

Inhaltsverzeichnis

Einführung und Übersicht	V
Vorwort.....	V
Vorwort zur 2. Auflage	VII
Einleitung.....	VIII
Kapitelübersicht – Beschreibung	XII
Kapitel 1 Einführung – Übersicht über die Programmiersprachen und die entsprechenden Hersteller	XII
Kapitel 2 Variablen in der Automatisierung – Basis einer Automatisierungslösung.....	XIII
Kapitel 3 Die Sprachelemente der ST/SCL-Programmierung im Detail ..	XIV
Kapitel 4 Erstellung von eigenen Funktionen und Funktionsbausteinen .	XIV
Kapitel 5 Anwendung von Funktionen und Funktionsbausteinen	XIV
Kapitel 6 Komplexere Daten und Datenstrukturen	XIV
Kapitel 7 Visualisierungen (Basic, „C“, VISU/3S, KTP 700).....	XIV
Kapitel 8 Revolution Pi – Ein modernes Automatisierungssystem – „Linux-basiert“	XV
Kapitel 9 ST in der „Safety-Welt“ (auf Basis der PSS 4000 der Fa. Pilz)....	XV
Kapitel 10 Übungen und Anwendungen	XV
Kapitel 11 Programmierung und Ausbildung mit Codesys 3.X und Raspberry Pi	XV
Trial-Versionen für das TIA Portal	XVI
1 Sprachen in der Automation	1
1.1 Software	1
1.1.1 Skript-Sprachen – Einführung	1
1.1.2 Programmiersprachen – Überblick.....	1
1.1.2.1 Basic	2
1.1.2.2 „C“	2
1.1.2.3 ST	3
1.1.2.4 SCL	3
1.2 Programmiersysteme – Anbieter	4
1.2.1 „Basic“	5
1.2.2 „C-Beispiel“: C-Control von Conrad Electronic	5
1.2.3 Strukturierter Text ST	6
1.2.3.1 Codesys – 3S (Kempton) ST	6

1.2.3.2	Multiprog – Phoenix Contact Software (Lemgo) ST	6
1.2.3.3	logi.CAD 3 von logi.cals (St. Pölten) ST	6
1.2.3.4	Siemens TIA Portal V13 SCL (Structured Control Language)	7
1.2.3.5	PAS4000 von Pilz	8
1.3	Anbieter von Basic-basierten Systemen	8
1.3.1	BasicMaster/Motron	9
1.3.2	C-Control von Conrad Electronic	9
1.3.3	ST auf Linux-basierten Steuerungen	11
1.3.3.1	Das I/O-Interface des Raspberry Pi	11
1.3.3.2	Das I/O-Modul vom Software-Anbieter logi.cals	12
1.3.3.3	Phoenix Contact Software/Lemgo und der Raspberry Pi	13
1.4	Steuerungshersteller – IEC 61131-3	14
1.4.1	Traditionelle Steuerungen	15
1.4.1.1	ABB-Steuerung	15
1.4.1.2	Eaton – easyControl	15
1.4.1.3	Phoenix Contact	17
1.4.2	S7-1200 von Siemens	19
1.4.3	Bediengeräte mit SPS-Funktion	20
1.4.4	Revolution Pi	20
1.5	Die Universal-Steuerung des Unternehmens Pilz „PSS 4000“	21
1.5.1	Phoenix Contact Software „Safeprog 3.00“	22
2	Variablen in der Datenverarbeitung – Automatisierung	25
2.1	Warum mit Variablen-Bezeichnungen arbeiten?	25
2.2	Die getesteten IDEs (Programmiersysteme)	26
2.3	Die Bekanntgabe von Variablen (Bezeichnern) nach IEC 61131-3	27
2.3.1	Variablen – Eingabe/Editoren	28
2.3.1.1	Conrad Electronic – C-Control-Software (IDE)	29
2.3.1.2	Codesys 2.3 – Variablendeklaration	29
2.3.1.3	Phoenix Contact – Variablendeklaration	30
2.3.1.4	TIA Portal V13 – Variablendeklaration	31
2.3.1.5	logi.CAD 3 – Variablendeklaration	32
2.3.1.6	PAS4000 von Pilz – der freie Editor wird angewendet	33
2.3.2	Variablenamen – Schlüsselworte	33
2.3.3	Arten von Variablen	33
2.3.3.1	Lokale Variable	33
2.3.3.2	Globale Variable	34
2.3.4	Die Variablen-Typen (Standard)	35
2.3.4.1	FS-Datentypen (FS = Fail SAFE)	36

2.4	Variablen – unterschiedliche Typen	38
2.4.1	Wandeln von Variablen – Typen-Konvertierungen (Funktionen)	39
2.4.2	Aufbau eines Variablen-Namens (Bezeichner).....	40
2.4.3	Wo kann der Anwender Variablen bekannt geben?	40
2.4.3.1	Globale Variablen unter Codesys 2.3	41
2.4.3.2	Globale Variablen unter PC Worx Express	42
2.4.3.3	Globale Variablen unter TIA Portal V13	42
2.4.3.4	Lokale Variable.....	43
2.4.3.5	IN-, OUT- und IN/OUT-Variablen.....	44
2.4.4	Variablen – Erhalt des Inhalts einer Variablen.....	45
2.4.4.1	RETAIN	46
2.4.4.2	PERSISTENT	47
2.4.4.3	RETAIN und PERSISTENT	47
2.4.4.4	Physikalische Adressen.....	47
2.4.4.5	Konstante – Literale.....	48
2.4.4.6	Kommentare	49
2.5	Arrays	50
2.5.1	Anwendung eines Arrays	51
2.5.2	Deklaration eines Arrays	51
2.5.2.1	Arrays unter Codesys 2.3	52
2.5.2.2	Arrays unter PC Worx	52
2.5.2.3	Arrays unter Mutiprogramm (PC Worx)	52
2.5.2.4	Arrays unter S7 (S7-1200) bzw. TIA Portal.....	53
2.5.3	Zugriff auf die Datenelemente eines Arrays	53
2.5.4	Initialisierung eines Arrays	54
2.5.4.1	Initialisierung eines Arrays (Codesys 2.3).....	54
2.5.4.2	Initialisierung eines Arrays (TIA Portal)	54
2.5.4.3	Initialisierung eines Arrays (PC Worx)	55
2.5.5	Zweidimensionales Array	55
2.5.6	Bekanntgabe Array – Zweidimensional – Typ: String	56
2.5.6.1	Codesys 2.3 – Variablendeklaration – Sprachumschaltung.....	57
2.5.6.2	Zweidimensionales Array unter PC Worx.....	58
2.5.6.3	Zweidimensionales Array bei der S7-1200	59
2.6	Strukturen (STRUCT)	60
2.6.1	Datentyp „Struct“ bei Codesys 2.3	61
2.6.2	Datentyp „Struct“ bei PC Worx	61
2.6.3	Datentyp „Struct“ bei der S7-1200	63
2.7	Weitere Beispiele für Arrays und Strukturen	63
2.7.1	Ampel	63

2.7.2	Sensoren	63
2.7.3	Käserei – Klimabahnen	64
2.8	Instanziierung, Instanzen bilden	64
2.8.1	Beispiele für Instanziierungen	64
2.8.1.1	Instanziierung eines Arrays bei PC Worx	65
2.8.1.2	Beispiel aus der Übung Kapitel_10_8 zu S7-1200 und TIA Portal	65
3	Die Sprache ST bzw. SCL im Detail	67
3.1	Unterschiede in den einzelnen Sprachen ST/SCL	67
3.1.1	Unterschiede in der Variablendarstellung	67
3.1.2	Unterschiede in der Programmierung	68
3.1.2.1	PC Worx – Ansprache eines Arrays (zweidimensional)	68
3.1.2.2	Unterschiede bei der Bekanntgabe von Startwerten (Initialisierung)	69
3.1.2.3	PC Worx – Unterschiede Bekanntgabe von Arrays oder Strukturen	69
3.1.2.4	PC Worx – Unterschiede bei Zuweisungs- und Testaktionen ...	69
3.1.2.5	Ansprache von Bit in Byte – WORD,	69
3.1.2.6	Unterschiede bei den Funktionen	70
3.1.2.7	Zusätzliche Möglichkeiten der Programmsteuerung mit SCL: GOTO	71
3.1.2.8	Unterschiede bei den CASE-Anweisungen	71
3.2	Mittel für den grafischen Vorentwurf	71
3.2.1	Struktogramme (bekannt unter Nassi-Schneidermann-Diagramm)	71
3.2.2	Flussdiagramme (ProgrammablaufPläne – PAP)	72
3.2.3	Ablaufsprache	72
3.3	Die einfache Anwendung – Zuweisung	74
3.3.1	Einfache Zuweisung	74
3.3.2	Mehrere Netzwerke in einer Zeile	74
3.3.3	Negation (NOT)	75
3.3.4	Klammern	75
3.3.5	Berechnung	75
3.3.6	Arbeiten mit Klammern	75
3.3.7	Abarbeitung durch den Prozessor	75
3.3.7.1	Bindungsregeln (Codesys 2.3)	76
3.3.7.2	Prioritäten (PC Worx)	76
3.3.7.3	Auswertungsreihenfolge (TIA Portal)	77
3.3.7.4	Beispiele zur Abarbeitung der Operatoren im TIA Portal	78

3.3.8	SET/RESET im Strukturierten Text.....	78
3.3.9	Aufruf eines Timers (PC Worx).....	79
3.3.10	Aufruf eines Timers – Vergleich	80
3.4	Boole'sche Logik.....	81
3.4.1	Klammern	82
3.4.2	NOT – Negation.....	82
3.4.3	AND – Logisches UND	83
3.4.4	OR – Logisches ODER.....	83
3.4.5	XOR.....	83
3.4.6	Übertragung von BIT an BYTE oder WORD bzw. DWORD....	84
3.5	Bit-Schiebe-Funktion	85
3.5.1	SHL(Anz, Element) – Schiebe links um Anz – Bit	86
3.5.1.1	SHL – Anwendung in einer Schrittkette	86
3.5.2	SHR(Anz, Element) – Schiebe rechts um Anz – Bit	87
3.5.3	ROL(Anz, Element) – Schiebe links um Anz – Bit – Rotierend ..	87
3.5.4	ROR(Anz, Element) – Schiebe rechts um Anz – Bit – Rotierend ..	88
3.5.5	BCD_TO_INT und INT_TO_BCD	88
3.6	Begrenzungen – Überwachungen	88
3.6.1	MAX(i_Wert1, i_Wert2)	89
3.6.2	MIN(i_Wert1, i_Wert2).....	89
3.6.3	Limit(MN,IN,MX).....	89
3.6.3.1	Beispiel – Anwendung „LIMIT“	90
3.6.4	MUX Auswahl – Funktion	91
3.6.4.1	Programm für SCL → S7-1200	91
3.6.4.2	Codesys 2.3	91
3.7	Zeit – Uhrzeit	92
3.7.1	Lesen Datum/Zeit mit Codesys 2.3.....	92
3.7.2	Lesen Datum/Zeit mit PC Worx	93
3.7.3	Lesen Datum/Zeit mit dem TIA Portal und der S7-1200	93
3.8	Mathematik.....	95
3.8.1	Potenzierung.....	96
3.8.2	Multiplikation „ * “	96
3.8.3	DIVISION „ / “	96
3.8.4	MOD – Modulo – Bestimmung des Rests.....	97
3.8.5	ADDITION „ + “ und SUBTRAKTON „ - “	97
3.9	Test-Operatoren.....	97
3.9.1	Test „< , > , <= , >=“	98
3.9.2	Test auf Gleichheit „ = “	98
3.9.3	Test auf Ungleichheit „<>“	98

3.9.4	Anwendung der Testoperatoren.....	99
3.10	Arbeiten – Bearbeiten eines Strings	99
3.10.1	ASCII-Zeichen	99
3.10.2	Sonderzeichen – Steuerzeichen	100
3.10.2.1	Array of String bei PC Worx.....	100
3.10.3	String-Bearbeitung	101
3.10.3.1	LEN(Str_name)	102
3.10.3.2	LEFT(Str_name,int_Anzahl)	102
3.10.3.3	RIGHT(Str_name,int_Anzahl).....	103
3.10.3.4	MID(str_name,usi_Anzahl,usi_Pos)	103
3.10.3.5	CONCAT(str_Text1,str_Text2)	104
3.10.3.6	INSERT(str_Text1,Str_Text2).....	104
3.10.3.7	DELETE(str_Text,usi_Anz,usi_Pos).....	104
3.10.3.8	REPLACE(str_Text1,str_Test2,usi_Anzahl,usi_Pos)	105
3.10.3.9	FIND(str_TextQuelle,str_TextSuch).....	105
3.10.3.10	INT_TO_STRING.....	105
3.10.3.11	STRING_TO_INT.....	106
3.11	Zugriff auf ein Array – Arbeiten mit Zeigern (Pointern)	107
3.11.1	Definition eines Arrays, des Zugriffzeigers und der Überwachungsgrenzen.....	107
3.12	Aufruf einer Funktion oder eines Funktionsbausteins.....	108
3.12.1	Aufruf eines Zählers	109
3.12.1.1	1. Schritt: die Instanziierung.....	109
3.12.1.2	Instanziierung beim TIA Portal (Datenbaustein)	109
3.12.1.3	Anwendung im Programm bei Codesys 2.3	110
3.13	Die Kontroll- und Schleifenanweisungen des Strukturierten Texts	112
3.13.1	IF...THEN.....	112
3.13.2	CASE-Anweisung, Fall-Entscheidung oder Auswahlanweisung	113
3.13.2.1	Was geschieht, wenn die Bedingung 2 + mal erfüllt ist?	114
3.13.3	WHILE...DO – kopfgesteuert.....	117
3.13.4	REPEAT...UNTIL – fußgesteuert	117
3.13.5	FOR...TO...DO – kopfgesteuert	119
3.13.6	FOR...TO...BY...DO – kopfgesteuert	119
3.13.7	EXIT-Anweisung (Abbruch).....	120
3.13.8	RETURN	121
3.13.9	CONTINUE – Rücksprung zur Schleifenbedingung	121
3.13.10	SCL – GOTO.....	122
3.13.11	Mischen der Kontroll- und Schleifenstrukturen	122

3.14	Die Programmieroberfläche – Programmierung – Besonderheiten.	123
3.14.1	Arbeiten mit einer Task	123
3.14.1.1	Task-Verarbeitung der einzelnen Systeme.	123
3.14.1.2	Allgemeine Übersicht möglicher Task – herstellerunabhängig .	126
4	Erstellung eigener Funktionen bzw. Funktionsbausteine	129
4.1	Programme – Funktionen – Funktionsbausteine (POEs)	131
4.1.1	Programme	132
4.1.1.1	Programme im Codesys 2.3	132
4.1.1.2	Programme im PC Worx – ILC-Steuerungen.	133
4.1.1.3	Programme in der S7-Welt	133
4.1.1.4	Programme in logi.CAD 3.	135
4.1.2	Funktion oder Funktionsbaustein?	136
4.2	Erstellen von Funktionen.	136
4.2.1	Funktionen unter Codesys 2.3	137
4.2.2	Funktionen unter Multiprog (PC Worx)	137
4.2.3	Funktionen mit der S7-1200 unter dem TIA Portal	137
4.2.4	Funktionen unter logi.CAD 3.	137
4.3	Beispiele für Funktionen	140
4.3.1	Zählen, wie viele Leuchten (max. 4) eingeschaltet sind.	140
4.3.1.1	Lösung einer ähnlichen Aufgabe mit logi.CAD 3	140
4.3.2	Erweiterung des vorherigen Beispiels um eine Leistungsberechnung	142
4.3.3	FC_ Welches Bit ist gesetzt?	142
4.3.3.1	„FC_WBitPos_TO_USINT“ im TIA Portal	142
4.3.3.2	„FC_WBitPos_TO_USINT“ bei PC Worx	143
4.3.4	FC_ Übertragen eines Status-Words in ein Array of Bool.	144
4.3.4.1	PC-Worx-Lösung zur Aufgabe als Funktionsbaustein.	144
4.3.4.2	Lösung mit FOR...NEXT im TIA Portal	145
4.3.4.3	Lösung mit WHILE...DO bei Codesys 2.3.	147
4.3.5	FC_ Ausmaskieren einer Schrittnummer – Matrixschaltwerk. .	148
4.3.5.1	DECODER Matrix-Schaltwerk im TIA Portal	149
4.3.5.2	DECODER Matrix-Schaltwerk bei PC Worx	149
4.3.5.3	Codesys 2.3	150
4.3.6	Funktion mit Datenübergabe als Struktur bei Codesys 2.3.	151
4.3.7	Stromstoß-Relais als Funktion?	155
4.4	Funktionen – Zusammenfassung: Arbeitsweise bei tiefergehenden Kenntnissen.	155
4.4.1	Verstecken von Eingangsvariablen	156

4.5	Erstellen von Funktionsbausteinen	156
4.5.1	Bausteinerstellung und Software-Engineering	157
4.6	Stromstoß-Relais (SSR)	157
4.6.1	Der realisierte SSR-Baustein mit SCL im TIA Portal	158
4.6.2	Die Variablendeklaration – Programmierung in SCL	158
4.6.3	Kommentare – Dokumentation in SCL	159
4.6.3.1	Anwendung im SCL/ST	159
4.6.4	SSR mit SET/RESET bei PC Worx	160
4.6.4.1	Anwendung in FUP bei PC Worx	160
4.6.4.2	SSR mit RS – Anwendung im ST (Online) bei PC Worx	160
4.7	Steuerung einer Lichtanlage – Anwendung von FB	160
4.7.1	Entwurf	161
4.7.2	Definition der Daten	161
4.7.3	Programmierung	162
4.7.4	Anwendung im KOP/FUP	162
4.8	Trenderkennung	163
4.8.1	Trenderkennung mit SCL	163
4.8.1.1	Variablendeklaration mit SCL (KAP4_8TRE)	164
4.8.1.2	Der Programmcode mit SCL (KAP4_8TRE)	164
4.8.1.3	Die Anwendung im SCL (KAP4_8TRE)	165
4.9	Erweiterte Trenderkennung mit SCL (KAP4_8TRE)	165
4.9.1	Variablendeklaration	166
4.9.1.1	Globaler Datenspeicher für das Archiv	167
4.10	Erstellen eines Skalierungsbausteins	169
4.10.1	Anwendung des Skalierungsbausteins in der Funktionsbaustein-Darstellung	171
4.10.2	Anwendung des Skalierungsbausteins in der Kontaktplan-Darstellung	171
4.10.3	Aufruf des Skalierungsbausteins im Strukturierten Text	171
4.10.4	Einfaches Beispiel zur Skalierung mit PC Worx	172
4.10.4.1	Anwendung Skalierung im ST mit PC Worx	173
4.11	Baustein zur String-Verarbeitung	173
4.11.1	SCL-Lösung – Einfügen einer Variablen	174
4.11.1.1	Speicherung der Texte in globalem Datenbaustein	174
4.11.1.2	I/O-Interface bzw. Variablen des Bausteins	175
4.11.1.3	Programmcode in SCL	175
4.11.1.4	Anwendung des Bausteins nach Aufruf (ohne Arbeitsparameter)	175
4.11.2	Codesys-Lösung mit einer Visualisierung	176

4.11.3	Lösung mit PC Worx	176
4.11.3.1	Deklaration des Arrays mit PC Worx	177
4.11.3.2	Initialisierung der einzelnen Texte mit PC Worx	177
4.11.3.3	Programm des FB mit PC Worx	177
4.11.3.4	Anwendung des FB – String mit Variablen in ST mit PC Worx	178
4.11.4	Unterschiede TIA Portal, Codesys 2.3 und PC Worx	178
4.12	Funktionsbausteine zur Ablaufsprache	179
4.12.1	Hintergründe GRAFCET, SFC, AS	179
4.12.2	Hintergründe – Sprachelemente der Ablaufsprache	179
4.12.2.1	Arten der Verzweigung in einem GRAPH oder CHART	180
4.12.2.2	Ablaufsprache und Zykluszeit der Steuerung	182
4.12.2.3	Mindestanforderungen – AS Flags	183
4.12.3	Ablaufsprache über Schritt-Merker	185
4.12.4	Ablaufsprache – Nachbildung über Status-Word	186
4.12.4.1	Lösung Codesys 2.3 (Ausbau in Kapitel 10)	186
4.12.5	Nachbildung über Matrixschaltwerke	187
4.12.6	Nachbildung über Schiebebefehle	188
4.12.7	Nachbildung mittels CASE...OF	190
4.12.7.1	Aufbau eines Schritts	190
4.12.7.2	Realisierung der 3-Zonen-Technik im Codesys 2.3	191
4.12.7.3	Grundaufbau einer Schrittkette über ein CASE...OF	191
4.12.7.4	Überlagerte Funktionen Initialisierung/Einfrieren	191
4.12.7.5	Alternativverzweigung mit einer CASE...OF-Anweisung	193
4.12.7.6	Simultanverzweigung bzw. Simultanprozess	194
4.12.8	Nachbildung durch ein Bit-Array	194
4.12.9	Zufahrtssteuerung mit Codesys 2.3	195
4.12.9.1	Bekanntgabe der Variablen	196
4.12.9.2	Globale Variablen	196
4.12.9.3	Grundbetriebsarten	197
4.12.9.4	Zusammenfassung der Weiterschaltbedingungen	197
4.12.9.5	Die Weiterschaltung	198
4.12.9.6	Auscodierung Nockenleiste, Schrittnummer, Text	198
4.12.9.7	Steuerung der Timer	199
4.12.9.8	Zuweisung der Ausgänge	199
4.12.9.9	Der Baustein in der Anwendung	199
4.12.10	Schlussbetrachtungen „Schrittketten“	200

5	Anwendung von ST/SCL mit Funktionen und Funktionsbausteinen ..	201
5.1	Bausteinsprache für Logik, Funktionen und Funktionsbausteine.....	202
5.1.1	Einfacher Funktionsplan (FUP) mit Codesys 2.3	202
5.1.2	CFC – Continuous Function Chart.....	202
5.1.2.1	CFC – Beispiele für Codesys 2.3	203
5.1.2.2	CFC – Beispiel mit PC Worx	204
5.2	Aufruf eines Funktionsbausteins in der FUP-Sprache.....	204
5.3	Aufruf im Kontaktplan (KOP)	205
5.4	Aufruf im Strukturierten Text bzw. SCL	205
5.4.1	Anwendung von IEC-Bausteinen	205
5.4.1.1	Einfacher Aufruf mit Konstanten	206
5.4.1.2	Aufruf mit Variablen.....	206
5.4.1.3	Komplexerer Aufruf mit adressierten Feldern	206
5.4.1.4	Komplexerer Aufruf mit adressierten Feldern – abgesetzt	207
5.5	Anwendung selbst geschriebener Funktionsbausteine.....	207
5.6	Anwendung von ST in der Ablaufsprache (AS, SFC oder GRAFCET)	207
5.6.1	Die Ablaufsprache – Sondermerker	208
5.6.1.1	Schritt看ette hat weitergeschaltet (SFCTrans)	209
5.6.1.2	Schritt看ette eingefroren (SFCPause).....	209
5.6.1.3	Schrittbetrieb (SFCTip)	209
5.6.1.4	SFCTip, SFCTipMode.....	209
5.6.1.5	Initialisierung der Schritt看ette (SFCInit).....	209
5.6.1.6	Löschen der Schritt看ette (SFCReset)	210
5.6.1.7	Nachführen der Schritt看ette	210
5.6.1.8	Aktionen – Was ist zu beachten?	210
5.6.2	Regeln in der Ablaufsprache	211
5.6.3	Die lineare Schritt看ette.....	211
5.6.4	Ergänzung zu den bisherigen Schritt看etten-Aktionen	211
5.6.5	Weiterschaltbedingungen mit Codesys 2.3	212
5.6.5.1	Schritte (Aktionen)	213
5.6.5.2	Alternative Verzweigung	213
5.6.6	Simultane Verzweigung	214
5.6.7	Verbindung beider Verzweigungsarten	215
5.6.8	Sprünge.....	215
5.6.9	Beispiel (Minimal „Alternativ-Verzweigung“).	215
5.7	Ablaufsprache mit PC Worx	217
5.7.1	Übersicht einer Kette (Chart oder Graph).....	219
5.7.2	Details – Elemente des PC Worx SFC.....	221
5.7.3	Unterschiede PC Worx zu Codesys 2.3	222

5.7.3.1	Unterscheidung in der Programmierung der Transition	222
5.7.3.2	Unterschiedliche Aktionsbereiche	223
5.8	Hersteller-Bibliotheken	223
5.8.1	Anwendung der Hersteller-Bibliotheken	223
5.8.1.1	Hersteller-Bibliotheken (easyControl) mit Codesys 2.3	223
5.8.2	Wo findet der Anwender die Bibliothek?	225
5.9	Bibliothek – Anwender FC oder FB	226
5.9.1	Anwender-Bausteine in PC Worx	226
5.9.2	Anwender-Bausteine in Codesys 2.3	227
5.9.2.1	Export von Funktionsbausteinen oder Funktionen bei Codesys 2.3	228
5.9.2.2	Import/Export von globalen Variablen bei Codesys 2.3	228
5.9.2.3	Import/Export von Strukturen bei Codesys 2.3	228
5.9.3	Verwalten und Zugriffe auf selbst geschriebene Bausteine	228
5.10	Steuern der Visualisierung	229
5.11	OSCAT – Eine Bibliothek mit einem umfangreichen Angebot	229
6	Komplexere Daten und Datenstrukturen	231
6.1	Strukturierte Daten (SDT)	232
6.1.1	Arrays und Strukturen – Vergleich	233
6.2	Arrays von Strukturen	234
6.2.1	Programmtechnische Anwendung in einem Funktionsbaustein	235
6.2.1.1	Eine STRUCT wird bekannt gegeben	235
6.2.1.2	Globale Variable bilden die Förderbänder ab	236
6.2.1.3	Die Anwendung eines Funktionsbausteins mit Array von Strukturen	236
6.2.2	Strukturen bei Codesys 2.3	237
6.2.3	Strukturen bei PC Worx bzw. Multiprog	238
6.2.4	Strukturen bei der S7-1200	239
6.3	Strukturen – Beispiele	239
6.3.1	Beispiel Zufahrtseinrichtung mit Ampel	239
6.3.2	Beispiel Käserei	240
6.3.3	Beispiel Abfüllbetrieb	240
6.3.4	Pumpenmanagement	241
6.3.5	Wasserversorgung	242
6.4	Kommunikation	243
6.5	Moderne Strukturen im Zeitalter des Internet	244
6.5.1	Beispiel Wasserversorgung	244
6.6	Strukturen und Funktionen, Funktionsbausteine sowie Visualisierungen	245

7	Visualisierung	247
7.1	Visualisierung des 3S-Entwicklungssystems.	247
7.1.1	Probleme bei der Grundmaske-Inbetriebnahme.	248
7.1.2	Die System-Variablen (implizite Variablen)	250
7.1.2.1	„Implizite Variablen“ des VISU	253
7.1.2.2	„Implizite Variablen“ des SFC.	253
7.1.3	Objekte – Grundlagen	253
7.1.3.1	Rechteck.	254
7.1.3.2	Abgerundetes Rechteck	255
7.1.3.3	Weitere grafische Elemente	255
7.1.3.4	Visualisierung	255
7.1.3.5	Schaltfläche	256
7.1.3.6	wmf-Datei	257
7.1.3.7	Tabelle	258
7.1.3.8	Alarmtabelle	259
7.1.3.9	AktiveX-Element	259
7.1.3.10	Zeigerinstrument	260
7.1.3.11	Trend.	260
7.1.3.12	Balkenanzeige	262
7.1.3.13	Histogramm	262
7.1.4	Menü-Steuerung (Maskenumschaltung)	264
7.1.4.1	Die Grundmaske bzw. Startmaske.	264
7.1.4.2	Maskenumschaltung durch den Bediener	264
7.1.4.3	Maskenumschaltung durch die Steuerung	265
7.1.5	Verschiedene Anzeigen-/Eingabemöglichkeiten	266
7.1.5.1	Mögliche Tastaturen	267
7.2	Das KTP 700 (bzw. KTP 600) mit S7-1200 und dem TIA Portal.	268
7.2.1	Gestaltung eines Bildaufbaus	270
7.2.2	Bilder und Bildumschaltungen.	271
7.2.3	Die Eigenschaften der Objekte und deren Steuerbarkeit	274
7.2.4	Verbindung S7-1200 und KTP 600/700 (Software).	275
7.3	Visualisierung Funktionen und Funktionsbausteine bei Codesys 2.3	277
7.3.1	Bekanntgabe der Strukturen und Variablen	277
7.3.2	Die Funktion zum Lesen und Schreiben	278
7.3.3	VISU zur Eingabe und Anzeige der Matrix-Zeilen.	279
7.4	Beispiel Käserei/Transportbänder mit Codesys 2.3.	283
7.4.1.1	Schaltfunktion mit dem Element „Rechteck“	284
7.4.1.2	Das Element Zeiger	285
7.4.1.3	Dynamisierung des Fahrwagens	287

