAC RTX	

56 ***** TEIL 4: Fremde Controller *****

56.1 Gliederung und Inhalt

Tabelle 56-1

Kapitel	Gliederung	Inhalt
57	Modbus/TCP	Kopplung über Netz PN/IE: <ul style="list-style-type: none"> • Merkmale • Anwenderschnittstellen
58	Modbus Seriell (RTU Format)	Kopplung über Serielle Schnittstelle: <ul style="list-style-type: none"> • Merkmale • Anwenderschnittstellen
59	Informationen	Gerätehandbücher, FAQs, Applikationen,

56.2 Vorbemerkung

Offene Standards

Ein SIMATIC Controller kann über offene Standards 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)

57 Modbus/TCP

57.1 Merkmale

Allgemeines

Modbus ist ein weltweit verbreitetes Protokoll, welches allen Anwendern offen gelegt ist. Modbus/TCP ermöglicht die Kommunikation über TCP/IP Netze.

Ein SIMATIC Controller kann sein:

- Modbus Server
- Modbus Client

Mit einem Modbus TCP Auftrag maximal übertragbare Daten:

Tabelle 57-1

Auftrag	Übertragung bitweise	Übertragung wortweise
Lesender Auftrag	250 Byte	250 Byte
Schreibender Auftrag	100 Byte	200 Byte

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/](#))

57.2 Überblick Anwenderschnittstellen

Tabelle 57-2

Schnittstelle		Anwenderschnittstelle
ET 200 CPU	• integrierter PN/IE Schnittstelle der CPU	Modbus Bausteine für CPU (*1)
S7-300	• integrierter PN/IE Schnittstelle der CPU • CP 343-1	
S7-400	• integrierter PN/IE Schnittstelle der CPU • CP 443-1	
S7-1200	• integrierter PN/IE Schnittstelle der CPU	Anweisungen in STEP 7 (TIA)

Zu (*1): Für die Modbus/TCP Kommunikation gibt es eigene Funktionsbausteine (Modbus Bausteine). Die Modbus Bausteine sind nicht in STEP 7 enthalten, sie müssen getrennt bestellt werden. Informationen zur Funktionalität und zur Bestellung sind unter [/11/](#) zu finden.

57.3 S7-1200: Modbus Client

57.3.1 Beschreibung

Die Anweisung MB_CLIENT kommuniziert als Modbus Client über die PN Schnittstelle der CPU.

57.3.2 Parameter MB_CLIENT

Tabelle 57-3

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Auftrag
DISCONNECT	BOOL	Verbindungsaufbau, Verbindungsabbau
CONNECT_ID	UINT	Identifikation der Verbindung
IP_OCTET_1	USINT	IP Adresse des Modbus Servers
IP_OCTET_2	USINT	
IP_OCTET_3	USINT	
IP_OCTET_4	USINT	
IP_PORT	UINT	Port des Modbus Servers
MB_MODE	USINT	Modus der Anfrage (Lesen, Schreiben, Diagnose)
MB_DATA_ADR	UDINT	Anfangsadresse der Daten, auf welche die Anweisung MB_CLIENT zugreift
DATA_LEN	UINT	Datenlänge
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag läuft / Auftrag fertig (*1)
BUSY	BOOL	
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
MB_DATA_PTR	VARIANT	Puffer für die vom Modbus Server zu empfangenden Daten, bzw. auf die zum Modbus Server zu sendenden Daten

57.4 S7-1200: Modbus Server

57.4.1 Beschreibung

Die Anweisung MB_SEVER kommuniziert als Modbus Server über die PN Schnittstelle der CPU.

57.4.2 Parameter MB_SERVER

Tabelle 57-4

INPUT	Typ	Bemerkung
DISCONNECT	BOOL	Reaktion auf Verbindungsanfrage: Verbindungsaufbau, Verbindungsabbau
CONNECT_ID	UINT	Identifikation der Verbindung
IP_PORT	UINT	Port des Modbus Clients
OUTPUT	Typ	Bemerkung
NDR	BOOL	Neue Daten durch Modbus Client geschrieben
DR	BOOL	Daten durch den Modbus Client gelesen
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
MB_DATA_PTR	VARIANT	Puffer für die vom Modbus Server zu empfangenden Daten, bzw. auf die zum Modbus Server zu sendenden Daten

58 Modbus Seriell (RTU Format)

58.1 Merkmale

Allgemeines

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 58-1

Auftrag	Operand	
	Eingänge, Zeiten, Zähler	Ausgänge, Merker, Datenbausteine
Lesen	x	x
Schreiben	---	x

Ein SIMATIC Controller kann sein:

- Modbus Master
- Modbus Slave

Adressierung Modbus Slave

Die Adresse eines Modbus Slaves kann im folgenden Bereich liegen:

- CP341, CP441-2: 1 bis 255
- CM PtP, ET200SP/CM PtP: 1 bis 247 (*1)

Zu (*1):

1 bis 65535, bei EXTENDED_ADDRESSING=true

Mit der Modbus Slave Adresse Null spricht der Master alle Slaves am Bus an (Broadcast).

58.2 Überblick Anwenderschnittstellen

58.2.1 Anschluss über CP bzw. CM

Zur Kommunikation über Modbus Seriell (RTU Format) wird ein SIMATIC CP bzw. CM und der entsprechende Treiber benötigt. Der Treiber ist entweder auf dem SIMATIC CP integriert, oder er ist 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

Zentrale Stationen:

- S7-1200: CM 1241, CB 1241
- S7-1500: CM PtP HF

Dezentrale Stationen:

- ET 200S: 1SI-Modul
- ET 200SP: CM PtP
- ET 200MP: CM PtP HF

SIMATIC CPs mit der Möglichkeit Treiber für Modbus Seriell zu laden

Zentrale Stationen:

- S7-300: CP 341
- S7-400: CP 441-2

Dezentrale Stationen:

- ET 200S: CP 341

58.2.2 Modbus Master

Überblick Anwenderschnittstellen:

Tabelle 58-2

Schnittstelle		Anwenderschnittstelle		(*x)
ET 200S	1SI	S_SEND / S_RCV	FB 3 / FB 2	(*1)
S7-300 ET200S	CP 341	P_SND_RK / P_RCV_RK	FB 8 / FB 7	(*2)
S7-400	CP 441-2	BSEND / BRCV	SFB 12 / SFB 13	(*3)
S7-1200	CM 1241	MB_MASTER		(*4)
S7-1500 ET 200MP	CM PtP HF	Modbus_Master	FB 641	(*4)
ET 200SP	CM PtP			

Erläuterungen zur Tabelle

Die Anwenderschnittstelle ist in STEP 7 zu finden:

Tabelle 58-3

(*x)	
(*1)	STEP 7 (nicht TIA): Bibliothek: ET200sSI / ET200S Serial Interface
(*2)	STEP 7 (nicht TIA): Bibliothek: CP PtP / CP 341
(*3)	STEP 7 (nicht TIA): Bibliothek: Standard Library / System Function Blocks
(*4)	STEP 7 (TIA): Anweisungen / Kommunikation

58.2.3 Modbus Slave

Überblick Anwenderschnittstelle:

Tabelle 58-4

Schnittstelle		Anwenderschnittstelle		(*x)
ET200S	1SI	S_MODB	FB 81	(*1)
		S_SEND / S_RCV	FB 3 / FB 2	(*1)
S7-300 ET 200S	CP 341	MODB_341	FB 80	(*2)
		P_SND_RK / P_RCV_RK	FB 8 / FB 7	(*3)
S7-400	CP 441-2	MODB_441	FB 180	(*2)
S7-1200	CM 1241	MB_SLAVE		(*4)
S7-1500 ET 200MP	CM PtP HF	Modbus_Slave	FB 642	(*4)
ET 200SP	CM PtP			

Erläuterungen zur Tabelle

Die Anwenderschnittstelle ist in STEP 7 zu finden:

Tabelle 58-5

(*x)	
(*1)	STEP 7 (nicht TIA): Bibliothek: ET200sSI / ET200S Serial Interface
(*2)	STEP 7 (nicht TIA): Nach Installation der Modbus Slave CD wird der FB in der Bibliothek „Modbus“ bereitgestellt.
(*3)	STEP 7 (nicht TIA): Bibliothek: CP PtP / CP 341
(*4)	STEP 7 (TIA): Anweisungen / Kommunikation

58.3 1SI: Modbus Master

Die Anwenderschnittstelle ist identisch mit der Anwenderschnittstelle für *ASCII* und 3964(R): Siehe Kapitel 43.

58.4 1SI: Modbus Slave

58.4.1 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. 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 43).

58.4.2 Parameter S_MODB

Tabelle 58-6

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 beendet
CP_START_OK	BOOL	Fehlerinformation
CP_START_ERROR	BOOL	
ERROR_NR	WORD	Fehlerinformation
ERROR_INFO	WORD	

58.5 CP 341: Modbus Master

58.5.1 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.

P_SND_RK Auftrag an Modbus Slave senden

P_RCV_RK Antworttelegramm vom Modbus Slave empfangen

58.5.2 Parameter P_SND_RK

Tabelle 58-7

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 fertig(*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): „Auftrag fertig“:

Bei schreibenden Funktionscodes: Nach Empfang des Antworttelegrammes.

Bei lesenden Funktionscodes: Nach Empfang des Antworttelegrammes und Übergabe der empfangenen Daten in die CPU

58.5.3 Parameter P_RCV_RK

Tabelle 58-8

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	Antworttelegramm empfangen
LEN	INT	Länge der empfangenen Daten
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

58.6 CP 341: Modbus Slave

58.6.1 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. 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 48).

58.6.2 Parameter FB80

Tabelle 58-9

INPUT	Typ	Bemerkung
LADDR	INT	Basisadresse des CP 341
START_TIMER	TIMER	Überwachungszeit
START_TIME	S5TIME	
OB_MASK	BOOL	Peripheriezugriffsfehler maskieren, Alarime verzögern.
CP_START	BOOL	Initialisierung
CP_START_FM	BOOL	
OUTPUT	Typ	Bemerkung
CP_START_NDR	BOOL	Modbus Auftrag: beendet
CP_START_OK	BOOL	Fehlerinformation
CP_START_ERROR	BOOL	
ERROR_NR	WORD	Fehlerinformation
ERROR_INFO	WORD	

58.7 CP 441-2: Modbus Master

58.7.1 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.

BSEND Auftrag an Modbus Slave senden

BRCV Antworttelegramm vom Modbus Slave empfangen

58.7.2 Parameter BSEND

Tabelle 58-10

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 fertig (*1)
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
SD_1	ANY	Sendebereich
LEN	WORD	Länge des zu sendenden Daten

Zu (*1): Bei schreibenden Funktionscodes: Nach Empfang Antworttelegramm.
Bei lesenden Funktionscodes: Nach Empfang des Antworttelegrammes und Übergabe der empfangenen Daten in die CPU

58.7.3 Parameter BRCV

Tabelle 58-11

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	Antworttelegramm empfangen
ERROR	BOOL	Fehlerinformation
STATUS	WORD	
IN_OUT	Typ	Bemerkung
RD_1	ANY	Empfangsbereich in lokaler CPU
LEN	WORD	Länge der empfangenen Daten

58.8 CP 441-2: Modbus Slave

58.8.1 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.

FB180

Ausführen eines Auftrages vom Modbus Master.

58.8.2 Parameter FB180

Tabelle 58-12

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, Alarmer verzögern.
CP_START	BOOL	Initialisierung
CP_START_FM	BOOL	
OUTPUT	Typ	Bemerkung
CS_START_NDR	BOOL	Modbus Auftrag: beendet
CP_START_OK	BOOL	Fehlerinformation Initialisierung
CP_START_ERROR	BOOL	
ERROR_NR	WORD	Fehlerinformation Auftrag
ERROR_INFO	WORD	

58.9 CM 1241: Modbus Master

58.9.1 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.

MB_MASTER

Auftrag an Modbus Slave senden

Antworttelegramm vom Modbus Slave empfangen

58.9.2 Parameter MB_MASTER

Tabelle 58-13

INPUT	Typ	Bemerkung
REQ	BOOL	Anstoß Auftrag
MB_ADR	UINT	Modbus Stationsadresse
MODE	USINT	Auswahl Modus
DATA_ADDR	UDINT	Anfangsadresse im Slave
DATA_LEN	UINT	Länge der zu lesenden / schreibenden Daten
DATA_PTR	VARIANT	Empfangspuffer (lesen) / Sendepuffer (schreiben)
OUTPUT	Typ	Bemerkung
DONE	BOOL	Auftrag fertig (*1)
BUSY	BOOL	Auftrag läuft
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

Zu (*1): Fallunterscheidung:

Bei schreibenden Funktionscodes: Nach Empfang Antworttelegramm

Bei lesenden Funktionscodes: Nach Empfang des Antworttelegrammes und Übergabe der empfangenen Daten in die CPU

58.10 CM 1241: Modbus Slave

58.10.1 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.

58.10.2 Parameter MB_SLAVE

Tabelle 58-14

INPUT	Typ	Bemerkung
MB_ADDR	UINT	Modbus Stationsadresse
MB_HOLD_REG	VARIANT	Modbus Haltereister DB
OUTPUT	Typ	Bemerkung
NDR	BOOL	Modbus Schreibauftrag: beendet
DR	BOOL	Modbus Leseauftrag: beendet
ERROR	BOOL	Fehlerinformation
STATUS	WORD	

58.11 CM PtP / CM PtP HF: Modbus Master

Hinweis

CM PtP: CM in ET 200SP

CM PtP HF: CM in S7-1500 (zentral), oder CM in ET 200MP (dezentral)

58.11.1 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.

Modbus_Master

Auftrag an Modbus Slave senden

Antworttelegramm vom Modbus Slave empfangen

58.11.2 Parameter Modbus_Master

Tabelle 58-15

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
REQ	BOOL	BOOL	Anstoß Auftrag
MB_ADR	WORD	UINT	Modbus Stationsadresse
MODE	BYTE	USINT	Auswahl Modus
DATA_ADDR	DWORD	UDINT	Anfangsadresse im Slave
DATA_LEN	WORD	UINT	Länge der zu lesenden / schreibenden Daten
OUTPUT			Bemerkung
DONE	BOOL	BOOL	Auftrag fertig (*1)
BUSY	BOOL	BOOL	Auftrag läuft
ERROR	BOOL	BOOL	Fehlerinformation
STATUS	WORD	WORD	
IN_OUT			Bemerkung
DATA_PTR	ANY	VARIANT	Empfangspuffer (lesen) / Sendepuffer (schreiben)
COM_RST	BOOL	---	Initialisierung

Zu (*1): Fallunterscheidung:

Bei schreibenden Funktionscodes: Nach Empfang Antworttelegramm

Bei lesenden Funktionscodes: Nach Empfang des Antworttelegrammes und Übergabe der empfangenen Daten in die CPU

58.12 CM PtP / CM PtP HF: Modbus Slave

Hinweis

CM PtP: CM in ET 200SP

CM PtP HF: CM in S7-1500 (zentral), oder CM in ET 200MP (dezentral)

58.12.1 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.

Modbus_Slave

Ausführen eines Auftrages vom Modbus Master.

58.12.2 Parameter Modbus_Slave

Tabelle 58-16

INPUT	Typ		Bemerkung
	S7-300, S7-400	S7-1200, S7-1500	
MB_ADDR	WORD	UINT	Modbus Stationsadresse
OUTPUT			Erläuterung
NDR	BOOL	BOOL	Modbus Schreibauftrag: beendet
DR	BOOL	BOOL	Modbus Leseauftrag: beendet
ERROR	BOOL	BOOL	Fehlerinformation
STATUS	WORD	WORD	
IN_OUT			Bemerkung
MB_HOLD_REG	ANY	VARIANT	Modbus Haltereister DB
COM_RST	BOOL	---	Initialisierung

59 Informationen zu Teil 4

In den Tabellen sind Verweise auf Informationen zu den Themen aus Teil 4 (fremde Controller) zu finden. Alle Verweise /x/ sind zentral im Kapitel 61 hinterlegt. Dort sind auch die entsprechenden Links ins Internet zu finden.

Tabelle 59-1

/x/	Titel	Informationen zu
/17/	Kommunikation zwischen SIMATIC S7 und einer Modicon M340 über Modbus TCP	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	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

60 ***** TEIL 5: Anhang *****

60.1 Gliederung und Inhalt

Tabelle 60-1

Kapitel	Gliederung	Inhalt
61	Literaturangaben	Verweis im Text: /x/
62	Zentrale Begriffe	Kurze Erläuterung. Muss ein Begriff detaillierter beschrieben werden, dann siehe Kapitel Hintergrundwissen.
63	Abkürzungen	
64	Hintergrundwissen	Beschreibung wichtiger Zusammenhänge
65	Betrachtete Komponenten	Bestelldaten und Versionen
66	Historie	Änderungen / Versionen der Dokumentation

61 Literaturangaben

Informationen

Die folgende Tabelle enthält Links zu den Schwerpunkten:
Kataloge, Broschüren, Handbücher, Applikationen, Tools

Tabelle 61-1

/x/	Titel	Link
/0/	Siemens Industry Online Support	http://support.automation.siemens.com
/1/	SIMATIC Controller / Die innovative Lösung für alle Automatisierungsaufgaben. (Übersicht SIMATIC Controller)	https://www.automation.siemens.com/sales-material-as/brochure/de/brochure_simatic-controller_overview_de.pdf
/2/	SIMATIC NET / Industrielle Kommunikation, Broschüre (Übersicht zur industriellen Kommunikation)	https://www.automation.siemens.com/mcms/infocenter/dokumentencenter/sc/ic/Documents/20Brochures/6ZB5530-1AE01-0BB5_K-Schrift_DE.pdf
/3/	SIMATIC / Kommunikation mit SIMATIC Systemhandbuch (Grundlagen zur Kommunikation mit SIMATIC)	http://support.automation.siemens.com/WW/view/de/25074283
/4/	Katalog ST 70 / Produkte für Totally Integrated Automation und Micro Automation (Übersicht und Bestelldaten für SIMATIC Controller)	https://www.automation.siemens.com/mcms/infocenter/content/de/Seiten/order_form.aspx?nodeKey=key_516908&infotype=1
/5/	Katalog IK PI / Industrielle Kommunikation (Übersicht und Bestelldaten für Geräte der industriellen Kommunikation)	https://www.automation.siemens.com/mcms/infocenter/content/de/Seiten/order_form.aspx?nodeKey=key_517518&infotype=1
/6/	SIMATIC System- und Standardfunktionen für S7-300/400, Referenzhandbuch	http://support.automation.siemens.com/WW/view/de/44240604
/7/	CPU 31xC und CPU 31x Technische Daten, Gerätehandbuch	http://support.automation.siemens.com/WW/view/de/12996906
/8/	Automatisierungssystem S7-400 CPU-Daten, Gerätehandbuch	http://support.automation.siemens.com/WW/view/de/23904550
/11/	S7 OpenModbus/TCP (Produktbeschreibung, Technische Daten, Bestelldaten, Kontakt, Downloads)	http://www.industry.siemens.com/industry solutions/global/de/it4industry/produkte/simatic_add_ons/s7_open_modbus_tcp/seiten/default_tab.aspx
/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	http://support.automation.siemens.com/WW/view/de/30564821
/15/	SIMATIC / Hardware konfigurieren und Verbindungen projektieren mit STEP 7 V5.5, Handbuch	http://support.automation.siemens.com/WW/view/de/45531110
/16/	Von PROFIBUS DP nach PROFINET IO, Programmierhandbuch	http://support.automation.siemens.com/WW/view/de/19289930
/17/	Kommunikation zwischen SIMATIC S7-300/400 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
/21/	S7-300 CPU 31xC Technologische Funktionen (CPU 312C, CPU 313C, CPU 314C)	http://support.automation.siemens.com/WW/view/de/12429336
/22/	SIMATIC NET Programmbausteine für SIMATIC NET S7-CPs Programmierhandbuch	http://support.automation.siemens.com/WW/view/de/62543517

/x/	Titel		Link
/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
/32/	SIMATIC STEP 7 Professional V2.0, Systemhandbuch (Ausdruck der Onlinehilfe)		http://support.automation.siemens.com/WW/view/de/68113685
/33/	SIMATIC Programmieren mit STEP 7 V5.5, Handbuch		http://support.automation.siemens.com/WW/view/de/45531107
/34/	S7-1200 Automatisierungssystem, Systemhandbuch		http://support.automation.siemens.com/WW/view/de/36932465
/35/	S7-1500 Automatisierungssystem, Systemhandbuch		http://support.automation.siemens.com/WW/view/de/59191792
/36/	SIMATIC WinAC RTX (F) 2010, Betriebsanleitung		http://support.automation.siemens.com/WW/view/de/43715176
/37/	PROFINET Systemanschluss für SIMATIC S7	S7-300 (Handbücher)	http://support.automation.siemens.com/WW/view/de/58686942/133300
		S7-400 (Handbücher)	http://support.automation.siemens.com/WW/view/de/58686811/133300
		PC-basierte Systeme (Handbücher)	http://support.automation.siemens.com/WW/view/de/58691933/133300
/38/	PROIBUS Systemanschluss für SIMATIC S7	S7-1200 (Handbücher)	http://support.automation.siemens.com/WW/view/de/44632196/133300
		S7-1500 (Handbücher)	http://support.automation.siemens.com/WW/view/de/67744877/133300
		S7-300 (Handbücher)	http://support.automation.siemens.com/WW/view/de/43484958/133300
		S7-400 (Handbücher)	http://support.automation.siemens.com/WW/view/de/43484515/133300
		S7-mEC (Handbücher)	http://support.automation.siemens.com/WW/view/de/43482516/133300
/39/	Serielle Kommunikation	CP340 (Handbücher)	http://support.automation.siemens.com/WW/view/de/24283637/133300
		CP341 (Handbücher)	http://support.automation.siemens.com/WW/view/de/24284824/133300
		Ladbare Treiber für CP 441-2 und CP 341 (Handbücher)	http://support.automation.siemens.com/WW/view/de/10805420/133300
		CP440 (Handbücher)	http://support.automation.siemens.com/WW/view/de/24254956/133300
		CP441 (Handbücher)	http://support.automation.siemens.com/WW/view/de/24255094/133300
		S7-1500 CM PtP (Handbücher)	http://support.automation.siemens.com/WW/view/de/67308868/133300
		ET200SP (Handbücher)	http://support.automation.siemens.com/WW/view/de/58532616/133300
		ET200S (Handbücher)	http://support.automation.siemens.com/WW/view/de/9260793
		PtP-Kommunikationsanweisungen im Einsatz auf dezentraler Peripherie einer S7-300/400	http://support.automation.siemens.com/WW/view/de/69124220
		S7-1500 / ET 200MP / ET 200SP CM PtP im Betrieb mit PROFINET	http://support.automation.siemens.com/WW/view/de/59062563

/x/	Titel	Link
	Controller	
/40/	SIMATIC S7-1500, ET 200MP, ET 200SP Kommunikation	http://support.automation.siemens.com/WW/view/de/59192925

FAQ

Die folgende Tabelle enthält Links zu FAQs.

Tabelle 61-2

/x/	Titel	Link
	Kommunikation projektieren und programmieren: Kommunikationsbausteine verwenden	
/100/	S7-300 CPU31x	http://support.automation.siemens.com/WW/view/de/22866139
/101/	S7-400 CPU41x	http://support.automation.siemens.com/WW/view/de/23522717
/102/	IE S7-300/400 CPs	http://support.automation.siemens.com/WW/view/de/22548794
/103/	PB S7-300/400 CPs	http://support.automation.siemens.com/WW/view/de/21629966
	Kommunikation projektieren und programmieren: Verbindungen projektieren:	
/105/	IE S7-300/400 CPs	http://support.automation.siemens.com/WW/view/de/22387424
/106/	PB S7-300/400 CPs	http://support.automation.siemens.com/WW/view/de/28526800
/119/	WinAC RTX	http://support.automation.siemens.com/WW/view/de/23337258
	MODBUS	
/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

Applikationsbeispiele

Im Siemens Industry Online Support sind zahlreiche Applikationsbeispiele zu finden. Diese Applikationsbeispiele beschreiben reale, funktionsfähige und branchenneutrale Lösungen. Sie bestehend aus Lösungsweg, Leistungsdaten, Projektierungsanleitung und getestetem Programmcode.

Tabelle 61-3

/x/	Inhalt	Link
/200/	Applikationsbeispiele zur Kommunikation	http://support.automation.siemens.com/WW/view/de/20229805/136000

62 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 62-1

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.	64.2
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 zentralen Stationen über PROFINET IO oder PROFIBUS DP 	5
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 	5
Daten	Mit Daten sind gemeint: Nettodaten, Nutzdaten, Anwenderdaten, SIMATIC Anwenderdatenbereiche Beispiele: Datenbaustein, Merker, Eingänge, Zeiten	---
Hardware-Konfiguration von STEP 7	Hier ist der Teil von STEP 7 gemeint, mit dem man die folgenden Aufgaben erledigen kann: <ul style="list-style-type: none"> Geräte konfigurieren und parametrieren Geräte miteinander vernetzen. 	
I-Slave I-Device	dezentrale Station mit CPU: I-Slave bei PROFIBUS I-Device bei PROFINET	
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 ohne Lizenz nutzen Die Protokolle sind herstellerunabhängig. 	---
Kommunikationsbausteine	Funktionsbausteine (FB, SFB, FC, SFC), zur Einbindung in das STEP 7 Anwenderprogramm. Damit wird die Datenübertragung realisiert (senden, empfangen).	---
Medium	<ul style="list-style-type: none"> Netze: MPI, PB, PN/IE Rückwandbus Serielle Schnittstelle 	---

Begriff	Erläuterung	Details
Netze	Mit Netze sind hier industrielle Netze gemeint. Diese Netze werden in der Automatisierungstechnik eingesetzt. Ein Netz kann aus einem oder mehreren Subnetzen bestehen.	1.2
Partner = Kommunikationspartner	Teilnehmer an der Kommunikation, bei der Daten ausgetauscht werden	---
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	Der Rückwandbus verbindet die Baugruppen (CPU, CP, ...) eines Controllers,	5
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 oder CM 	---
Send/Receive-Bausteine	Sammelbegriff für folgende Kommunikationsbausteine: AG_SEND, AG_LSEND, AG_SSEND, AG_RECV, AG_LRECV, AG_SRECV	
STEP 7	STEP 7 ist das Engineering Tool für SIMATIC Controller. Im Dokument werden folgende Abkürzungen verwendet: "STEP 7": STEP 7 bis V5.5 und/oder ab V10 "STEP 7 (nicht TIA)": STEP 7 nur bis V5.5 "STEP 7 (TIA)": STEP 7 nur ab V10	
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	
Typ "USEND / URCV"	Unter der Bezeichnung Typ "USEND / URCV" sind alle Varianten der Kommunikationsbausteine (FBx, SFBx) und der Bezeichnungen in STEP 7 (USEND, USEND_E, ...) zusammengefasst	
Typ "PUT, GET"	Unter der Bezeichnung Typ "PUT, GET" sind alle Varianten der Kommunikationsbausteine (FBx, SFBx) und der Bezeichnungen in STEP 7 (PUT, PUT_E, ...) zusammengefasst.	
Verbindung	CPU-CPU Kommunikation erfolgt bei der SIMATIC im Allgemeinen über Verbindungen. Eine Verbindung legt fest, wo die Endpunkte der Kommunikation liegen.	6
Verbindungsbausteine	Funktionsbausteine (FB, SFB, FC, SFC), zur Einbindung in das STEP 7 Anwenderprogramm. Damit werden die Verbindungen eingerichtet und verwaltet.	---

63 Abkürzungen

63.1 Im gesamten Dokument

Die Tabelle enthält Abkürzungen, die im gesamten Dokument verwendet werden.

Tabelle 63-1

Abkürzung	Erläuterung
ASCII	Im vorliegenden Dokument wird die Abkürzung *ASCII* für eine Kommunikationsart verwendet: *ASCII* steht im Dokument für eine serielle Datenübertragung, bei der die übertragenen Zeichen ASCII codiert sind.
ASCII	American Standard Code for Information Interchange
Box PC	Embedded Box PC-Bundles
CBA	PROFINET CBA (Component Based Automation)
CP	communication processor: Baugruppe, die sich mit Kommunikationsaufgaben befasst, und einen Controller an ein Medium anbindet.
CPU	central processing 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
IOC	PROFINET IO Controller
IOD	PROFINET IO Device
IoT	ISO-on-TCP (Die Bezeichnung IoT wurde im Dokument aus Platzgründen gewählt.)
IPC	Industrie PC
MPI	multi point interface
OP	Bediengerät
Panel PC	Embedded Panel PC-Bundles
PB	PROFIBUS
PG	Programmiergerät
PN/IE	PROFINET / Industrial Ethernet
PNIO	PROFINET IO
S7-CP	CP der SIMATIC S7
S7-CPU	CPU der SIMATIC S7
S7-mEC	S7-modular Embedded Controller
TIA	Totally Integrated Automation

63.2 Nur in Tabellen

Die folgende Tabelle enthält Abkürzungen, die nur in den Tabellen Schnittstellen und Kombinationen verwendet werden. Aus Platzgründen mussten in diesen Tabellen eigene Abkürzungen eingeführt werden.

Tabelle 63-2

Abkürzung	Bedeutung
C1	Controller 1
C2	Controller 2
C1 Server	Controller 1 ist Server
C2 Server	Controller 2 ist Server
DP	DP-Kommunikation
GD	Globaldaten-Kommunikation
IOD	Die Abkürzung hat hier die Bedeutung: IO-Controller als I-Device
OC	Offene-Kommunikation. Der Begriff steht für die Kommunikationsarten: <ul style="list-style-type: none"> • Offene-Kommunikation mit T-Bausteinen • Offene-Kommunikation mit Send/Receive-Bausteinen
PN	PN-Kommunikation: Kommunikation zwischen PROFINET IO Controller und PROFINET IO Device.
S7	S7-Kommunikation
S7 B	S7-Basiskommunikation
S7 Basis	
Schnittstelle: IE	PROFINET Schnittstelle ohne PN Funktionalität
Schnittstelle: PN	PROFINET Schnittstelle mit PN Funktionalität
X: IE	Abkürzung für: "Schnittstelle: IE"
X: PN	Abkürzung für: "Schnittstelle: PN"

64 Hintergrundwissen

Im Kapitel wird Hintergrundwissen vermittelt.

64.1 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 64-1

Schicht	Bezeichnung	Aufgabe (Beispiele)	Einordnung
Schicht 7 (layer 7)	Anwendungsschicht (Application Layer)	Schnittstelle zum STEP 7 Anwenderprogramm (Quittierung auf Anwendungsebene, ...).	Anwen- dungsorien- tiert
Schicht 6 (layer 6)	Darstellungsschicht (Presentation Layer)	Interpretation der Daten (Umsetzung der normierten Darstellung des Kommunika- tionssystems in eine gerätespezifische Form.)	
Schicht 5 (layer 5)	Kommunikations- steuerungsschicht (Session Layer)	Organisation des Datenaustausches. Verhalten bei Störungen und Verbindungsabbruch.	
Schicht 4 (layer 4)	Transportschicht (Transport Layer)	Herstellung einer Transportverbindung 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) • Flusskontrolle • Quittierung auf Transportebene 	Transport- orientiert
Schicht 3 (layer 3)	Netzwerkschicht (Network Layer)	Vermittlung und Zustellung von Daten: <ul style="list-style-type: none"> • Festlegung der Kommunikationswege (Routing) • Adressierung der Kommunikationspartner im Netz 	
Schicht 2 (mac / layer 2)	Datensicherungs- schicht (Data Link Layer)	Überwachung und Organisation des Zugriffs auf das Übertragungsmedium Korrekte Übermittlung der Daten (Prüfsumme, ...)	
Schicht 1 (layer 1)	Physikalische Schicht (Physical Layer)	Festlegung der physikalische Verbindung zwischen zwei Geräten (Übertragungsmedium, Baudrate, ...)	

64.2 Kommunikationsmodelle

Kommunikationsmodelle (kurz: Modelle) beschreiben das Prinzip einer Kommunikationsbeziehung. Sie geben an, welche Rolle die beiden Kommunikationspartner beim Datenaustausch spielen.

Im Kapitel wird beschrieben, welche Kommunikationsmodelle und welche Begriffe im vorliegenden Dokument verwendet werden.

64.2.1 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.

64.2.2 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, die anderen Kommunikationspartner sind Slaves. 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 Schnittstelle

Master / Master Kommunikation ist nicht möglich.

64.2.3 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.

64.3 Quittierung

Werden Daten übertragen, dann gibt es unterschiedliche Rückmeldungen (Quittungen) an das STEP 7 Anwenderprogramm.

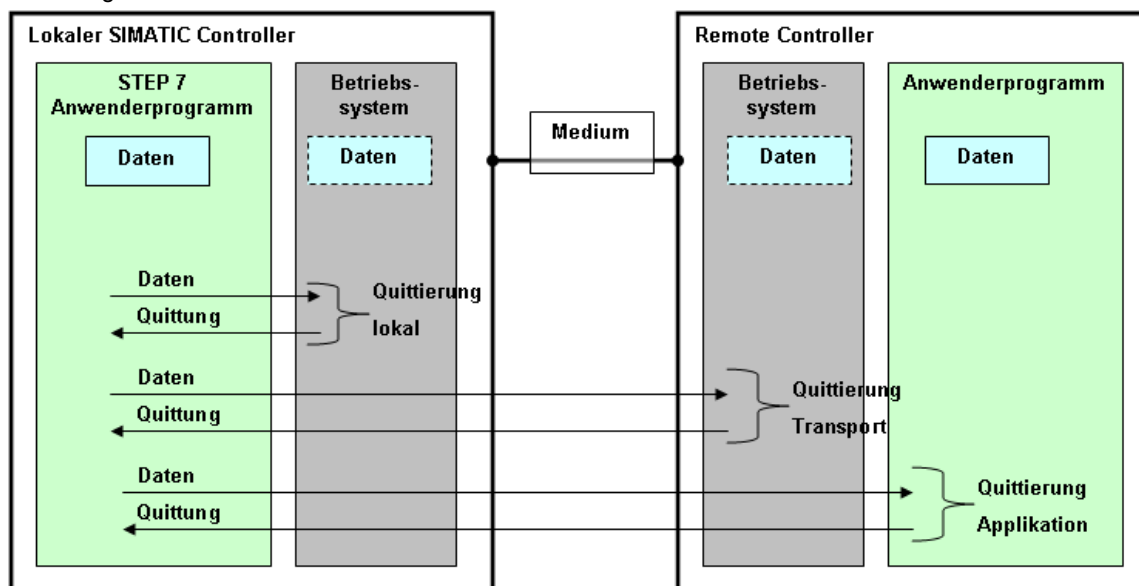
Im Dokument wird zwischen folgenden Quittierungen unterschieden:

Tabelle 64-2

Quittierung	Bedeutung	Bemerkungen
lokal	die Daten liegen im Betriebssystem (CPU, CP, CM) des lokalen Controllers	Keine Aussage, ob die Daten über das Medium (PROFINET/IE, ...) übertragen wurden.
Transport	die Daten liegen im Betriebssystem (CPU, CP, CM) 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 64-1



65 Betrachtete Komponenten

Hier werden die Komponenten aufgelistet, die im Dokument berücksichtigt wurden (Stand März 2013).

65.1 SIMATIC CPU

Tabelle 65-1

Familie	CPU		MLFB	Version
ET 200 CPU	ET 200S	IM151-8(F) PN/DP CPU	6ES7 151-8AB01-0AB0 (6ES7 151-8FB01-0AB0)	FW V3.2
		IM151-7(F) CPU	6ES7 151-7AA21-0AB0 (6ES7 151-7FA21-0AB0)	FW V3.3
	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 V3.3
	CPU 314		6ES7 314-1AG14-0AB0	FW V3.3
	CPU 312C		6ES7 312-5BF04-0AB0	FW V3.3
	CPU 313C		6ES7 313-5BG04-0AB0	FW V3.3
	CPU 313C-2 DP		6ES7 313-6CG04-0AB0	FW V3.3
	CPU 314C-2 DP		6ES7 314-6CH04-0AB0	FW V3.3
	CPU 313C-2 PtP		6ES7 313-6BG04-0AB0	FW V3.3
	CPU 314C-2 PtP		6ES7 314-6BH04-0AB0	FW V3.3
	CPU 314C-2 PN/DP		6ES7 314-6EH04-0AB0	FW V3.3
	CPU 315(F)-2 DP		6ES7 315-2AH14-0AB0 (6ES7 315-6FF04-0AB0)	FW V3.3
	CPU 317(F)-2 DP		6ES7 317-2AK14-0AB0 (6ES7 317-6FF04-0AB0)	FW V3.3
	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 (MPI/DP)		6ES7 412-1XJ05-0AB0	FW V5.3
	CPU 412-2 (MPI/DP, DP)		6ES7 412-2XJ05-0AB0	FW V5.3
	CPU 412-2 PN (MPI/DP, PN)		6ES7412-2EK06-0AB0	FW V6.0
	CPU 414-2 (MPI/DP, DP)		6ES7 414-2XK05-0AB0	FW V5.3
	CPU 414-3 (MPI/DP, DP, IFM)		6ES7 414-3XM05-0AB0	FW V5.3
	CPU 414(F)-3 PN/DP (MPI/DP, PN, IFM)		6ES7 414-3EM06-0AB0 (6ES7 414-3FM06-0AB0)	FW V6.0
	CPU 416(F)-2 (MPI/DP, DP)		6ES7 416-2XN05-0AB0 (6ES7 416-2FN05-0AB0)	FW V5.3
	CPU 416-3 MPI/DP, DP, IFM)		6ES7 416-3XR05-0AB0	FW V5.3
	CPU 416(F)-3 PN/DP (MPI/DP, PN, IFM)		6ES7 416-3ES06-0AB0 (6ES7 416-3FS06-0AB0)	FW V6.0
	CPU 417-4 (MPI/DP, DP, IFM, IFM)		6ES7 417-4XT05-0AB0	FW V5.3

Familie	CPU	MLFB	Version
S7-1200	CPU 1211C	6ES7211-xx31-0XB0	FW V3.0
	CPU 1212C	6ES7212-xx31-0XB0	FW V3.0
	CPU 1214C	6ES7214-xx31-0XB0	FW V3.0
	CPU1215C	6ES7215-xx31-0XB0	FW V3.0
S7-1500	CPU 1511-1 PN	6ES7511-1AK00-0AB0	FW V1.0
	CPU 1513-1 PN	6ES7513-1AL00-0AB0	FW V1.0
	CPU 1516-3 PN/DP	6ES7516-1AN00-0AB0	FW V1.0
S7-mEC (*1)	EC31-RTX (F)	6ES7677-1DD10-0BB0 (6ES7677-1FD10-0FB0)	08/2010
Box PC (*1)	SIMATIC Embedded Bundles (mit RTX):	IPC2x7D, IPC4x7C	
Panel PC (*1)	SIMATIC Embedded Bundles (mit RTX):	IPC277D, IPC477C	
WinAC RTX	WinAC RTX (F) 2010 (Software)	6ES7 671-0RC08-0YA0 (6ES7 671-1RC08-0YA0)	V4.6

Zu (*1): mit WinAC RTX (F) 2010 als Software Controller

65.2 SIMATIC CP oder CM

65.2.1 Einsatz in Stationen mit CPU

Die Tabelle enthält alle CPs und CMs, die im Teil 2 (Auswahlhilfe) des Dokumentes betrachtet werden.

Tabelle 65-2

Familie	CP oder CM			MLFB	Version
ET 200 CPU	PB	ET 200S	DP Mastermodul	6ES7138-4HA00-0AB0	FW V1.0
	PtP (*1)	ET 200S	1SI 3964/*ASCII*	6ES7138-4DF01-0AB0	FW V1.4
	PtP (*1)	ET 200S	1SI Modbus/USS	6ES7138-4DF11-0AB0	FW V1.4
S7-300	PB	CP 342-5		6GK7 342-5DA03-0XE0	FW V6.0
	PB	CP 342-5 FO		6GK7342-5DF00-0XE0	FW V5.7
	PB	CP 343-5		6GK7 343-5FA01-0XE0	FW V4.2
	PN/IE	CP 343-1 Lean		6GK7 343-1CX10-0XE0	FW V3.0
	PN/IE	CP 343-1		6GK7 343-1EX30-0XE0	FW V3.0
	PN/IE	CP 343-1 Advanced		6GK7 343-1GX31-0XE0	FW V3.0
	PN/IE	CP 343-1 ERPC		6GK7343-1FX00-0XE0	FW V1.0
	PtP (*2)	CP 340		6ES7340-1xH02-0AE0	FW V1.0
	PtP (*2)	CP 341		6ES7341-1xH02-0AE0	FW V2.1
S7-400	PB	CP 443-5 Basic		6GK7 443-5FX02-0XE0	FW V4.0
	PB	CP 443-5 Extended		6GK7 443-5DX05-0XE0	FW V7.0
	PN/IE	CP 443-1		6GK7 443-1EX30-0XE0	FW V3.1
	PN/IE	CP 443-1 Advanced		6GK7 443-1GX30-0XE0	FW V3.1
	PtP	CP 440		6ES7440-1CS00-0YE0	FW V1.0
	PtP	CP 441-1		6ES7441-1AA05-0AE0	FW V2.0
	PtP	CP 441-2		6ES7441-2AA05-0AE0	FW V2.0
S7-1200	PB	CM 1242-5 PB Slave		6GK7242-5DX30-0XE0	FW V1.0
	PB	CM 1243-5 PB Master		6GK7243-5DX30-0XE0	FW V1.2
	PtP	CM 1241 RS422/485		6ES7241-1CH31-0XB0	FW V1.0
	PtP	CM 1241 RS232		6ES7241-1AH30-0XB0	FW V1.0
	PtP	CB 1241 RS485		6ES7241-1CH30-1XB0	FW V1.0
S7-1500	PtP (*3)	CM PtP RS232 BA		6ES7 540-1AD00-0AA0	FWV1.0
	PtP (*3)	CM PtP RS232 HF		6ES7 541-1AD00-0AB0	FWV1.0
	PtP (*3)	CM PtP RS422/485 BA		6ES7 540-1AB00-0AA0	FWV1.0
	PtP (*3)	CM PtP RS422/485 HF		6ES7 541-1AB00-0AB0	FWV1.0
	IE	CP 1543-1		6GK7 543-1AX00-0XE0	FWV1.0
	PB	CM 1542-5		6GK7 542-5DX00-0XE0	FWV1.0
S7-mEC	PB	EM PCI-104		6ES7677-1DD60-1AA0	FWV1.0
	PN/IE	EM PC		6ES7677-1DD50-2AA0	FWv1.0
	PtP	CP 340		6ES7340-1xH02-0AE0	FWV1.0

Erläuterungen zu (*x): Siehe folgendes Kapitel 65.2.2.

65.2.2 Einsatz in dezentralen Stationen ohne CPU

CPs oder CMs mit serieller Schnittstelle können auch in dezentralen Stationen betrieben werden, in denen keine CPU gesteckt ist.

CPs oder CMs aus der Tabelle im Kapitel 65.2.1

(*1): auch in dezentraler Station ET 200S

(*2): auch in dezentraler Station ET 200M

(*3): auch in dezentraler Station ET 200MP

Zusätzliche Möglichkeiten

Neben den oben aufgeführten CPs oder CMs gibt es noch folgende Möglichkeiten:

Tabelle 65-3

dezentrale Station	CM	MLFB	Version
ET 200SP	CM PtP	6ES7137-6AA00-0AB0	FW V1.0

Hinweis:

Der CM PtP beherrscht die Protokolle: 3964(R), Freeport, USS, Modbus RTU.

Der CM PtP für ET 200SP hat die gleiche Funktionalität wie der CM PtP für S7-1500 (zentral) bzw. ET 200MP (dezentral).

66 Historie

66.1 Versionen

Tabelle 66-1

Version	Datum	
V2.1	04 / 2013	Update um neue SIMATIC CPUs
V2.01	01 / 2011	Fehlerbeseitigung
V2.0	11 / 2010	Komplette Überarbeitung
V1.0	04 / 2004	Erste Ausgabe

66.2 Wesentliche Änderungen

Tabelle 66-2

Version	Änderung
V2.01 -> V2.1	Aktualisierung aller Daten
	S7-1200: neue Baugruppen
	S7-1500: neues System
	Zusammenfassung der SIMATIC Familien: WinAC = WinAC + S7-mEC + Panel PC + Box PC
	Multipanel mit WinAC MP 2008: Gelöscht, da Auslaufprodukt
	Unterscheidung, wo erforderlich: STEP 7 (nicht TIA) / STEP 7 (TIA)
V2.0 -> V2.01	Seite 368: Absatz gelöscht: SIMATIC S7-1200
	Seite 398: SIMATIC CPU ergänzt: IM151-7 F CPU
V1.0 -> V2.0	Aktualisierung mit neuen Komponenten
	Neue Struktur des Dokumentes