Detailed Table of Contents

Preface 13

Chapter 1: Introduction to CODESYS 15
- Agenda 16
- Why CODESYS and IEC 61131-3? 16
- Company Portrait 18
- Steady Growth 18
- Dominant Market Share 18
- CODESYS Staff 19
- CODESYS in the Field 19
- Industry Standardization in the 1980s and 2000s 20
- Controls Standardization 20
- CODESYS Ecosystem 21
- Virtualized Automation 22
- CODESYS Development System 23
- Benefits of IEC 61131-3 23
- CODESYS Installer 24
- Plug-Ins and Profiles 25
- Caveat 25
- Quick Start – The Device Tree 26
- Starting a New Project 27
- CODESYS ControlWin 27
- Scanning for Devices 28
- Runtime Security Policy 29
- Logging In and Running 29
- Process Cycle – Implied While Loop 30
- Intro to Debugging 31
- Lab Materials 31
- Lab Notes 32
- LAB Overview 32
- LAB: Creating a Project 33
- LAB: Creating a Simple Program 33
- LAB: Connecting and Logging into the Control Win V3 35
- LAB: Running the Application 36

Chapter 2: Help and Settings 37
- Helpme-CODESYS.com (and Help->Contents) 37
- Search Operators 38
- CODESYS Store 38
- Caution Installing Packages 39
- CODESYS Forum 39
- CODESYS Clips on YouTube 40
- Support 40
- System Partners 41
- Tools -> Options 41
- Options -> Load and Save 42
- Options -> Smart Coding 43
- Customize Menus 43
- Project Settings 44
- Source Download 44
- PLC Settings and Properties 45
- Page Layout 46
- Helpful Hints 47

Chapter 3: Projects and the IDE 51
- The Devices View – The project roadmap 51
- Adding a Device 52
- Device Editor 53
- Programmable vs Parameterizable 55
- Programmable Objects 56
- Parameterizable Objects 57
- Adding a POU 58
- Programmable Object Management 59
- Folders 60
- Active Application 60
- Interacting with the Active Application 61
- Multiple Applications on One Device 61
- Online Change 62
- Online Mode 63
- Opening a Project in a Different Version of CODESYS 63
- Updating a Project 64
- Updating a Device 65
- Project Archives 65
- Saving a Project Archive 66
- Extracting a Project Archive 67
- Lab Overview 67
- Lab: Update a Project 68
- Lab: Extracting a Project Archive 69

Chapter 4: Variables and Data Types 71
- Variables and Data Types 71
- Auto Declare 72
- How to Declare a Variable 73
- Variable Names 74
- Data Types: Integer and Real 75
- Data Types: Time and Memory 76

Data Types in Modern CPUs 77
Data Types: Strings 78
String Literals and Functions 79
Type TIME and TIME() Function 80
SysTimeRTC: System Real Time Clock 81
Variable Scope 82
Scope: VAR (Local Private Variables) and Rules for Variable Access 83
Scope: VAR_INPUT (Local Input Variables) 84
Scope: VAR_OUTPUT (Local Output Variables) 85
Scope: VAR_IN_OUT 86
Scope: Global 87
Flags 88
Retain and Persistent Variables 89
Type Conversions 90
Display Radix 90
Refactoring vs Search/Replace 91
Refactoring POU Names 91
Arrays (preview) 92
Array Initialization (preview) 92
Declaring Instances of Function Blocks 93
Declaring a TON 93
LAB: Declaring Variables 94
LAB: Declaring Variables (continued) 95

Chapter 5: Ladder Logic and FBD 97
IEC 61131-3 and CODESYS Languages 97
The Right Tool for the Job 98
History of LD 98
Ladder Logic 101 99
LD for Boolean Logic 99
Function Block Diagram 100
FBD for Logic 100
Ladder Diagram (new with SP19) 101
Drag and Drop (or Select and Click) 101
Drag and Drop Networks, Contacts, and Coils (or Select and Click) 102
Branching, Moving, and Copying 102
Block, Jump, Return, and ST Block 103
Negate, Set/Reset, Rising/Falling Edge, EN/ENO 104
Comments, Conversion, and OutCommented 105
Original LD Editor 106
LD Editor Menus 106
Using the Toolbox 107
Ladder and Math 107
Drag and Drop from ToolBox and Declaration 108
Branches and SCE 108
Box vs Box with EN/ENO 109
Adding Comments 110
Close all Open Editors between Labs 110
LAB: Overview 110
Optional LAB: Create a Traffic Light in the original Ladder Logic (LD) 112
LAB: Extra Credit 113

Chapter 6: Sequential Function Chart 115
Sequential Function Chart (SFC) 115
What is a State Machine? 116
State Machine in SFC 117
How to add SFC Elements 117
SFC Step 118
SFC Transition and Transition Function 119
Step Action and Action Function 121
Behavior of Actions when a Step is Copied 123
Creating Actions and Transitions Manually 124
Jumps 124
Branches 125
Adding Alternate and Simultaneous Branches 126
Dangling Branches 127
Macros 128
SFC Setup, Properties, and Flags 129
More on SFC Implicit Variables 130
Hint: Off Diagram Global Operation 131
Display your SFC/LD2 on an HMI 132
Conditional Equations in Actions, and Alternatives to SFC 132
State Machine in CFC 133
Recommended Practice – Set all VAR_OUTPUT Variables on Every Entry Action 134
Hints for Approaching a State Machine Design 135
LAB: Implement the Traffic Light in SFC 136
LAB: Extra Credit 137
LAB: Extra Credit Answer 138
LAB: More Extra Credit 138

Chapter 7: Structured Text 139
Introduction to Structured Text 139
ST Statements: Assignments and Calls 140
Assignment Operators and Precedence 141
IF THEN ELSIF ELSE Conditional Statement 141
CASE Conditional Statement 142
FOR Iteration Statement 143
WHILE Iteration Statement 144
REPEAT Iteration Statement 145
Other Language Elements 145
ST Options and Features 146
Using Expressions in place of Variables 147
Enumerations – Putting a Name with a Number 147
Accessing FB Input and Output Variables 148
State Machine in SFC and ST 149
Alternative ST State Machine 150
LAB: Implement a Traffic Light in Structured Text 151

Extra Credit – Use a TON 152
Extra Credit 153
Extra Exercises 153

Chapter 8: Advanced Debugging 155
Simulation vs ControlWin 155
ControlWin on your Computer 156
Breakpoints 157
Data Breakpoints and Breakpoint View 158
Single Stepping and Call Stack 159
Watch List 160
Process Cycle – Online Variable Values 161
Watch List with Execution Points and Flow Control 162
Flow Control, continued 163
Force and Write 164
Browse Definition, Call Tree, Cross Reference 165
Warning: Cross Reference does not work well with Function Blocks 166
Project Compare 167
Comparing Device Tree Objects 168
Reset and Clean 169
Starting with a Clean Project 170
Device Log 171
Configuring Device Log File Storage 172
Open Project – Read Only 172
Project Information 173
Messages View 173
Message Pragmas 174
Exclude From Build 174
POU for Implicit Checks 175
Online Change 176
Operating Mode 176
Core Dump 177
Trace 177
WireShark for Debugging Network Issues 178
CODESYS Troubleshooting Continuum 178
Programming Suggestions to Optimize CPU Performance 179
LAB: Debugging 180

Chapter 9 : POUs: Programs, Functions, and Function Blocks 181
POUs 181
PROGRAMS 182
Functions 183
Standard Functions 184
Function Block 185
Comparison of Programs to Function Blocks 186
Function Block as a Data Type 187
Multiple Instances of a Function Block 188
Declaring Instance of Function Blocks 189
Comparison of POU Types 190
Language Agnostic 190
Input and Output Variables 191
Accessing FB Input and Output Variables 192
Accessing FB Input and Output Variables 193
Accessing FB Input and Output Variables in LD 193
Method: A Function that *Belongs* to a Function Block 194
Other Variable Scopes 195
Difference between Methods, Actions, Transitions, and Properties 196
Input Assistant and Auto Declare 197
Nine Ways to Initialize Variables 198
Cross Reference does not work well with Function Blocks 201
LAB: Converting a Program to a Function Block 202
LAB: Extracting Code to a Function 203
LAB: Extra Credit 204
LAB: Extra Credit 2 204

Chapter 10: Continuous Function Chart (CFC) 205
Continuous Function Chart (CFC) Overview 205
CFC for Hierarchical Design 206
CFC vs LD 207
CFC for Block-Based Design 208
CFC for the Trunk and Branches 209
CFC Basics 209
Editing Basics 210
Editing Pins and Blocks 212
Auto Data Flow 214
Using Methods to simplify your CFC 215
Page Oriented CFC 215
Lines Color-Coded by Data Type 215
Parameters in Function Block 216
Parameters Offline and Online 217
Input and Output Toolbox Elements 218
CFC Productivity Hints 218
Which IEC 61131-3 Language is Best? 219
Multi-Language Hierarchical Plant Design 220
LAB: Overview 220
LAB: Continuous Function Chart 221
LAB: Continuous Function Chart continued 222
LAB: CFC Parameters 223
LAB: Extra Credit 224

Chapter 11: Object-Oriented Industrial Programming 225
OOP vs OOIP 225
Why OOIP? 227
Evolution from Flat Programming 227
Task-Oriented vs Object-Oriented 228
Encapsulation, Abstraction, and Interfaces 229

Sample Analog Input Object 229
Analog Input Object used in Plant Design 230
Object of Objects 231
Example of Object-Oriented Plant Hierarchy 232
The Plant View 233
Control and Equipment View (C&E)..... 234
Control View 234
Motor View 234
Variable Naming Convention 235
Hierarchical vs Flat Design – Best Practice 236
IO and Configuration Mapping in OOIP 237
Hard-Coded Configuration and I/O Mapping 237
Object-Oriented Configuration and I/O Mapping 238
Sample of an Actual Configuration/Persistence CSV File..... 239
I/O Mapping as a Configuration..... 240
Future Enhancement 240
Plant-Level Simulation 241
Simulation Code..... 242
Simulation Model..... 243
Reusable Simulation Objects 244
Simulation Models Embedded in Control Code.. 246
Other Simulation Options..... 247
Simulation Results..... 248
Fully Automated Test..... 248
Other OOIP Resources 249
LAB: Object-Oriented Industrial Programming Examples 250
LAB: Extra Credits..... 250

Chapter 12: User Defined Types (DUT)..... 253
Arrays..... 253
Two or Three Dimension Arrays 254
Array of Array..... 255
Limiting the Display Range of an Array..... 255
Variable Length Arrays..... 256
VAR_GENERIC CONSTANT..... 257
Structures..... 258
Array of Structures..... 259
Structure Initialization 259
Structure Pins..... 260
Structures and Arrays in CFC 261
Enumerations – Putting a Name with a Number 262
Enumeration Numbering 263
Enumeration (aka Implicit Enumeration) 264
Enumeration Localization 265
Union..... 266
Union – Example (and Endian)..... 266
Union – Example 2 267
Union for Variant Data Type 268
Union and Structure for Fieldbus Mapping 269
Using a Function Block as a DUT 270
Alias 270
LAB: Replacing loose data with a Data Structure 271
LAB: Extra Credit..... 272

Chapter 13: Task Manager and Tasks.....273
Task Configuration Object and Overview..... 273
Task Assignment and Priority 274
Types of Tasks..... 274
Watchdog Timer and Sensitivity..... 275
Sensitivity Multiplier..... 276
Sensitivity with other Cycle Types..... 276
System Events..... 276
Recovery from a Watchdog Timeout 277
Task Monitoring..... 277
Multi-Tasking and Semaphores..... 278
Multithread Multicore..... 280
LAB: Create Fast and Slow Programs and Tasks. 282
LAB: Extra Credit – Event and Status Driven Tasks 283
LAB: Extra Credit – Create a System Event..... 283
LAB: Extra Credit – Protect code with a Semaphore 283

Chapter 14: Introduction to I/O and Fieldbuses 285
Categories of I/O 285
Local I/O: Plugging Devices..... 286
Local I/O – Configuration..... 287
Remote I/O – Step 1: Installing Device Description Files..... 288
Remote I/O – Step 2: Add Network, Protocol, and End Devices 289
Remote I/O – Step 3: Configure Network and End Devices..... 290
Remote I/O – Exponential Complexity without DD File 290
OPC Field Level Communications (FLC) 291
I/O Update..... 292
Online Config Mode..... 292
Object-Oriented I/O – Cognitive Leaps 293
I/O Mapping: Direct to I/O Port..... 294
I/O Mapping: Through a Global Variable List..... 294
I/O Mapping: Full Path from I/O to Program 295
I/O Mapping: Mapping as a Configuration Parameter..... 295
I/O Mapping: I/O FB communicate directly with I/O 297
OOIO In Practice 298
I/O Mapping: Future I/O Mapping 299
Object-Oriented I/O 299
Equipment Oriented I/O (EOIO) 300
Object-Oriented I/O Demo..... 301

LAB: Overview 302
LAB: Fieldbus Slave....302
LAB: Fieldbus Master....303
LAB: Connecting the Master and Slave 304
LAB: Extra Credit – OOIO Demo 305

Chapter 15: Libraries and Library Manager 307
Overview 307
Types of Libraries 308
Library Repository 309
Library Repository Locations....309
Library Manager Overview....309
Library Manager 310
Adding a Library to the Application....311
Download Missing Libraries 311
Library Properties....312
Placeholders....313
Available Libraries 314
UTIL Library 314
CODESYS Automation Alliance (CAA)....315
CODESYS Store 315
OSCAT Basic Library contents....316
IIoT Library....317
Other Common Libraries....317
PLCopen Common Behavioral Library (CBML)....317
Creating a Library 318
Library Information 319
Library Properties....320
Saving a Library 321
Customizing a Library with a Parameters List 322
Using Alias in a Library 323
Restricting Use of your Compiled Library....324
Signing a Library – Authenticate the Author....325
Documenting Libraries with restructured Text...325
Library Guidelines....326
Package Designer 326
LAB: Create a New Library containing your Traffic Light FB....327
LAB: Install and Use the new Library....328
LAB: Extra Credit....329

Chapter 16: Introduction to Visualization....331
Benefits of CODESYS Visualization 331
Visualization Hardware Options....332
Visualization Manager....333
Visualization Manager – Settings....333
Visualization Project Settings....334
Configuration and Properties....334
Visualization Editor....335
Visualization Toolbox 336
Common Input and Output Elements....337
Text Field – Value Output....338
Displaying Enumerated Text....338
Visualization Productivity 339
Fixed Properties....340
Programmable Properties....341
Rotation and Interior Rotation 342
Combining Rotation and Interior Rotation....342
Animation and Bring to Foreground....343
Input Configuration – Action on Mouse Events..344
Input Configuration – Input Values....345
Input Configuration – Change Pages 346
Input Configuration – Execute Command 346
Input Configuration – Changing Visualization in a Frame....347
Input Configuration – Execute ST Code and Multiple Sessions....348
Input Configuration – Transfer Files between Visualization and PLC....349
Input Configuration – Change Language with Text Lists and Dynamic Text 350
Input Configuration – Security and User Management....351
Input Configuration – Access Rights Visualization Elements 352
HTTPS WebVisu Security....352
Using Images in Visualization....353
Dynamic Images....354
Displaying Motion Pictures....354
Image Element Scaling Modes....355
WebVisu Scaling Modes....356
Code Interface (aka Declaration Area) 356
Local Variables and Visualization Automation ...357
Dialogs....358
Programmatically open a Dialog using VisuUtils 359
HTLM5 Elements in WebVisu – Building Elements 360
Traditional Visualization – Copy/Paste/Edit 362
Reusable Visualization Block....362
Frame Config – Make your VB look like an Element 363
Frame Config – Building your FC....364
Building a Visualization for a Function Block....365
Placing and Associating a Visualization Block....366
Placing a Visualization Block into a Frame....367
Placing and Associating Multiple Visualization Blocks....368
Single-Tag Plant-Level Connectivity....368
Object-Oriented Industrial Visualization 369
Visualization Consistency – Enhancement Request 370
VisuSymbols....372
Programmatic Visualization....372

Other Visualization Features....373
LAB: Create a Reusable Visualization Block....374
LAB: Instantiate a Visualization Block into a Visualization....375
LAB: Text Fields....376
Lab: Making a Pop-Up Dialog....377
LAB: Extra Credit....378

Chapter 17: OPC UA and Peer to Peer....379
OPC and CODESYS OPC UA Server....379
OPC UA Server using Symbol Configuration (SC) 380
SC: Selecting Variables....380
SC: Exposing by Attribute....381
SC: Supports DUTs....381
SC: Optional OPC UA Settings....382
Communications Manager (CM): OPC UA Server....382
CM: Symbol Type Editor....383
CM: OPC UA Information Model....384
CM: Symbol Sets....385
Data Source Manager – CODESYS Application (DSM/CA)....386
DSM/CA – Specifying Sources....387
DSM/CA – Specifying Sources....388
Data Source Manager With OPC UA Server (DSM/OUS)....389
DSM/OUS – Connecting to Client....389
DSM/OUS – Selecting Server Variables....390
DSM/OUS – Variables tab and "Create or Map" 391
DSM/OUS – Structured / CreateNew Example....392
DSM/OUS – UnStructured / MapToExisting Example....393
DMS/OUS – Manual Entry of Type and Var....393
OPC UA – DUT Transfer....394
DSM/OUS – Using a Local DUT....394
DSM/OUS – Type Mappings Tab and Create or Map....395
DSM/OUS – Mapping to a Local Data Type Example....396
DSM/OUS – Mapping to a Local Variable and Data Type Example....396
DSM/OUS – CreateToExisting TYPE....397
DSM/OUS – Transferring Arrays....398
DSM/OUS – Mapping Members of Structures or FBs....398
OPC UA Client....399
OPC UA CLIENT API....399
Object-Oriented OPC UA (OOOpcUa)....400
OOOpcUa – Quick Start....401
OOOpcUa – Implementation Options....402
OOOpcUa – PLC/OPC Synchronized Cycle....403
Network Variables....404
Network Variables – Sender....404
Network Variables – Receiver....406
Mapping Network Variables to/from Objects....407
IIoT Libraries....408
Other Peer to Peer Options....408
Comparison of Peer to Peer Sharing Options....409
LAB: OPC UA Server: Create a Symbol Configuration....410
LAB: OPC UA Server using the Communications Manager....411
LAB: Access OPC UA Variables with Data Source Manager....412
LAB: Extra Credit – Access OPC UA Variables with CODESYS OPC UA Client....413
LAB: Extra Credit – Access OPC UA Variables with UaExpert™....414
Lab: Extra Credit – Explore the OOOpcUa Demo 415
LAB: Extra Credit – DSM CODESYS Application (DSM/CA)....415
LAB: Extra Credit – Network Variables....415

Chapter 18: Security....417
CODESYS Security Overview....417
1. Project Permissions – Configure an Owner....418
1. Project Permissions – Create Users and Groups....418
1. Project Permissions – Objects....419
1. Project Permissions – Actions....419
1. Project Permissions – Logging on/off....420
2. Project File Encryption....420
3. IDE to Device Connection Encryption....421
4. Device Security....422
4. Device Security – OPC UA....422
4. Device Security – Creating an OPC UA Certificate....423
4. Device Security – OPC UA Sets....424
4. Device Security – Access Rights....424
LAB: Setup Users and Groups....425
LAB: Extra Credit....426

Chapter 19: Persistence and Recipes....427
Categories of Variables....427
Techniques to Manage Persistent Variables....428
Recipes: A Spreadsheet of Data....429
Adding a Recipe Manager and Recipes....430
Recipe Manager....430
Recipe Definitions....431
Adding Variables to a Recipe Definition....432
Adding Recipes....432
Recipe Commands....433
Programmatic Recipe Management....434
CODESYS Persistent Variables....435

Periodically Storing and Restoring Persistent Variables....436
Saving Persistent Variables on an Unhandled Exception....437
Persistent Variables –Impact on Function Blocks....438
Combining Persistent Variables with Recipes for Human-Readable Files....439
ControlSphere Parameter and Persistence (CPP) Library....440
ControlSphere CPP Library – Designed for Distributed Objects....441
CPP Library – Storing, Restoring, and File Format....441
CPP Library – How it Works....443
CPP Library – Future Enhancement....444
CPP Library – Actual Configuration/Persistence File....445
LAB: Create Recipes for Traffic Light Timing....446
LAB: Writing Recipes....447
LAB: Recipes Extra Credit....448
LAB: Extra Credit CODESYS Retain/Persistent....448
LAB: Working with the CPP Library – Configuring Objects....449
LAB: CPP Library – Saving and Loading Parameter Values....450
LAB: CPP Library – Adding Instances....451
LAB Extra Credit: CPP Library – Add Config File to the Application....452

Chapter 20: Trace and Trend....453
Introduction to Trace....453
Trace Basics....454
Adding a Trace....454
Trace Configuration....455
Configuring the X Axis....457
Configuring the Y axis (Adding Diagrams)....458
Adding Variables to the Y Axis....459
Starting the Trace....460
Trace Tools....461
Visualization Trace Element....462
Trend and Other Information....462
LAB: Building a Trace....463
LAB: Viewing a Trace and Setting a Trigger....464
LAB: Extra Credit – Add a Trace Vis....464

Chapter 21: Alarm Manager....465
Introduction to Alarm Manager....465
Alarm Groups and Classes....466
Alarm Group Overview....467
Alarm Group Details....467
Alarm Group – Additional Hints....469
Alarm Group Templates for FBs and Structures....470
Alarm Class Overview....471
Alarm Class Options....472
Alarm Storage....473
Alarm Visualization....474
Alarm Table – Additional Controls....474
Object-Oriented Alarming (OOA) – Registering Alarms....475
Object-Oriented Alarming (OOA) – Triggering Alarms....476
Other Information....476
Lab Overview:....476
LAB: Create Alarm Groups and an Event....477
LAB: Configure Classes and place Alarm Banner 478
LAB: Object-Oriented Alarming....479
LAB: Extra Credit....480

Chapter 22: Professional Developer Edition....481
CODESYS PDE Tool Suite....481
Git and SubVersion....481
GIT Views....482
Project Compare before Commit....482
Comparison to the Rung, Block, or Line of code. 483
CODESYS Static Analysis....483
Sampling of Static Analysis Rules....484
Static Analysis – Propagation of Constants....484
Static Analysis – Metrics and Complexity....485
Static Analysis – Sample Metrics Report....485
Profiler....486
Test Manager....487
PDE on the CODESYS Store....488
The CODESYS Store....488
CODESYS Store – Sample of Contents....489
Application Composer – Code Generator....490
Single-License (SL) Runtimes....491
Automation Server....491
Matlab™ Simulink™ Importer....492
Demo Projects....492
Access to Runtime File System (File I/O)....492
Matrix Library....493
Control Loop Library....493
Developing Libraries for the CODESYS Store....494
Security Keys and Packages....494

Chapter 23: Miscellaneous....495
Controller Handling....495
Multiple IDE Access to a Controller....495
Source Upload and Download....496
Import/Export....496
Generic Data Types – ANY_<type> (parametric polymorphism)....497
Online Change....498

Online Change Memory Reserve 499
Online Change Status Reporting 500
Steps to Allow a Future Online Change 500
Online Editing 501
External File 502
AND_THEN and OR_ELSE Short Circuit Operators 502
Pragmas 503
Conditional and Unconditional Pragmas 503
TRY/CATCH 504
Adding a Custom Project Template 504
Python Scripting 505
Text Lists Outside of Visualization 505
UTF-8 Support for Strings 506
Intel™ Time Coordinated Computing (TCC) 506
Unit Conversion 507
Switchable Conversion Sets 508
Unit Conversions in Visualizations 508
System of Measurement example 509
C Code Module 510
Installing Package – Where did it go??? 510
LAB: Extra Credit 510

Chapter 24: Advanced Topics and OOP 511
OOP vs OOIP 511
OOP: Inheritance 512
Difference between Composition and Inheritance 513
OOP: Methods 513
Methods – Pass Data and Initiate Action 514
Properties – Assign Values and Initiate Action 515
Access Specifiers and FINAL flag 516
Access to Parent Variables, Methods, and Properties 517
Polymorphism – Overriding Methods 518
More Polymorphism 519
Extended Structures in Extended Function Blocks 519
Extended FB – Traditional Scanned Approach 520
Extended FB – Event-Driven Approach 521
Alternate Technique using Composition 522
Alternative Universal technique 522
Base-Class Calls Derived-Class Methods 523
Pop Quiz – Inheritance and Polymorphism 523
Answers to Pop Quiz 524
VAR_IN_OUT, Pointers, and References 524
VAR_IN_OUT – Safe “under-the-cover” Pointers 525
Reference Variables – “under-the-cover” Pointers 526
Applications for Reference 528
Addresses and Pointers 529
Stored Pointers and Online Change 530
Advanced Pointers 531
Ref= and Pointer Assignments 532
Addresses of Reference and VAR_IN_OUT 534
Possible Future Pointer Enhancements 535
Summary of VAR_IN_OUT, REFERENCE, and POINTER 536
OOP: Interfaces 538
Interface Example – Day or Night 539
Using Interfaces to Auto-Register Instance of Objects 540
Interfaces in ControlSphere CPP Library 541
Interfaces: Online Change Safe Dependency Injection 542
__QUERYINTERFACE(<source>, <target>) 543
Visualization Blocks with common Base Class 544
Visualization Block with Interfaces 545
Programmatic Visualization References 546
Intro to Dynamic Memory 547
Compiling Function Block and DUT Instances 548
Initializing Dynamically Allocated Objects 549
A1: Static Array of Union of Function Blocks 550
B1: __NEW and __DELETE 551
B2-4: SysMem, MemoryUtils, and Memory Block Manager 552
LAB: Pointers 553
LAB: Extra Credit 554
LAB: Inheritance 555
LAB: Interface and Event-Driven Programming 556
Lab: Pure OOP Traffic Light Example 556

Chapter 25: Final Project 557
Final Project – Bottling Plant 557
Achievement Level Options 558
Bachelors Level – Bottler System Diagram 559
Bachelors Level – Hints 559
Bachelors Level Hints – Process Sequence 560
Bachelor Level Review – State Machine Design 560
Bachelors Level Review – OOIP Design 561
Bachelors Level Review – OOIP Example 561
Bachelors Level –Traditional PLC Programming Alternative 562
Bachelors with Honors Level – Visualization 562
Masters Level – Bottler System Diagram 563
Masters with Honors – Bottler System Diagram 563
Masters with Distinction – Plant with Multiple Bottlers 564
Masters with Distinction – Configure with CPP Library 564
Masters with Distinction – Add Settling Time to the CSV 565
Masters with Distinction – Supplemental 565
Masters with Distinction – File I/O 566

Masters with Distinction – Persistent Variables .567
Ph.D. Level.....567
Master Professional Engineer 568
SFC Approach to Bachelors Level 568
ST Approaches to Bachelors Level.....568

Detailed Table of Content 575