STRATON Compliance list to IEC 61131-3 *ed3*

IEC 61131-3 "PLC Programming Languages"

Implementer: STRATON AUTOMATION
Product: STRATON Version 12.0
Date: March 2023

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 1 – Character set					
1	ISO/IEC 10646 2011	✓	✓	✓	✓	
2a	Lower case characters	✓	✓	✓	✓	
2b	Number sign	✓	✓	✓	✓	
2c	Dollar sign	✓	✓	✓	✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 2 - Identifiers					
1	Upper case letters and numbers	✓	✓	✓	✓	
2	Upper and lower case letters, numbers, embedded underscore	✓	✓	✓	✓	
3	Upper and lower case, numbers, leading or embedded underscore	✓	✓	✓	✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 3 – Comments					
1	Single-line comment with // …			✓	✓	
2a	Multi-line comment with (* … *)			✓	✓	
2b	Multi-line comment with /* … */					
3a	Nested comment with (* .. (* .. *) ..*)					
3b	Nested comment with /* .. /* .. */ .. */					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 4 - Pragma					
1	Pragma with { … } curly brackets	✓	✓	✓	✓	Non IEC names

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 5 – Numeric Literals					
1	Integer literal	✓	✓	✓	✓	
2	Real literal	✓	✓	✓	✓	
3	Real literals with exponent	✓	✓	✓	✓	
4	Binary literal	✓	✓	✓	✓	
5	Octal literals				✓	
6	Hexadecimal literal	✓	✓	✓	✓	
7	Boolean zero and one	✓	✓	✓	✓	
8	Boolean FALSE and TRUE	✓	✓	✓	✓	
9	Typed literal	✓	✓	✓	✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 6 – Character string literals					
	Single-byte characters or character strings with ' '					
1a	Empty string (length zero)	✓	✓	✓	✓	
1b	String of length one or character CHAR containing a single character	✓	✓	✓	✓	

1c	String of length one or character CHAR containing the “space” character	✓	✓	✓	✓	
1d	String of length one or character CHAR containing the “single quote” character	✓	✓	✓	✓	
1e	String of length one or character CHAR containing the “double quote” character	✓	✓	✓	✓	
1f	Support of two character combinations of Table 7	✓	✓	✓	✓	
1g	Support of a character representation with ‘$’ and two hexadecimal characters	✓	✓	✓	✓	
	Double-byte characters or character strings with "					
2a	Empty string (length zero)	✓	✓	✓	✓	
2b	String of length one or character WCHAR containing a single character	✓	✓	✓	✓	
2c	String of length one or character WCHAR containing the “space” character	✓	✓	✓	✓	
2d	String of length one or character WCHAR containing the “single quote” character	✓	✓	✓	✓	
2e	String of length one or character WCHAR containing the “double quote” character	✓	✓	✓	✓	
2f	Support of two character combinations of Table 7					
2h	Support of a character representation with ‘$’ and four hexadecimal characters	✓	✓	✓	✓	
	Single-byte typed characters or string literals with #					
3a	Typed string					
3b	Typed character					
	Double-byte typed string literals with					
4a	Typed double-byte string (using “double quote” character)	✓	✓	✓	✓	
4b	Typed double-byte character (using “double quote” character)	✓	✓	✓	✓	
4c	Typed double-byte string (using “single quote” character)					
4d	Typed double-byte character (using “single quote” character)					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 7 - Two-character combinations in character strings					
1	Dollar sign	✓	✓	✓	✓	
2	Single quote	✓	✓	✓	✓	
3	Line feed	✓	✓	✓	✓	
4	Newline	✓	✓	✓	✓	
5	Form feed (page)	✓	✓	✓	✓	
6	Carriage return	✓	✓	✓	✓	
7	Tabulator	✓	✓	✓	✓	
8	Double quote					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 8 – Duration literals					
1a	d	✓	✓	✓	✓	
1b	h	✓	✓	✓	✓	
1c	m	✓	✓	✓	✓	
1d	s	✓	✓	✓	✓	
1e	ms	✓	✓	✓	✓	
1f	us					

1g	ns					
2a	short prefix without underscore	✓	✓	✓	✓	
2b	long prefix without underscore	✓	✓	✓	✓	
3a	short prefix with underscore	✓	✓	✓	✓	
3b	long prefix with underscore	✓	✓	✓	✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 9 - Date and time of day literals					
1a	Date literal (long prefix)					
1b	Date literal (short prefix)					
2a	Long date literal (long prefix)					
2b	Long date literal (short prefix)					
3a	Time of day literal (long prefix)					
3b	Time of day literal (short prefix)					
4a	Long time of day literal (short prefix)					
4d	Long time of day literal (long prefix)					
5a	Date and time literal (long prefix)					
5b	Date and time literal (short prefix)					
6a	Long date and time literal (long prefix)					
6b	Long date and time literal (short prefix)					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 10 - Elementary data types					
1	Boolean BOOL	✓	✓	✓	✓	
2	Short integer SINT	✓	✓	✓	✓	
3	Integer INT	✓	✓	✓	✓	
4	Double integer DINT	✓	✓	✓	✓	
5	Long integer LINT	✓	✓	✓	✓	
6	Unsigned short integer USINT	✓	✓	✓	✓	
7	Unsigned integer UINT	✓	✓	✓	✓	
8	Unsigned double integer UDINT	✓	✓	✓	✓	
9	Unsigned long integer ULINT	✓	✓	✓	✓	
10	Real numbers REAL	✓	✓	✓	✓	
11	Long reals LREAL	✓	✓	✓	✓	
12a	Duration TIME	✓	✓	✓	✓	
12b	Duration LTIME					
13a	Date DATE					
13b	Long date LDATE					
14a	Time of day TOD					
14b	Time of day LTOD					
15a	Date and time of day DT					
15b	Date and time of day LDT					
16a	Variable-length single-byte character string STRING	✓	✓	✓	✓	
16b	Variable-length double-byte character string WSTRING	✓	✓	✓	✓	
17a	Single-byte character CHAR					
17b	Double-byte character WCHAR					
18	Bit string 8 BYTE	✓	✓	✓	✓	
19	Bit string 16 WORD	✓	✓	✓	✓	
20	Bit string 32 DWORD	✓	✓	✓	✓	
21	Bit string 64 LWORD	✓	✓	✓	✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*

	Table 11 - Declaration of user-defined data types and initialization					
1a,b	Enumerated data types	✓	✓	✓	✓	
2a,b	Data types with named values					
3a,b	Subrange data types					
4a,b	Array data types	✓	✓	✓	✓	
5a,b	FB types and classes as array elements	✓	✓	✓	✓	
6a,b	Structured data type	✓	✓	✓	✓	
7a,b	FB types and classes as structure elements					
8a,b	Structured data type with relative addressing					
9a,b	Structured data type with relative addressing and overlap					
10a,b	Directly represented elements of a structure – partly specified					
11a,b	Directly derived data types					
12	Initialization using constant expressions	✓	✓	✓	✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 12 - Reference operations					
1	Declaration of a reference type					
2a	Assignment reference to reference					
2b	Assignment reference to parameter of function, function block and method					
2c	Comparison with NULL					
3a	REF(<variable>)					
3b	REF(<function block instance>)					
4	<reference>^					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 13 - Declaration of variables					
1	Variable with elementary data type	✓	✓	✓	✓	
2	Variable with user-defined data type	✓	✓	✓	✓	
3	Array	✓	✓	✓	✓	
4	Reference					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 14 - Initialization of variables					
1	Initialization of a variable with elementary data type	✓	✓	✓	✓	
2	Initialization of a variable with user-defined data typ	✓	✓	✓	✓	
3	Array	✓	✓	✓	✓	
4	Declaration and initialization of constants					
5	Initialization using constant expressions	✓	✓	✓	✓	
6	Initialization of a reference					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 15 - Variable-length ARRAY variables					
1	Declaration using *					
2a	Graphical representation					
2b	Graphical representation					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 16 - Directly represented variables					
1	Input location	✓	✓	✓	✓	

2	Output location	✓	✓	✓	✓	
3	Memory location					
4a	Single bit size	✓	✓	✓	✓	
4b	Single bit size (none)					
5	Byte	✓	✓	✓	✓	
6	Word	✓	✓	✓	✓	
7	Dword	✓	✓	✓	✓	
8	Lword					
9	Simple addressing					
10	Hierarchical addressing using “.“	✓	✓	✓	✓	
11	Partly specified variables using asterisk ”*”					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 17 – Partial access of ANY_BIT variables					
1a	BYTE – bit					
1b	WORD – bit					
1c	DWORD – bit					
1d	LWORD – bit					
2a	WORD – byte					
2b	DWORD – byte					
2c	LWORD – byte					
3a	DWORD – word					
3b	LWORD – word					
4	LWORD - dword					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 18 – Execution control graphically using EN and ENO					
1	Usage without EN and ENO	✓	✓			
2	Usage of EN only					
3	Usage of ENO only					
4	Usage of EN and ENO	✓	✓			

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 19 – Function declaration					
1a	Without result	✓	✓	✓	✓	
1b	With result	✓	✓	✓	✓	
2a	Inputs	✓	✓	✓	✓	
2b	Outputs	✓	✓	✓	✓	
2c	In-outs	✓	✓	✓	✓	
2d,e	Temp variables	✓	✓	✓	✓	
2f	External variables	✓	✓	✓	✓	Implicit
2g	External constants	✓	✓	✓	✓	Implicit
3a	Initialization of inputs					
3b	Initialization of outputs					
3c	Initialization of temp variables	✓	✓	✓	✓	
--	EN/ENO input and output					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 20 – Function call					
1a	Complete formal call (textual only)	✓	✓	✓	✓	
1b	Incomplete formal call (textual only)					

2	Non-formal call (textual only)	✓	✓	✓	✓	
3	Function without function result	✓	✓	✓	✓	
4	Graphical representation	✓	✓			
5	Usage of negated boolean input and output in graphical representation		✓			Input only
6	Graphical usage of VAR_IN_OUT					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 21 – Typed and overloaded functions					
1a	Overloaded function	✓	✓	✓	✓	
1b	Conversion of inputs	✓	✓	✓	✓	
2a	Typed functions					
2b	Conversion	✓	✓	✓	✓	ANY_TO_...

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 22 – Data type conversion function					
1a	Typed conversion	✓	✓	✓	✓	ANY_TO_...
1b	Overloaded conversion	✓	✓	✓	✓	ANY_TO_...
2a	“Old” overloaded truncation	✓	✓	✓	✓	
2b	Typed truncation input_TRUNC_output					
2c	Overloaded truncation TRUNC_output					
3a	Typed input_BCD_TO_output					
3b	Overloaded BCD_TO_output	✓	✓	✓	✓	DINT
4a	Typed input_TO_BCD_output					
4b	Overloaded input_TO_BCD_output	✓	✓	✓	✓	DINT

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 23 – Data type conversion of numeric data types					
1..90	…_TO_...	✓	✓	✓	✓	ANY_TO_

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 24 – Data type conversion of bit data types					
1..21	_TO_ (bit data)	✓	✓	✓	✓	ANY_TO_
22..28	_TO_ (CHAR/WCHAR)					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 25 – Data type conversion of bit and numeric types					
1..76	_TO_	✓	✓	✓	✓	ANY_TO_

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 26 - Data type conversion of date and time types					
1..12	_TO_					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 27 - Data type conversion of character types					
1..8	_TO_					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 28 – Numerical and arithmetic functions					

1	ABS	✓	✓	✓	✓	
2	SQRT	✓	✓	✓	✓	
3	LN	✓	✓	✓	✓	
4	LOG	✓	✓	✓	✓	
5	EXP	✓	✓	✓	✓	
6	SIN	✓	✓	✓	✓	
7	COS	✓	✓	✓	✓	
8	TAN	✓	✓	✓	✓	
9	ASIN	✓	✓	✓	✓	
10	ACOS	✓	✓	✓	✓	
11	ATAN	✓	✓	✓	✓	
12	ATAN2	✓	✓	✓	✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 29 – Arithmetic functions					
1	ADD	✓	✓	✓	✓	
2	MUL	✓	✓	✓	✓	
3	SUB	✓	✓	✓	✓	
4	DIV	✓	✓	✓	✓	
5	MOD	✓	✓	✓	✓	
6	EXPT	✓	✓	✓	✓	
7	MOVE	✓	✓	✓	✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 30 – Bit shift functions					
1	SHL	✓	✓	✓	✓	
2	SHR	✓	✓	✓	✓	
3	ROL	✓	✓	✓	✓	
4	ROR	✓	✓	✓	✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 31 – Bitwise Boolean functions					
1	AND	✓	✓	✓	✓	
2	OR	✓	✓	✓	✓	
3	XOR	✓	✓	✓	✓	
4	NOT	✓	✓	✓	✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 32 – Selection functions					
1	MOVE	✓	✓	✓	✓	
2	SEL	✓	✓	✓	✓	
3	MAX	✓	✓	✓	✓	
4	MIN	✓	✓	✓	✓	
5	LIMIT	✓	✓	✓	✓	
6	MUX	✓	✓	✓	✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 33 – Comparison functions					
1	GT	✓	✓	✓	✓	
2	GE	✓	✓	✓	✓	
3	EQ	✓	✓	✓	✓	
4	LE	✓	✓	✓	✓	

5	LT	✓	✓	✓	✓	
6	NE	✓	✓	✓	✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 34 – Character string functions					
1	LEN	✓	✓	✓	✓	
2	LEFT	✓	✓	✓	✓	
3	RIGHT	✓	✓	✓	✓	
4	MID	✓	✓	✓	✓	
5	CONCAT	✓	✓	✓	✓	
6	INSERT	✓	✓	✓	✓	
7	DELETE	✓	✓	✓	✓	
8	REPLACE	✓	✓	✓	✓	
9	FIND	✓	✓	✓	✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 35 – Numerical functions of time and duration data types					
1..3	ADD	✓	✓	✓	✓	1a only
4..9	SUB	✓	✓	✓	✓	4a only
10	MUL	✓	✓	✓	✓	10b only
11	DIV	✓	✓	✓	✓	11b only

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 36 – Additional functions of time data types CONCAT and SPLIT					
1..8						

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 37 – Function for endianess conversion					
1	TO_BIG_ENDIAN					
2	TO_LITTLE_ENDIAN					
3	BIG_ENDIAN_TO					
4	LITTLE_ENDIAN_TO					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 38 – Functions of enumerated data types					
1	SEL	✓	✓	✓	✓	
2	MUX	✓	✓	✓	✓	
3	EQ	✓	✓	✓	✓	
4	NE	✓	✓	✓	✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 39 – Validate functions					
1	IS_VALID					
2	IS_VALID_BCD					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 40 – Function block type declaration					
1	Declaration of function block type	✓	✓	✓	✓	
2a	Declaration of inputs	✓	✓	✓	✓	
2b	Declaration of outputs	✓	✓	✓	✓	

2c	Declaration of in-outs	✓	✓	✓	✓	
2d	Declaration of temporary variables	✓	✓	✓	✓	
2e	Declaration of static variables	✓	✓	✓	✓	Implicit
2f,g	Declaration of external variables	✓	✓	✓	✓	Implicit
3a..d	Initialization	✓	✓	✓	✓	
4,5	RETAIN, NON_RETAIN					
6a,b	Textual declaration of edge inputs					
7	Graphical declaration of edge input					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 41 – Function block instance declaration					
1	Declaration of FB instance(s)	✓	✓	✓	✓	
2	Declaration of FB instance with initialization of its variables					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 42 – Function block call					
1	Complete formal call (textual only)			✓	✓	
2	Incomplete formal call (textual only)			✓	✓	
3	Graphical call	✓	✓			
4	Graphical call with negated boolean input and output		✓			Input only
5a,b	Graphical call with usage of VAR_IN_OUT					
6a	Textual call with separate assignment of input				✓	
6b	Graphical call with separate assignment of input	✓	✓			
7	Textual output read after FB call			✓	✓	
8a,b	Textual output assigned in FB call					
9a	Textual call with function block instance name as input			✓	✓	
9b	Graphical call with function block instance name as input	✓	✓			
10a	Textual call with function block instance name as VAR_IN_OUT			✓	✓	
10b	Graphical call with function block instance name as VAR_IN_OUT	✓	✓			
11a	Textual call with function block instance name as external variable			✓	✓	
11b	Graphical call with function block instance name as external variable	✓	✓			

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 43 – Standard bistable function blocks					
1a	SR (S1,R,Q1)					
1b	SR (SET1, RESET, Q1)	✓	✓	✓	✓	
2a	RS(S, R1, Q1)					
2b	RS(SET,RESET1, Q1)	✓	✓	✓	✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 44 – Standard edge detection function blocks					
1	R_TRIG	✓	✓	✓	✓	
2	F_TRIG	✓	✓	✓	✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 45 – Standard counter function blocks					
1	CTU	✓	✓	✓	✓	DINT
2	CTD	✓	✓	✓	✓	DINT

3	CTUD	✓	✓	✓	✓	DINT

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 46 – Standard timer function blocks					
1a,b	TP	✓	✓	✓	✓	TIME
1c	TP (LTIME)					
1a,b	TON	✓	✓	✓	✓	TIME
1c	TON (LTIME)					
1a,b	TOF	✓	✓	✓	✓	TIME
1c	TOF (LTIME)					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 47 – Program declaration					
1	Declaration of a program	✓	✓	✓	✓	
2a	Declaration of inputs					
2b	Declaration of outputs					
2c	Declaration of in-outs					
2d	Declaration of temporary variables	✓	✓	✓	✓	
2e	Declaration of static variables	✓	✓	✓	✓	Implicit
2f	Declaration of external variables	✓	✓	✓	✓	
2g	Declaration of external variables	✓	✓	✓	✓	
3a	Initialization of inputs					
3b	Initialization of outputs					
3c	Initialization of static variables	✓	✓	✓	✓	
3d	Initialization of temporary variables	✓	✓	✓	✓	
4a	Declaration of RETAIN qualifier on input variables					
4b	Declaration of RETAIN qualifier on output variables					
4c	Declaration of NON_RETAIN qualifier on input variables					
4d	Declaration of NON_RETAIN qualifier on output variables					
4e	Declaration of RETAIN qualifier on static variables	✓	✓	✓	✓	
4f	Declaration of NON_RETAIN qualifier on static variables	✓	✓	✓	✓	
5a	Declaration of RETAIN qualifier on local FB instances					
5b	Declaration of NON_RETAIN qualifier on local FB instances					
6a	Textual declaration of - rising edge inputs					
6b	Textual declaration of - falling edge inputs					
7a	Graphical declaration of - rising edge inputs					
7b	Graphical declaration of - falling edge inputs					
8a	VAR_GLOBAL...END_VAR declaration within a PROGRAM					
8b	VAR_GLOBAL CONSTANT declarations within PROGRAM					
9	VAR_ACCESS...END_VAR declaration within a PROGRAM					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 48 – Class					

1..11	…					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 49 – Class instance declaration					
1..2	…					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 50 - Textual call of methods – