

# INTERNATIONAL STANDARD

# IEC 61804-2

First edition  
2004-05

---

---

## Function blocks (FB) for process control –

### Part 2: Specification of FB concept and Electronic Device Description Language (EDDL)

© IEC 2004 — Copyright - all rights reserved

No part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the publisher.

International Electrotechnical Commission, 3, rue de Varembe, PO Box 131, CH-1211 Geneva 20, Switzerland  
Telephone: +41 22 919 02 11 Telefax: +41 22 919 03 00 E-mail: [inmail@iec.ch](mailto:inmail@iec.ch) Web: [www.iec.ch](http://www.iec.ch)



Commission Electrotechnique Internationale  
International Electrotechnical Commission  
Международная Электротехническая Комиссия

PRICE CODE

**XN**

*For price, see current catalogue*

## CONTENTS

FOREWORD.....	22
INTRODUCTION.....	24
1 Scope .....	25
2 Normative references .....	26
3 Terms and definitions, abbreviated terms and acronyms, and conventions for lexical structures .....	27
3.1 Terms and definitions .....	27
3.2 Abbreviated terms and acronyms .....	34
3.3 Conventions for lexical structures .....	34
4 General Function Block (FB) definition and EDD model .....	36
4.1 Device structure (device model).....	36
4.1.1 Device model description .....	36
4.1.2 FB type.....	40
4.1.3 FB execution.....	41
4.1.4 Reference between IEC/PAS 61499-1, IEC/PAS 61499-2 and IEC 61804 models .....	42
4.1.5 UML specification of the device model.....	42
4.1.6 Classification of the algorithms.....	44
4.1.7 Algorithm description .....	45
4.1.8 Input and output variables and parameter definition .....	45
4.1.9 Choice of variables and parameters.....	46
4.1.10 Mode, Status and Diagnosis.....	46
4.2 Block combinations.....	46
4.2.1 Measurement channel .....	46
4.2.2 Actuation channel .....	47
4.2.3 Application.....	48
4.3 EDD and EDDL model .....	49
4.3.1 Overview of EDD and EDDL .....	49
4.3.2 EDD architecture .....	49
4.3.3 Concepts of EDD .....	49
4.3.4 Principles of the EDD development process.....	49
4.3.5 Interrelations between the lexical structure and formal definitions .....	50
4.3.6 Builtins .....	51
4.3.7 Profiles .....	51
5 Detailed block definition.....	51
5.1 General.....	51
5.2 Application FBs .....	51
5.2.1 Analog Input FB .....	51
5.2.2 Analog Output FB .....	53
5.2.3 Discrete Input FB .....	54
5.2.4 On/Off Actuation (Output) FB Discrete Output FB .....	56
5.2.5 Calculation FB .....	57
5.2.6 Control FB .....	58
5.3 Component FBs.....	59
5.4 Technology Block .....	59
5.4.1 Temperature Technology Block .....	59

5.4.2	Pressure Technology Block .....	62
5.4.3	Modulating Actuation Technology Block .....	64
5.4.4	On/Off Actuation Technology Block .....	66
5.5	Device (Resource) Block .....	68
5.5.1	Identification .....	68
5.5.2	Device state .....	69
5.5.3	Message .....	71
5.5.4	Initialization .....	71
5.6	Algorithms common to all blocks .....	71
5.6.1	Data Input/Data Output status .....	71
5.6.2	Validity .....	71
5.6.3	Restart Initialization .....	71
5.6.4	Fail-safe .....	72
5.6.5	Remote Cascade Initialization .....	72
6	FB Environment .....	73
7	Mapping to System Management .....	73
8	Mapping to Communication .....	73
9	Electronic Device Description Language .....	75
9.1	Overview .....	75
9.1.1	EDDL features .....	75
9.1.2	Syntax representation .....	75
9.1.3	EDD language elements .....	75
9.1.4	Basic construction elements .....	75
9.1.5	Common attributes .....	81
9.1.6	Special elements .....	81
9.1.7	Rules for instances .....	81
9.1.8	Rules for list of VARIABLES .....	81
9.2	EDD identification information .....	82
9.2.1	General structure .....	82
9.2.2	Specific attributes .....	82
9.3	BLOCK .....	84
9.3.1	BLOCK_A .....	84
9.3.2	BLOCK_B .....	88
9.4	COLLECTION .....	90
9.4.1	General structure .....	90
9.4.2	Specific attributes - item-type .....	90
9.5	COMMAND .....	91
9.5.1	General structure .....	91
9.5.2	Specific attributes .....	92
9.6	CONNECTION .....	96
9.6.1	General structure .....	96
9.6.2	Specific attribute - APPINSTANCE .....	96
9.7	DOMAIN .....	97
9.7.1	General structure .....	97
9.7.2	Specific attribute - HANDLING .....	97
9.8	EDIT_DISPLAY .....	98
9.8.1	General structure .....	98
9.8.2	Specific attributes .....	98

9.9	IMPORT .....	100
9.9.1	General structure .....	100
9.9.2	Specific attributes – attribute-redefinition .....	102
9.10	LIKE .....	108
9.11	MENU .....	108
9.11.1	General structure .....	108
9.11.2	Specific attributes .....	109
9.11.3	Sequence diagrams for actions .....	114
9.12	METHOD .....	118
9.12.1	General structure .....	118
9.12.2	Specific attributes .....	118
9.13	PROGRAM .....	119
9.13.1	General structure .....	119
9.13.2	Specific attributes - ARGUMENT .....	119
9.14	RECORD .....	120
9.15	REFERENCE_ARRAY .....	120
9.15.1	General structure .....	120
9.15.2	Specific attributes - ELEMENTS .....	121
9.16	Relations .....	121
9.16.1	General structure .....	121
9.16.2	REFRESH .....	121
9.16.3	UNIT .....	122
9.16.4	WRITE_AS_ONE .....	122
9.17	RESPONSE_CODES .....	122
9.18	VALUE_ARRAY .....	123
9.18.1	General structure .....	123
9.18.2	Specific attributes .....	123
9.19	VARIABLE .....	124
9.19.1	General structure .....	124
9.19.2	Specific attributes .....	125
9.20	VARIABLE_LIST .....	138
9.21	Common attributes .....	138
9.21.1	DEFINITION .....	138
9.21.2	HELP .....	139
9.21.3	LABEL .....	139
9.21.4	MEMBERS .....	139
9.21.5	RESPONSE_CODES .....	140
9.22	Output redirection (OPEN and CLOSE) .....	141
9.23	Conditional expression .....	141
9.24	Referencing .....	142
9.24.1	Referencing an EDD instance .....	142
9.24.2	Referencing members of a RECORD .....	142
9.24.3	Referencing elements of a VALUE_ARRAY .....	143
9.24.4	Referencing members of a COLLECTION .....	143
9.24.5	Referencing elements of a REFERENCE_ARRAY .....	144
9.24.6	Referencing members of a VARIABLE_LISTS .....	144
9.24.7	Referencing elements of BLOCK_A PARAMETERS .....	145
9.24.8	Referencing elements of BLOCK_A PARAMETER_LISTS .....	145
9.24.9	Referencing BLOCK_A CHARACTERISTICS .....	146

9.25	Strings.....	146
9.25.1	Specifying a string as a string literal.....	146
9.25.2	Specifying a string as a string variable.....	146
9.25.3	Specifying a string as an enumeration value.....	146
9.25.4	Specifying a string as a dictionary reference.....	147
9.25.5	Referencing HELP and LABEL attributes of EDD instances.....	147
9.25.6	String operations.....	147
9.25.7	Prompt String Formats.....	148
9.26	Expression.....	148
9.26.2	Primary expressions.....	148
9.26.3	Unary expressions.....	150
9.26.4	Binary expressions.....	150
9.27	Text dictionary.....	153
10	Conformance statement.....	154
	Annex A (informative) Parameter description.....	155
	Annex B (normative) IEC 61804 Conformance Declaration.....	160
	Annex C (normative) EDDL Formal Definition.....	161
C.1	EDDL Preprocessor.....	161
C.1.1	General structure.....	161
C.1.2	Directives.....	161
C.1.3	Predefined macros.....	164
C.1.4	NEWLINE characters.....	164
C.1.5	Comments.....	164
C.2	Conventions.....	164
C.2.1	Integer constants.....	164
C.2.2	Floating point constants.....	165
C.2.3	String literals.....	165
C.2.4	Using language codes in string constants.....	166
C.3	Operators.....	166
C.4	Keywords.....	167
C.5	Terminals.....	169
C.6	letter (letter digit _)*Formal EDDL syntax.....	169
C.6.1	General.....	169
C.6.2	EDD identification information.....	169
C.6.3	BLOCK_A and BLOCK_B.....	171
C.6.4	COLLECTION.....	173
C.6.5	COMMAND.....	173
C.6.6	CONNECTION.....	177
C.6.7	DOMAIN.....	177
C.6.8	EDIT_DISPLAY.....	178
C.6.9	IMPORT.....	179
C.6.10	LIKE.....	180
C.6.11	MENU.....	182
C.6.12	METHOD.....	185
C.6.13	PROGRAM.....	185
C.6.14	RECORDS.....	186
C.6.15	REFERENCE_ARRAY.....	186
C.6.16	Relations.....	187

C.6.17	RESPONSE_CODES .....	188
C.6.18	VALUE_ARRAY .....	189
C.6.19	VARIABLE .....	189
C.6.20	VARIABLE_LIST .....	197
C.6.21	Common attributes .....	197
C.6.22	OPEN, CLOSE .....	199
C.6.23	Expression .....	199
C.6.24	C-Grammer .....	201
C.6.25	Redefinition .....	204
C.6.26	References .....	216
Annex D (normative)	EDDL Builtin Library .....	218
D.1	General .....	218
D.2	Conventions for Builtin descriptions .....	218
D.3	Builtin abort .....	218
D.4	Builtin abort_on_all_comm_errors .....	219
D.5	Builtin ABORT_ON_ALL_COMM_STATUS .....	219
D.6	Builtin ABORT_ON_ALL_DEVICE_STATUS .....	220
D.7	Builtin ABORT_ON_ALL_RESPONSE_CODES .....	220
D.8	Builtin abort_on_all_response_codes .....	221
D.9	Builtin abort_on_comm_error .....	221
D.10	Builtin ABORT_ON_COMM_ERROR .....	222
D.11	Builtin ABORT_ON_COMM_STATUS .....	222
D.12	Builtin ABORT_ON_DEVICE_STATUS .....	223
D.13	Builtin ABORT_ON_NO_DEVICE .....	223
D.14	Builtin ABORT_ON_RESPONSE_CODE .....	224
D.15	Builtin abort_on_response_code .....	225
D.16	Builtin ACKNOWLEDGE .....	225
D.17	Builtin acknowledge .....	225
D.18	Builtin add_abort_method (version A) .....	226
D.19	Builtin add_abort_method (version B) .....	226
D.20	Builtin assign .....	227
D.21	Builtin assign_double .....	227
D.22	Builtin assign_float .....	228
D.23	Builtin assign_int .....	228
D.24	Builtin assign_var .....	228
D.25	Builtin atof .....	229
D.26	Builtin atoi .....	229
D.27	Builtin dassign .....	229
D.28	Builtin Date_to_DayOfMonth .....	230
D.29	Builtin Date_to_Month .....	230
D.30	Builtin Date_to_Year .....	231
D.31	Builtin DELAY .....	231
D.32	Builtin delay .....	231
D.33	Builtin DELAY_TIME .....	232

D.34	Builtin delayfor .....	232
D.35	Builtin DICT_ID .....	233
D.36	Builtin discard_on_exit.....	233
D.37	Builtin display.....	234
D.38	Builtin display_builtin_error.....	234
D.39	Builtin display_comm_error.....	234
D.40	Builtin display_comm_status .....	235
D.41	Builtin display_device_status .....	235
D.42	Builtin display_dynamics.....	236
D.43	Builtin display_message .....	236
D.44	Builtin display_response_code .....	237
D.45	Builtin display_response_status .....	238
D.46	Builtin display_xmtr_status .....	238
D.47	Builtin edit_device_value .....	238
D.48	Builtin edit_local_value .....	239
D.49	Builtin ext_send_command .....	240
D.50	Builtin ext_send_command_trans.....	240
D.51	Builtin fail_on_all_comm_errors .....	241
D.52	Builtin fail_on_all_response_codes .....	242
D.53	Builtin fail_on_comm_error .....	242
D.54	Builtin fail_on_response_code .....	243
D.55	Builtin fassign.....	243
D.56	Builtin fgetval .....	244
D.57	Builtin float_value .....	244
D.58	Builtin fsetval .....	244
D.59	Builtin ftoa.....	245
D.60	Builtin fvar_value.....	245
D.61	Builtin get_acknowledgement.....	245
D.62	Builtin get_comm_error.....	246
D.63	Builtin get_comm_error_string .....	247
D.64	Builtin get_date .....	247
D.65	Builtin get_date_value .....	247
D.66	Builtin get_dds_error .....	248
D.67	Builtin GET_DEV_VAR_VALUE .....	249
D.68	Builtin get_dev_var_value.....	249
D.69	Builtin get_dictionary_string.....	250
D.70	Builtin get_double.....	250
D.71	Builtin get_double_value.....	251
D.72	Builtin get_float .....	251
D.73	Builtin get_float_value .....	252
D.74	Builtin GET_LOCAL_VAR_VALUE .....	252
D.75	Builtin get_local_var_value .....	253
D.76	Builtin get_more_status .....	253

D.77	Builtin get_resolve_status.....	254
D.78	Builtin get_response_code.....	254
D.79	Builtin get_response_code_string .....	255
D.80	Builtin get_signed.....	255
D.81	Builtin get_signed_value.....	256
D.82	Builtin get_status_code_string .....	256
D.83	Builtin get_status_string .....	257
D.84	Builtin get_stddict_string.....	257
D.85	Builtin get_string .....	258
D.86	Builtin get_string_value .....	259
D.87	Builtin GET_TICK_COUNT .....	259
D.88	Builtin get_unsigned .....	259
D.89	Builtin get_unsigned_value.....	260
D.90	Builtin iassign.....	260
D.91	Builtin igetval .....	261
D.92	Builtin IGNORE_ALL_COMM_STATUS.....	261
D.93	Builtin IGNORE_ALL_DEVICE_STATUS .....	262
D.94	Builtin IGNORE_ALL_RESPONSE_CODES .....	262
D.95	Builtin IGNORE_COMM_ERROR .....	263
D.96	Builtin IGNORE_COMM_STATUS .....	263
D.97	Builtin IGNORE_DEVICE_STATUS.....	264
D.98	Builtin IGNORE_NO_DEVICE .....	264
D.99	Builtin IGNORE_RESPONSE_CODE .....	265
D.100	Builtin int_value.....	265
D.101	Builtin is_NaN .....	266
D.102	Builtin isetval.....	266
D.103	Builtin ITEM_ID .....	266
D.104	Builtin itoa.....	267
D.105	Builtin ivar_value.....	267
D.106	Builtin lassign.....	267
D.107	Builtin lgetval .....	268
D.108	Builtin LOG_MESSAGE .....	268
D.109	Builtin long_value .....	268
D.110	Builtin lsetval.....	269
D.111	Builtin lvar_value.....	269
D.112	Builtin MEMBER_ID.....	269
D.113	Builtin method_abort.....	270
D.114	Builtin ObjectReference .....	270
D.115	Builtin process_abort.....	270
D.116	Builtin put_date .....	271
D.117	Builtin put_date_value .....	271
D.118	Builtin put_double.....	272
D.119	Builtin put_double_value.....	272

D.120	Builtin put_float .....	273
D.121	Builtin put_float_value .....	273
D.122	Builtin PUT_MESSAGE .....	274
D.123	Builtin put_message .....	274
D.124	Builtin put_signed .....	275
D.125	Builtin put_signed_value .....	275
D.126	Builtin put_string .....	276
D.127	Builtin put_string_value .....	276
D.128	Builtin put_unsigned .....	277
D.129	Builtin put_unsigned_value .....	278
D.130	Builtin READ_COMMAND .....	278
D.131	Builtin read_value .....	279
D.132	Builtin remove_abort_method (version A) .....	279
D.133	Builtin remove_abort_method (version B) .....	280
D.134	Builtin remove_all_abort_methods .....	280
D.135	Builtin resolve_array_ref .....	280
D.136	Builtin resolve_block_ref .....	281
D.137	Builtin resolve_param_list_ref .....	281
D.138	Builtin resolve_param_ref .....	282
D.139	Builtin resolve_record_ref .....	282
D.140	Builtin retry_on_all_comm_errors .....	283
D.141	Builtin RETRY_ON_ALL_COMM_STATUS .....	283
D.142	Builtin RETRY_ON_ALL_DEVICE_STATUS .....	284
D.143	Builtin RETRY_ON_ALL_RESPONSE_CODES .....	284
D.144	Builtin retry_on_all_response_codes .....	285
D.145	Builtin RETRY_ON_COMM_ERROR .....	285
D.146	Builtin retry_on_comm_error .....	286
D.147	Builtin RETRY_ON_COMM_STATUS .....	286
D.148	Builtin RETRY_ON_DEVICE_STATUS .....	287
D.149	Builtin RETRY_ON_NO_DEVICE .....	287
D.150	Builtin RETRY_ON_RESPONSE_CODE .....	288
D.151	Builtin retry_on_response_code .....	288
D.152	Builtin rspcode_string .....	289
D.153	Builtin save_on_exit .....	289
D.154	Builtin save_values .....	290
D.155	Builtin SELECT_FROM_LIST .....	290
D.156	Builtin select_from_list .....	291
D.157	Builtin select_from_menu .....	291
D.158	Builtin send .....	292
D.159	Builtin send_all_values .....	292
D.160	Builtin send_command .....	293
D.161	Builtin send_command_trans .....	293
D.162	Builtin send_on_exit .....	294

D.163	Builtin send_trans.....	294
D.164	Builtin send_value .....	295
D.165	Builtin SET_NUMBER_OF_RETRIES.....	296
D.166	Builtin To_Date_and_Time.....	296
D.167	Builtin VARID .....	296
D.168	Builtin vassign .....	297
D.169	Builtin WRITE_COMMAND .....	297
D.170	Builtin XMTR_ABORT_ON_ALL_COMM_STATUS.....	297
D.171	Builtin XMTR_ABORT_ON_ALL_DEVICE_STATUS.....	298
D.172	Builtin XMTR_ABORT_ON_ALL_RESPONSE_CODES .....	298
D.173	Builtin XMTR_ABORT_ON_COMM_ERROR.....	299
D.174	Builtin XMTR_ABORT_ON_COMM_STATUS.....	299
D.175	Builtin XMTR_ABORT_ON_DATA .....	300
D.176	Builtin XMTR_ABORT_ON_DEVICE_STATUS .....	300
D.177	Builtin XMTR_ABORT_ON_NO_DEVICE.....	301
D.178	Builtin XMTR_ABORT_ON_RESPONSE_CODE .....	301
D.179	Builtin XMTR_IGNORE_ALL_COMM_STATUS.....	302
D.180	Builtin XMTR_IGNORE_ALL_DEVICE_STATUS.....	302
D.181	Builtin XMTR_IGNORE_ALL_RESPONSE_CODES .....	303
D.182	Builtin XMTR_IGNORE_COMM_ERROR.....	303
D.183	Builtin XMTR_IGNORE_COMM_STATUS.....	304
D.184	Builtin XMTR_IGNORE_DEVICE_STATUS.....	304
D.185	Builtin XMTR_IGNORE_NO_DEVICE.....	305
D.186	Builtin XMTR_IGNORE_RESPONSE_CODE .....	305
D.187	Builtin XMTR_RETRY_ON_ALL_DEVICE_STATUS.....	306
D.188	Builtin XMTR_RETRY_ON_ALL_RESPONSE_CODE .....	306
D.189	Builtin XMTR_RETRY_ON_ALL_RESPONSE_CODES .....	307
D.190	Builtin XMTR_RETRY_ON_COMM_ERROR.....	307
D.191	Builtin XMTR_RETRY_ON_COMM_STATUS.....	308
D.192	Builtin XMTR_RETRY_ON_DATA .....	308
D.193	Builtin XMTR_RETRY_ON_DEVICE_STATUS.....	309
D.194	Builtin XMTR_RETRY_ON_NO_DEVICE.....	309
D.195	Builtin XMTR_RETRY_ON_RESPONSE_CODE .....	310
D.196	Builtin YearMonthDay_to_Date .....	310
D.197	Builtins Return Codes.....	310
Annex E	(informative) EDD Example .....	312
Annex F	(normative) Profiles of EDDL and Builtins .....	331
F.1	Profile of EDDL and Builtins.....	331
F.2	Profiles for PROFIBUS .....	332
F.2.1	EDDL profile .....	332
F.2.2	Builtin profile.....	334
F.2.3	EDDL Formal Definition profile .....	337
F.3	Profiles for Fieldbus Foundation ®.....	338

F.3.1	EDDL profile .....	338
F.3.2	Builtin profile.....	341
F.3.3	EDDL Formal Definition profile .....	344
F.4	Profiles for HART® Communication Foundation (HCF) .....	345
F.4.1	EDDL profile .....	345
F.4.2	Builtin profile.....	347
F.4.3	EDDL Formal Definition profile .....	351
F.5	Data types.....	351
F.5.1	METHOD DEFINITIONS data types .....	351
F.5.2	Coding of data DATE .....	353
F.5.3	Coding of data DATE_AND_TIME.....	353
F.5.4	Coding of data DURATION.....	353
F.5.5	Coding of data TIME .....	353
F.5.6	Coding of data TIME_VALUE .....	354
F.5.7	Coding of PACKED_ASCII (6-BIT ASCII) DATA FORMAT .....	354
<b>Bibliography</b>	.....	<b>356</b>
Figure 1	– Position of the IEC 61804 series related to other standards and products .....	26
Figure 3	– FB structure may be distributed between devices (according to IEC/PAS 61499–1).....	37
Figure 4	– IEC 61804 FBs can be implemented in different devices.....	38
Figure 5	– General components of devices .....	38
Figure 7	– IEC 61804 block overview (graphical representation not normative).....	40
Figure 8	– UML class diagram of the device model .....	43
Figure 9	– Measurement process signal flow .....	47
Figure 10	– Actuation process signal flow.....	47
Figure 11	– Application process signal flow .....	48
Figure 12	– EDD generation process .....	50
Figure 13	– Analog Input FB .....	52
Figure 14	– Analog Output FB .....	53
Figure 15	– Discrete input FB.....	55
Figure 16	– Discrete Output FB .....	56
Figure 17	– Calculation FB.....	57
Figure 18	– Control FB.....	58
Figure 19	– Temperature Technology Block.....	60
Figure 20	– Pressure Technology Block.....	63
Figure 21	– Modulating actuation technology block .....	65
Figure 22	– On/Off Actuation Technology Block.....	67
Figure 23	– Harel state chart.....	70
Figure 24	– Application structure of ISO OSI Reference Model .....	73
Figure 25	– Client/Server relationship in terms of OSI Reference Model.....	74
Figure 26	– Mapping of IEC 61804 FBs to APOs.....	74
Figure 27	– BLOCK_A.....	76
Figure 28	– COLLECTION.....	76
Figure 29	– COMMAND .....	77

Figure 30 – DOMAIN.....	77
Figure 31 – EDIT_DISPLAY .....	77
Figure 32 – LIKE.....	78
Figure 33 – MENU .....	78
Figure 34 – PROGRAM.....	78
Figure 35 – RECORD.....	79
Figure 36 – REFERENCE_ARRAY .....	79
Figure 37 – REFRESH .....	79
Figure 38 – UNIT .....	79
Figure 39 – WRITE_AS_ONE.....	80
Figure 40 – VALUE_ARRAY.....	80
Figure 41 – VARIABLE.....	80
Figure 42 – VARIABLE_LIST.....	81
Figure 43 – EDDL import mechanisms.....	100
Figure 44 – MENU activation (ACCESS OFFLINE) .....	114
Figure 45 – Action performed after a new value is entered .....	114
Figure 46 – Action performed after all VARIABLE inputs of the MENU are accepted (ACCESS OFFLINE) .....	115
Figure 47 – Method execution .....	115
Figure 48 – MENU activation (ACCESS ONLINE) .....	116
Figure 49 – Cyclic reading of dynamic VARIABLES (ACCESS ONLINE) .....	117
Figure 50 – Action performed after all VARIABLE inputs of the MENU are accepted (ACCESS ONLINE).....	117
Figure 51 – Time for read and write operation .....	137
Figure E.1 – Example of an operator screen using EDD.....	312
Table 1 – Field attribute descriptions.....	35
Table 2 – References of model elements.....	42
Table 3 – Variables and parameter description template.....	45
Table 4 – Example of temperature sensors of Sensor_Type.....	61
Table 5 – Device status state table.....	69
Table 6 – Device status transition table .....	70
Table 7 – DD_REVISION attribute.....	82
Table 8 – DEVICE_REVISION attribute .....	83
Table 9 – DEVICE_TYPE attribute .....	83
Table 10 – EDD_PROFILE attribute .....	83
Table 11 – EDD_VERSION attribute.....	84
Table 12 – MANUFACTURER attribute.....	84
Table 13 – MANUFACTURER_EXT attribute .....	84
Table 14 – BLOCK_A attributes.....	85
Table 15 – CHARACTERISTIC attribute .....	85
Table 16 – PARAMETER attributes .....	86
Table 17 – COLLECTION_ITEMS attribute .....	86
Table 18 – EDIT_DISPLAY_ITEMS attribute.....	86

Table 19 – MENU_ITEMS attribute.....	86
Table 20 – METHOD_ITEMS attribute .....	87
Table 21 – PARAMETER_LISTS attributes .....	87
Table 22 – REFERENCE_ARRAY_ITEMS attribute.....	87
Table 23 – REFRESH_ITEMS attribute.....	88
Table 24 – UNIT_ITEMS attribute.....	88
Table 25 – WRITE_AS_ONE_ITEMS attribute .....	88
Table 26 – BLOCK_B attributes.....	89
Table 27 – NUMBER attributes.....	89
Table 28 – TYPE attributes .....	89
Table 29 – COLLECTION attributes.....	90
Table 30 – item-type .....	90
Table 31 – COMMAND attributes .....	91
Table 32 – OPERATION attribute .....	92
Table 33 – TRANSACTION attributes .....	92
Table 34 – REPLY and REQUEST attributes .....	93
Table 35 – INDEX attribute .....	94
Table 36 – BLOCK_B attribute .....	94
Table 37 – NUMBER attribute .....	95
Table 38 – SLOT attribute.....	95
Table 39 – CONNECTION attribute .....	95
Table 40 – HEADER attribute.....	96
Table 41 – MODULE attribute .....	96
Table 42 – CONNECTION attribute .....	96
Table 43 – APPINSTANCE attribute .....	97
Table 44 – DOMAIN attributes.....	97
Table 45 – HANDLING attribute .....	97
Table 46 – EDIT_DISPLAY attributes .....	98
Table 47 – EDIT_ITEMS attribute.....	98
Table 48 – DISPLAY_ITEM attributes.....	99
Table 49 – POST_EDIT_ACTIONS, PRE_EDIT_ACTIONS attribute .....	99
Table 50 – Importing Device Description .....	101
Table 51 – Redefinition attributes.....	102
Table 52 – Redefinition rules for BLOCK_A attributes .....	102
Table 53 – Redefinition rules for BLOCK_B attributes .....	103
Table 54 – Redefinition rules for COLLECTION attributes.....	103
Table 55 – Redefinition rules for COMMAND attributes.....	103
Table 56 – Redefinition rules for CONNECTION attributes .....	104
Table 57 – Redefinition rules for DOMAIN attributes .....	104
Table 58 – Redefinition rules for EDIT_DISPLAY attributes.....	104
Table 59 – Redefinition rules for MENU attributes.....	104
Table 60 – Redefinition rules for METHOD attributes .....	105
Table 61 – Redefinition rules for PROGRAM attributes .....	105

Table 62 – Redefinition rules for RECORD attributes .....	105
Table 63 – Redefinition rules for REFERENCE_ARRAY attributes .....	106
Table 64 – Redefinition rules for RESPONSE_CODES attributes .....	106
Table 65 – Redefinition rules for VALUE_ARRAY attributes .....	106
Table 66 – Redefinition rules for VARIABLE attributes .....	107
Table 67 – Redefinition rules for VARIABLE_LIST attributes .....	107
Table 68 – LIKE attributes .....	108
Table 69 – MENU attribute .....	108
Table 70 – ITEMS attribute .....	109
Table 71 – ACCESS attribute .....	110
Table 72 – ENTRY attribute .....	110
Table 73 – POST_EDIT_ACTIONS, PRE_EDIT_ACTIONS, POST_READ_ACTIONS, PRE_READ_ACTIONS, POST_WRITE_ACTIONS, PRE_WRITE_ACTIONS attributes.....	110
Table 74 – PURPOSE attribute .....	112
Table 75 – ROLE attribute.....	113
Table 76 – STYLE attribute .....	113
Table 77 – VALIDITY attributes .....	114
Table 78 – METHOD attributes.....	118
Table 79 – ACCESS attributes .....	118
Table 80 – VALIDITY attributes .....	119
Table 81 – PROGRAM attributes.....	119
Table 82 – ARGUMENT attribute.....	120
Table 83 – RECORD attributes.....	120
Table 84 – REFERENCE_ARRAY attribute.....	120
Table 85 – ELEMENTS attribute.....	121
Table 86 – REFRESH attributes .....	121
Table 87 – UNIT attributes .....	122
Table 88 – WRITE_AS_ONE attribute .....	122
Table 89 – RESPONSE_CODES attributes.....	123
Table 90 – VALUE_ARRAY attributes.....	123
Table 91 – NUMBER_OF_ELEMENTS attribute .....	124
Table 92 – TYPE attribute.....	124
Table 93 – VARIABLE attributes.....	125
Table 94 – CLASS attributes .....	126
Table 95 – TYPE attributes .....	127
Table 96 – DOUBLE, FLOAT, INTEGER, UNSIGNED_INTEGER attributes .....	128
Table 97 – BIT_ENUMERATED attributes .....	130
Table 98 – status–class attributes .....	131
Table 99 – ALL, AO, DV, TV attributes .....	132
Table 100 – Enumerated types attributes .....	132
Table 101 – Index type attributes .....	133
Table 102 – Object reference type attribute .....	133
Table 103 – DEFAULT_REFERENCE attributes .....	133

Table 104 – String types attributes.....	134
Table 105 – CONSTANT_UNIT attribute.....	135
Table 106 – HANDLING attribute .....	135
Table 107 – POST_EDIT_ACTIONS, PRE_EDIT_ACTIONS, POST_READ_ACTIONS, PRE_READ_ACTIONS, POST_WRITE_ACTIONS, PRE_WRITE_ACTIONS attributes.....	135
Table 108 – READ/WRITE_TIMEOUT attributes .....	137
Table 109 – STYLE attribute .....	137
Table 110 – VALIDITY attributes .....	138
Table 111 – VARIABLE_LIST attributes.....	138
Table 112 – DEFINITION attributes .....	139
Table 113 – HELP attribute .....	139
Table 114 – LABEL attribute .....	139
Table 116 – RESPONSE_CODES attribute.....	140
Table 117 – OPEN and CLOSE attributes.....	141
Table 118 – IF, SELECT conditional.....	142
Table 119 – Referencing an EDD instance .....	142
Table 120 – Referencing elements of RECORD.....	143
Table 121 – Referencing elements of VALUE_ARRAY .....	143
Table 122 – Referencing members of COLLECTION.....	143
Table 123 – Referencing members of REFERENCE_ARRAY .....	144
Table 124 – Referencing members of VARIABLE_LISTS .....	144
Table 125 – Referencing members of a BLOCK_A PARAMETERS.....	145
Table 126 – Referencing members of BLOCK_A PARAMETER_LISTS.....	145
Table 127 – Referencing BLOCK_A CHARACTERISTICS.....	146
Table 128 – String as a string literal.....	146
Table 129 – String as a string variable .....	146
Table 130 – String as an enumeration value.....	147
Table 131 – String as a dictionary reference.....	147
Table 132 – Referencing HELP and LABEL attributes of EDD instances.....	147
Table 133 – String operation .....	148
Table 134 – Format specifier.....	148
Table 135 – Primary expressions .....	149
Table 136 – Attribute values of VARIABLES.....	149
Table 137 – Unary expressions .....	150
Table 138 – Multiplicative operators .....	151
Table 139 – Additive operators.....	151
Table 140 – Shift operators.....	151
Table 141 – Relational operators.....	151
Table 142 – Equality operators.....	152
Table 143 – Text dictionary attributes.....	153
Table A.1 – Parameter description .....	155
Table B.1 – Conformance (sub)clause selection table.....	160
Table B.2 – Contents of (sub)clause selection tables .....	160

Table C.1 – Conventions for Integer Constants.....	165
Table C.2 – Using escape sequences in string literals .....	166
Table C.3 – Using language codes in string literals.....	166
Table C.4 – EDDL Operators.....	167
Table C.5 – EDDL Keywords.....	167
Table D.1 – Format for the Builtins lexical element tables .....	218
Table D.2 – Contents of the lexical element table .....	218
Table D.3 – Builtin abort .....	219
Table D.4 – Builtin abort_on_all_comm_errors.....	219
Table D.5 – Builtin ABORT_ON_ALL_COMM_STATUS.....	220
Table D.6 – Builtin ABORT_ON_ALL_DEVICE_STATUS.....	220
Table D.7 – Builtin ABORT_ON_ALL_RESPONSE_CODES .....	221
Table D.8 – Builtin abort_on_all_response_codes.....	221
Table D.9 – Builtin abort_on_comm_error.....	221
Table D.10 – Builtin ABORT_ON_COMM_ERROR.....	222
Table D.11 – Builtin ABORT_ON_COMM_STATUS.....	222
Table D.12 – Builtin ABORT_ON_DEVICE_STATUS.....	223
Table D.13 – Builtin ABORT_ON_NO_DEVICE.....	223
Table D.14 – Builtin ABORT_ON_RESPONSE_CODE .....	224
Table D.15 – Builtin abort_on_response_code.....	225
Table D.16 – Builtin ACKNOWLEDGE .....	225
Table D.17 – Builtin acknowledge.....	226
Table D.18 – Builtin add_abort_method .....	226
Table D.19 – Builtin add_abort_method .....	227
Table D.20 – Builtin assign .....	227
Table D.21 – Builtin assign_double .....	228
Table D.22 – Builtin assign_float .....	228
Table D.23 – Builtin assign_int.....	228
Table D.24 – Builtin assign_var .....	229
Table D.25 – Builtin atof .....	229
Table D.26 – Builtin atoi.....	229
Table D.27 – Builtin dassign.....	230
Table D.28 – Builtin Date_to_DayOfMonth.....	230
Table D.29 – Builtin Date_to_Month .....	230
Table D.30 – Builtin Date_to_Year .....	231
Table D.31 – Builtin DELAY .....	231
Table D.32 – Builtin delay .....	232
Table D.33 – Builtin DELAY_TIME .....	232
Table D.34 – Builtin delayfor .....	232
Table D.35 – Builtin DICT_ID .....	233
Table D.36 – Builtin discard_on_exit .....	234
Table D.37 – Builtin display.....	234
Table D.38 – Builtin display_builtin_error .....	234

Table D.39 – Builtin display_comm_error.....	235
Table D.40 – Builtin display_comm_status.....	235
Table D.41 – Builtin display_device_status.....	236
Table D.42 – Builtin display_dynamics.....	236
Table D.43 – Builtin display_message.....	237
Table D.44 – Builtin display_response_code.....	237
Table D.45 – Builtin display_response_status.....	238
Table D.46 – Builtin display_xmtr_status.....	238
Table D.47 – Builtin edit_device_value.....	239
Table D.48 – Builtin edit_local_value.....	240
Table D.49 – Builtin ext_send_command.....	240
Table D.50 – Builtin ext_send_command_trans.....	241
Table D.51 – Builtin fail_on_all_comm_errors.....	241
Table D.52 – Builtin fail_on_all_response_codes.....	242
Table D.53 – Builtin fail_on_comm_error.....	243
Table D.54 – Builtin fail_on_response_code.....	243
Table D.55 – Builtin fassign.....	244
Table D.56 – Builtin fgetval.....	244
Table D.57 – Builtin float_value.....	244
Table D.58 – Builtin fsetval.....	245
Table D.59 – Builtin ftoa.....	245
Table D.60 – Builtin fvar_value.....	245
Table D.61 – Builtin get_acknowledgement.....	246
Table D.62 – Builtin get_comm_error.....	246
Table D.63 – Builtin get_comm_error_string.....	247
Table D.64 – Builtin get_date.....	247
Table D.65 – Builtin get_date_value.....	248
Table D.66 – Builtin get_dds_error.....	248
Table D.67 – Builtin GET_DEV_VAR_VALUE.....	249
Table D.68 – Builtin get_dev_var_value.....	249
Table D.69 – Builtin get_dictionary_string.....	250
Table D.70 – Builtin get_double.....	250
Table D.71 – Builtin get_double_value.....	251
Table D.72 – Builtin get_float.....	251
Table D.73 – Builtin get_float_value.....	252
Table D.74 – Builtin GET_LOCAL_VAR_VALUE.....	252
Table D.75 – Builtin get_local_var_value.....	253
Table D.76 – Builtin get_more_status.....	253
Table D.77 – Builtin get_resolve_status.....	254
Table D.78 – Builtin get_response_code.....	255
Table D.79 – Builtin get_response_code_string.....	255
Table D.80 – Builtin get_signed.....	256
Table D.81 – Builtin get_signed_value.....	256

Table D.82 – Builtin get_status_code_string .....	257
Table D.83 – Builtin get_status_string .....	257
Table D.84 – Builtin get_stddict_string .....	258
Table D.85 – Builtin get_string .....	258
Table D.86 – Builtin get_string_value .....	259
Table D.87 – Builtin GET_TICK_COUNT .....	259
Table D.88 – Builtin get_unsigned .....	260
Table D.89 – Builtin get_unsigned_value .....	260
Table D.90 – Builtin iassign.....	261
Table D.91 – Builtin igetval .....	261
Table D.92 – Builtin IGNORE_ALL_COMM_STATUS .....	262
Table D.93 – Builtin IGNORE_ALL_DEVICE_STATUS .....	262
Table D.94 – Builtin IGNORE_ALL_RESPONSE_CODES .....	263
Table D.95 – Builtin IGNORE_COMM_ERROR .....	263
Table D.96 – Builtin IGNORE_COMM_STATUS .....	264
Table D.97 – Builtin IGNORE_DEVICE_STATUS .....	264
Table D.98 – Builtin IGNORE_NO_DEVICE .....	265
Table D.99 – Builtin IGNORE_RESPONSE_CODE .....	265
Table D.100 – Builtin int_value .....	265
Table D.101 – Builtin is_NaN .....	266
Table D.102 – Builtin isetval.....	266
Table D.103 – Builtin ITEM_ID .....	266
Table D.104 – Builtin itoa .....	267
Table D.105 – Builtin ivar_value .....	267
Table D.106 – Builtin lassign.....	267
Table D.107 – Builtin lgetval .....	268
Table D.108 – Lexical elements of Builtin LOG_MESSAGE .....	268
Table D.109 – Builtin long_value .....	268
Table D.110 – Builtin lsetval.....	269
Table D.111 – Builtin lvar_value .....	269
Table D.112 – Builtin MEMBER_ID.....	269
Table D.113 – Builtin method_abort .....	270
Table D.114 – Builtin process_abort.....	270
Table D.115 – Builtin process_abort.....	271
Table D.116 – Builtin put_date .....	271
Table D.117 – Builtin put_date_value .....	272
Table D.118 – Builtin put_double.....	272
Table D.119 – Builtin put_double_value.....	273
Table D.120 – Builtin put_float .....	273
Table D.121 – Builtin put_float_value .....	274
Table D.122 – Builtin PUT_MESSAGE .....	274
Table D.123 – Builtin put_message .....	275
Table D.124 – Builtin put_signed.....	275

Table D.125 – Builtin put_signed_value.....	276
Table D.126 – Builtin put_string .....	276
Table D.127 – Builtin put_string_value .....	277
Table D.128 – Builtin put_unsigned .....	278
Table D.129 – Builtin put_unsigned_value .....	278
Table D.130 – Lexical elements of Builtin READ_COMMAND .....	279
Table D.131 – Builtin read_value.....	279
Table D.132 – Builtin remove_abort_method .....	279
Table D.133 – Builtin remove_abort_method .....	280
Table D.134 – Builtin remove_all_abort_methods .....	280
Table D.135 – Builtin resolve_array_ref.....	281
Table D.136 – Builtin resolve_block_ref.....	281
Table D.137 – Builtin resolve_param_list_ref.....	282
Table D.138 – Builtin resolve_parm_ref.....	282
Table D.139 – Builtin resolve_record_ref.....	283
Table D.140 – Builtin retry_on_all_comm_errors.....	283
Table D.141 – Builtin RETRY_ON_ALL_COMM_STATUS .....	284
Table D.142 – Builtin RETRY_ON_ALL_DEVICE_STATUS .....	284
Table D.143 – Builtin RETRY_ON_ALL_RESPONSE_CODES .....	285
Table D.144 – Builtin retry_on_all_response_codes.....	285
Table D.145 – Builtin RETRY_ON_COMM_ERROR .....	286
Table D.146 – Builtin retry_on_comm_error.....	286
Table D.147 – Builtin RETRY_ON_COMM_STATUS.....	287
Table D.148 – Builtin RETRY_ON_DEVICE_STATUS.....	287
Table D.149 – Builtin RETRY_ON_NO_DEVICE .....	288
Table D.150 – Builtin RETRY_ON_RESPONSE_CODE .....	288
Table D.151 – Builtin retry_on_response_code.....	289
Table D.152 – Builtin rspcode_string .....	289
Table D.153 – Builtin save_on_exit .....	290
Table D.154 – Builtin save_values.....	290
Table D.155 – Builtin SELECT_FROM_LIST.....	290
Table D.156 – Builtin select_from_list.....	291
Table D.157 – Builtin select_from_menu.....	292
Table D.158 – Builtin send .....	292
Table D.159 – Builtin send_all_values .....	293
Table D.160 – Builtin send_command.....	293
Table D.161 – Builtin send_command_trans .....	294
Table D.162 – Builtin send_on_exit .....	294
Table D.163 – Builtin send_trans.....	295
Table D.164 – Builtin send_value .....	295
Table D.165 – Builtin SET_NUMBER_OF_RETRIES.....	296
Table D.166 – Builtin To_Date_and_Time.....	296
Table D.167 – Builtin VARID .....	297

Table D.168 – Builtin vassign .....	297
Table D.169 – Builtin WRITE_COMMAND .....	297
Table D.170 – Builtin XMTR_ABORT_ON_ALL_COMM_STATUS.....	298
Table D.171 – Builtin XMTR_ABORT_ON_ALL_DEVICE_STATUS.....	298
Table D.172 – Builtin XMTR_ABORT_ON_ALL_RESPONSE_CODES .....	299
Table D.173 – Builtin XMTR_ABORT_ON_COMM_ERROR.....	299
Table D.174 – Builtin XMTR_ABORT_ON_COMM_STATUS .....	300
Table D.175 – Builtin XMTR_ABORT_ON_DATA .....	300
Table D.176 – Builtin XMTR_ABORT_ON_DEVICE_STATUS .....	301
Table D.177 – Builtin XMTR_ABORT_ON_NO_DEVICE.....	301
Table D.178 – Builtin XMTR_ABORT_ON_RESPONSE_CODE .....	302
Table D.179 – Builtin XMTR_IGNORE_ALL_COMM_STATUS.....	302
Table D.180 – Builtin XMTR_IGNORE_ALL_DEVICE_STATUS.....	303
Table D.181 – Builtin XMTR_IGNORE_ALL_RESPONSE_CODES .....	303
Table D.182 – Builtin XMTR_IGNORE_COMM_ERROR.....	304
Table D.183 – Builtin XMTR_IGNORE_COMM_STATUS.....	304
Table D.184 – Builtin XMTR_IGNORE_DEVICE_STATUS.....	305
Table D.185 – Builtin XMTR_IGNORE_NO_DEVICE.....	305
Table D.186 – Builtin XMTR_IGNORE_RESPONSE_CODE .....	306
Table D.187 – Builtin XMTR_RETRY_ON_ALL_DEVICE_STATUS.....	306
Table D.188 – Builtin XMTR_RETRY_ON_ALL_RESPONSE_CODE .....	307
Table D.189 – Builtin XMTR_RETRY_ON_ALL_RESPONSE_CODES .....	307
Table D.190 – Builtin XMTR_RETRY_ON_COMM_ERROR.....	307
Table D.191 – Builtin XMTR_RETRY_ON_COMM_STATUS.....	308
Table D.192 – Builtin XMTR_RETRY_ON_DATA .....	308
Table D.193 – Builtin XMTR_RETRY_ON_DEVICE_STATUS.....	309
Table D.194 – Builtin XMTR_RETRY_ON_NO_DEVICE.....	309
Table D.195 – Builtin XMTR_RETRY_ON_RESPONSE_CODE .....	310
Table D.196 – Builtin YearMonthDay_to_Date .....	310
Table D.197 – Contents of the return codes description table.....	311
Table D.198 – Return Code Description.....	311
Table F.1 – Profile selection tables .....	331
Table F.2 – EDDL Formal Definition profile tables.....	331
Table F.3 – Contents of selection tables.....	331
Table F.4 – EDDL element selection for PROFIBUS .....	332
Table F.5 – Builtin profile for PROFIBUS .....	334
Table F.7 – EDDL element selection for Fieldbus Foundation .....	338
Table F.8 – Builtin profile for Fieldbus Foundation .....	341
Table F.9 – EDDL element selection for HCF.....	344
Table F.10 – Builtin profile for HCF .....	345
Table F.11 – METHOD DEFINITIONS data types.....	347
Table F.12 – VARIABLE TYPES.....	351
Table F.13 – DATE coding .....	351

Table F.14 – DATE_AND_TIME coding .....	352
Table F.15 – DURATION coding.....	353
Table F.16 – TIME coding.....	353
Table F.17 – TIME_VALUE coding .....	353
Table F.18 – PACKED_ASCII coding.....	354

# INTERNATIONAL ELECTROTECHNICAL COMMISSION

---

## FUNCTION BLOCKS (FB) FOR PROCESS CONTROL –

### Part 2: Specification of FB concept and Electronic Device Description Language (EDDL)

#### FOREWORD

- 1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
- 2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
- 3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
- 4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
- 5) IEC provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with an IEC Publication.
- 6) All users should ensure that they have the latest edition of this publication.
- 7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
- 8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.

The International Electrotechnical Commission (IEC) draws attention to the fact that it is claimed that compliance with this document may involve the use of patents

U.S. Patent No. 5,333,114

U.S. Patent No. 5,485,400

U.S. Patent No. 5,825,664

U.S. Patent No. 5,909,368

U.S. Patent Pending No. 08/916,178

Australian Patent No. 638507

Canadian Patent No. 2,066,743

European Patent No. 0495001

Validated in:

UK – Patent No. 0495001

France – Patent No. 0495001

Germany – Patent No. 69032954.7

Netherlands – Patent No. 0495001

Japan Patent No. 3137643

IEC take no position concerning the evidence, validity and scope of this patent right. The holder of this patent right has assured the IEC that he is willing to negotiate licenses under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holder of this patent right is registered with IEC. Information may be obtained from:

Fieldbus Foundation,  
9390 Research Boulevard, Suite II-250,  
Austin, Texas, USA 78759,  
Attention: President.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights other than those identified above. IEC shall not be held responsible for identifying any or all such patent rights.

The International Standard IEC 61804-2 has been prepared by subcommittee 65C: Digital communications, of IEC technical committee 65: Industrial-process measurement and control.

This first edition of IEC 61804-2 cancels and replaces the IEC/PAS 61804-2 (2002).

The text of this standard is based on the following documents:

FDIS	Report on voting
65C/324/FDIS	65C/337/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

Parts of this standard are derived from "Fieldbus Foundation Specification FF-890 rev. 1.5 (undated)" and "Fieldbus Foundation Specification FF-900 rev. 1.4 (dated 1999-06-29)" and are used with permission of the Fieldbus Foundation<sup>1</sup>.

Parts of this standard are derived from "HART Device Description Language Specification, rev. 11.0 (August 5, 1996)" and "HART Device Description Language Method Builtins Library, rev. 10.1 (August 5, 1996)" and are used with the permission of the HART Communication Foundation<sup>2</sup>.

This publication has been drafted in accordance with ISO/IEC Directives, Part 2.

IEC 61804 consists of the following parts:

Part 1: *Overview of system aspects* (Technical Report)

Part 2: *Specification of FB concept and Electronic Device Description Language (EDDL)*

The committee has decided that the contents of this publication will remain unchanged until 2006. At this date, the publication will be

- reconfirmed;
- withdrawn;
- replaced by a revised edition, or
- amended.

<sup>1</sup> For additional information on Fieldbus Foundation, please contact: The Fieldbus Foundation, 9390 Research Boulevard, Austin, Texas 78759, USA. Tel: +1 512 794 8890. URL: [www.fieldbusfoundation.org](http://www.fieldbusfoundation.org).

<sup>2</sup> For additional information on HART Communication Foundation, please contact: HART Communication Foundation, 9390 Research Boulevard, Suite I-350, Austin, Texas 78759, USA. Tel: +1 512 794 0369. URL: [www.hartcomm.org](http://www.hartcomm.org).

## INTRODUCTION

This part of IEC 61804 provides conceptual Function Block specifications, which can be mapped to specific communication systems and their accompanying definitions by industrial groups and specify the Electronic Device Description Language (EDDL).

The EDDL and the device-related Electronic Device Description (EDD) is intended for use in industrial automation applications. These applications may include devices such as generic digital and analog input/output modules, motion controllers, human machine interfaces, sensors, closed-loop controllers, encoders, hydraulic valves, and programmable controllers.

This standard specifies a generic language for describing the properties of automation system components. The specified language is capable of describing

- device parameters and their dependencies;
- device functions, for example, simulation mode, calibration;
- graphical representations, for example, menus;
- interactions with control devices.

The language is called "Electronic Device Description Language (EDDL)" and is used to create "Electronic Device Descriptions (EDD)". These EDDs may be used with appropriate tools to generate interpretative code to support parameter handling, operation, and monitoring of automation system components such as remote I/Os, controllers, sensors, and programmable controllers. Tool implementation is outside the scope of this standard.

This standard specifies the semantic and lexical structure in a syntax independent manner. A specific syntax is defined in Annex C, but it is possible to use the semantic model also with different syntaxes.

NOTE The EDDL may also be used for the description of product properties in other domains.

## **FUNCTION BLOCKS (FB) FOR PROCESS CONTROL –**

### **Part 2: Specification of FB concept and Electronic Device Description Language (EDDL)**

#### **1 Scope**

This Part of IEC 61804 is applicable to Function Blocks (FB) for process control and specifies the Electronic Device Description Language (EDDL).

This standard specifies FB by using the result of harmonization work as regards several elements.

- a) The device model which defines the components of an IEC 61804-2 conformant device.
- b) Conceptual specifications of FBs for measurement, actuation and processing. This includes general rules for the essential features to support control, whilst avoiding details which stop innovation as well as specialization for different industrial sectors.
- c) The Electronic Device Description (EDD) technology, which enables the integration of real product details using the tools of the engineering life cycle.

This standard defines a subset of the requirements of IEC 61804-1 (hereafter referred to as Part 1) only, while Part 1 describes requirements for a distributed system.

The conformance statement in Annex B, which covers the conformance declaration, is related to this standard only. Requirements of Part 1 are not part of these conformance declarations.

The standardization work for FB was carried out by harmonizing the description of concepts of existing technologies. It results in an abstract level that allowed the definition of the common features in a unique way. This abstract vision is called here the conceptual FB specification and mapped to specific communication systems and their accompanying definitions by the industrial groups. This standard is also based on the abstract definitions of IEC/PAS 61499-1.

NOTE This standard can be mapped to ISO 15745-1.

The EDDL is ready for use and fills the gap between the conceptual FB specification and a product implementation. It allows the manufacturers to use the same description method for devices based on different technologies and platforms. Figure 1 shows these aspects.

There are solutions on the market today, which fulfil the requirements of this standard and show how the conceptual specification is implemented in a given technology. New technologies will need to find equivalent solutions (see Figure 4).

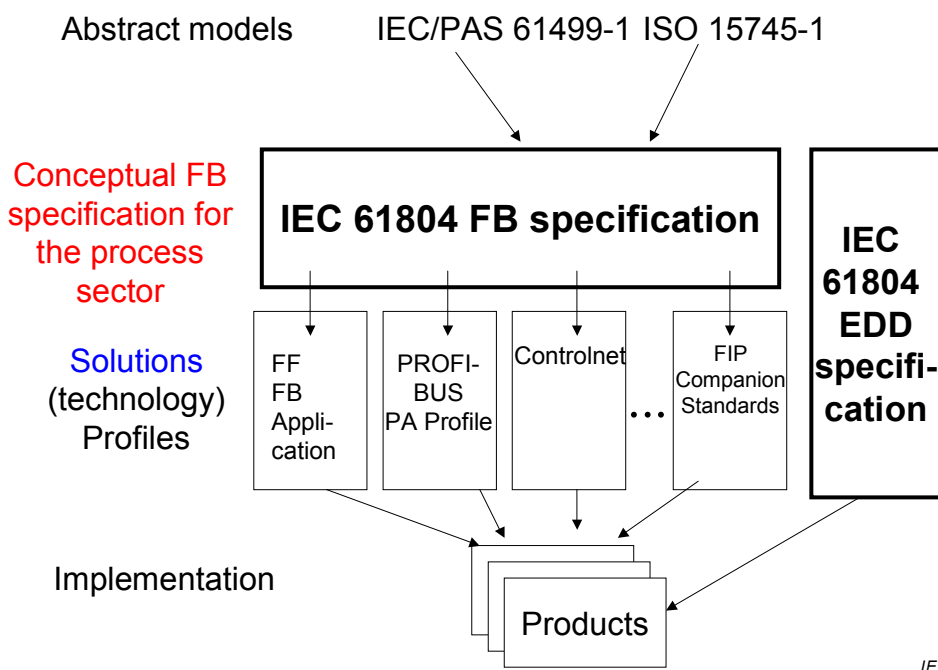


Figure 1 – Position of the IEC 61804 series related to other standards and products

## 2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 60050-351:1998, *International Electrotechnical Vocabulary (IEV) – Part 351: Automatic control*

IEC 60584-1, *Thermocouples – Part 1: Reference tables*

IEC 61131-3:2003, *Programmable controllers – Part 3: Programming languages*

IEC 61158:2003 (all parts), *Digital data communications for measurement and control – Fieldbus for use in industrial control systems*

IEC/PAS 61499-1:2000, *Function blocks for industrial-process measurement and control systems – Part 1: Architecture*

IEC/PAS 61499-2:2001, *Function blocks for industrial-process measurement and control systems – Part 2: Software tools requirements*

IEC 61804-1:2003, *Function blocks (FB) for process control – Part 1: Overview of system aspects*

ISO/IEC 2022:1994, *Information technology – Character code structure and extension techniques*

ISO/IEC 2375:2003, *Information technology – Procedure for registration of escape sequences and coded character sets*

ISO/IEC 7498-1:1994, *Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model*

ISO/IEC 8859-1, *Information technology - 8-bit single-byte coded graphic character sets – Part 1: Latin alphabet No. 1*

ISO/IEC 9899, *Programming languages – C*

ISO/IEC 10646-1, *Information technology – Universal Multiple-Octet Coded Character Set (UCS) – Part 1: Architecture and Basic Multilingual Plane*

IEEE 754:1985 (R1990), *Binary Floating-Point Arithmetic*