7.4.2	Betrachtungen zu den Regelungen der Klimabahnen	287
7.4.2.1	Klimabahn – Struktur	288
7.4.3.2	Klimabahn – Daten der Regler als „globale Variable“	289
7.4.3.3	Aktualisierung des Reglers über einzelne Funktionen	289
7.4.3.4	Aktualisierung der Regler über ST, Tasksystem mit einer Funktion	289
7.5	Wasserversorgung – Höhenstand eines Hochbehälters	290
7.6	Zusammenfassung	291
8	Linux-basierte Automation	293
8.1	Die Basis – der Raspberry Pi	294
8.1.1	Raspberry Pi-Trainer	294
8.2	Einige Linux-Hinweise	294
8.2.1	Anmeldung des Revolution Pi	295
8.2.1.1	SSH-Anmeldung	295
8.2.1.2	PiCtory-Anmeldung	295
8.2.2	Shutdown	295
8.2.3	Abfrage der eingestellten IP-Konfiguration	295
8.2.4	Sudo („substitute user do“)	296
8.3	Werkzeuge Putty, Advanced IP Scanner	296
8.3.1	Advanced IP Scanner	296
8.3.2	Putty	296
8.3.3	Hilfreiche Internet-Adressen	296
8.4	Zugriffe – Einstellungen als PC	297
8.5	Die Hardware des Revolution Pi	297
8.5.1	Die CPU (RevPi Core)	297
8.5.1.1	Neue CPUs sorgen für mehr Leistung	299
8.5.2	Verfügbare I/O-Ebene bzw. Pi-Module	300
8.5.2.1	RevPi DIO Modul: 14 Eingänge / 14 Ausgänge	300
8.5.2.2	RevPi DI Modul: 16 Eingänge	301
8.5.2.3	RevPi DO Modul: 16 Ausgänge	301
8.5.2.4	Kommunikationsmodule (RevPi Gates)	301
8.5.3	Die Verbindung der Module über PiBridg	302
8.6	Der integrierte Webserver	303
8.6.1	PiCtory	303
8.7	logi.CAD 3 – Aufbau des Editors (IDE)	304
8.7.1	Konfiguration des Zielsystems (hier Revolution Pi)	305
8.7.2	Kostenlose Variante	305
8.7.3	Die Globalen Variablen	305

8.7.3.1	Direkt deklarierte Variable	305
8.7.3.2	Globale Variable in einer Datei deklarieren	306
8.7.3.3	Lokale Konfiguration	307
8.7.3.4	Zielsystem-Konfiguration	307
8.8	logi.CAD 3 und Raspberry Pi	307
8.8.1	Variante 1 – Ein-/Ausgänge über GPIO	308
8.8.2	Variante 2 – Ein-/Ausgänge über I2C	308
8.8.3	Variante 3 – Ein-/Ausgänge über SPI	308
8.8.3.1	PiFace	308
8.8.3.2	minitrainer	308
8.9	SpiderControl	309
8.9.1	Der Webserver in der Anwendung	309
8.9.2	Sprachverwaltung	309
8.9.2.1	Eingabe der Sprachelemente bzw. Begriffe	309
8.9.2.2	Der Taster (Button)	310
8.9.2.3	Parametrierung	310
8.9.3	Organisation der Seiten/Masken	311
8.9.3.1	Definition einer neuen Seite	311
8.9.3.2	Festlegung der Startseite	311
8.9.3.3	Umschaltung/Aufruf der Seiten	311
8.9.4	Objekte	311
8.9.4.1	Lights – Leuchtmelder	311
8.9.4.2	Gauques – Anzeigen	312
8.9.4.3	Switches – Schalter	312
8.9.4.4	Text Fields – Textfelder	313
8.9.4.5	Rectangles – Rechtecke	313
8.9.4.6	Images – Bilder – Piktogramme	314
8.9.4.7	Buttons – Schalter – Steuerelemente	314
8.9.4.8	Taster	315
8.9.4.9	Events – Ereignisse	315
8.9.4.10	Technologie-Schemata	316
8.10	Das Zusammenspiel der Datenkommunikation	318
8.10.1	Lösung mittels Raspberry Pi und „minitrainer“	319
8.10.2	Lösung mittels Revolution Pi	319
8.11	Beispiele zur Anwendung	319
8.11.1	Ampel	319
8.11.1.1	Beispiel Aufzug-Kabinenanzeige	320

9	Strukturierter Text in FailSafe-Anwendungen	323
9.1	Einführung – Strukturierter Text in der Safety-Technik	323
9.2	PSS 4000 – Die Hardware	325
9.2.1	Die CPU (Kopfstation)	327
9.3	Die Entwicklungsumgebung PAS4000 (Pilz Automation Suite)	328
9.3.1	Einstellung des Netzwerks	329
9.3.2	Definition des Steuerungsaufbaus	330
9.3.3	Einstellung der Hardware	330
9.4	Programmierung	332
9.4.1	TASK bzw. TASK-Organisation	333
9.4.2	Variablen	334
9.4.2.1	Grundvariablen – Grundtypen	335
9.4.2.2	Globale Variablen	335
9.4.2.3	Abgeleitete Datentypen (DDT)	335
9.4.3	POUs neu erstellen und inbetriebnehmen	338
9.4.3.1	Variablendeklaration	338
9.4.3.2	Aufbau einer STL-POU/Erstellung des STL-CODE	340
9.4.4	Schützen einer POU	340
9.5	Anwendung verschiedener POU	341
9.5.1	Beispiel SSR	342
9.5.2	Inbetriebnahme – Online-Funktion	343
9.6	Hersteller-Bausteine/Abbild von Kompetenz und Erfahrung	344
9.6.1	Einbindung eines Hersteller-Bausteins in der STL-Sprache	345
9.6.2	Übersicht über die Herstellerbausteine	347
9.7	Ergänzungen	348
9.7.1	Notwendige Dokumentationen	348
9.8	Zusammenfassung – abschließende Gedanken	348
10	Übungen Strukturierter Text	349
10.1	Getestete Ausbildungssysteme	350
10.1.1	Arbeiten mit dem Ethernet	351
10.1.2	System von Phoenix Contact (ILC 131 Starterkit)	352
10.1.3	Mini-Trainer (S7-1200/TIA Portal V13) mit einem KTP 700 ..	352
10.1.4	Trainer von KAFTANmedia	353
10.1.5	easyControl (Codesys)	354
10.2	Einfache Anweisungen	354
10.2.1	Einfache Logik	354
10.2.1.1	Einfache Zuweisung	355
10.2.1.2	Aufruf von Funktionsbausteinen in ST/SCL (Beispiel S7-1200)	355

10.2.1.3	Umcodierung BCD → Dezimal.....	356
10.2.1.4	Strom-Stoß-Relais	358
10.2.2	Analog-Technik	358
10.2.2.1	Vergleiche.....	358
10.2.2.2	Codesys – Zuweisen eines Ausgangswerts	359
10.2.2.3	TIA Portal – Bedingtes Zuweisen eines Ausgangswerts.....	359
10.3	Lichtsteuerung – Steuerung von 3 getrennten Beleuchtungssträngen.....	359
10.3.1	Die fertigen Bausteine.....	360
10.3.2	Das Programm der Lichtsteuerung.....	360
10.3.3	Variablendeklaration zur Lichtsteuerung	361
10.3.3.1	Variablendeklaration – Codesys 2.3 Lichtsteuerung.....	361
10.3.3.2	Variablendeklaration – PC Worx Lichtsteuerung	362
10.3.3.3	Variablendeklaration – TIA Portal S7-1200 Lichtsteuerung.....	363
10.3.4	Kapitel 3B eine Funktion (FC_AnzahlBit).....	363
10.3.4.1	Realisierung mit PC WORX (FC_AnzahlBit)	364
10.3.4.2	Übung 10.3.4 mit der S7-1200 (FC_AnzahlBit)	365
10.3.5	Übung „Anzahl und Leistung mit Codesys“ wird mit dem Visu erweitert.....	366
10.4	Lichtsteuerung mit Visualisierung	371
10.4.1	Lösung 3S.....	371
10.4.1.1	Die entsprechenden Funktionsbausteine.....	372
10.4.1.2	Die Bildschirm-Maske mit den Elemente-Nummern	372
10.5	Analogverarbeitung	373
10.5.1	Lösung PC Worx Express.....	373
10.5.1.1	PC Worx – Variablendeklaration	374
10.5.1.2	PC Worx Skalierungsbaustein – Das Programm	374
10.5.2	Lösung TIA Portal V13	375
10.5.2.1	Variablendeklaration TIA Portal V13.....	375
10.5.2.2	Programm TIA Portal V13	376
10.5.3	Umrechnung von Maßeinheiten	376
10.6	Arbeiten mit ASCII-Zeichen (Codesys 2.3)	376
10.7	„ST“ in der Schritttechnik – Modell Ampel	380
10.7.1	Variablendeklaration.....	382
10.7.1.1	Globale Variablen.....	382
10.7.2	Das Programm	384
10.7.2.1	Hauptprogramm	384
10.7.2.2	Graph	385
10.7.2.3	Überlagerte Kontrolle des Graphen.....	386
10.7.3	Visualisierung.....	388

10.8	Ampelsteuerung (Lösung mit S7-1200)	389
10.8.1	Die Variablendeklaration	390
10.8.2	Ampelsteuerung – Programm im TIA Portal	390
10.9	Ampelsteuerung mit einem Bediengerät ergänzt	394
10.9.1	Vernetzung im TIA Portal	395
11	Raspberry Pi und Codesys 3.X	397
11.1	Vorbereitungen	397
11.2	Neuerungen bei Codesys 3.5 zu Codesys 2.3	398
11.3	Arbeiten mit den GPIOs	398
11.4	Die Trainer / Raspberry Pi	400
11.4.1	Raspberry-Pi-Trainer (SPI-basiert)	400
11.4.2	Raspberry-Pi-Trainer (I2C-basiert)	401
11.5	Aufbau eines Projekts	402
11.5.1	Die Hardware	402
11.5.2	Arbeiten mit dem Task-Manager	403
11.5.3	Variablen	404
11.5.4	Retain der Variablen	405
11.6	Online-Verbindung zum Raspberry Pi	405
11.6.1	Gezielter Zugriff auf einen Teilnehmer – Passwort	405
11.6.2	Einstellung für den Teilnehmer	406
11.7	Beispiele	407
11.7.1	Die einfache Zuweisung	407
11.7.2	Aufruf eines Baustein – komplett:	408
11.7.3	Aufruf eines Bausteins – im Programm verteilt	408
11.7.4	Beispiel: StromStossRelais mit SPI und Codesys 3.5	409
11.7.5	Online-Debugging – Einstellungen	409
Nachwort	411
Stichwortverzeichnis	413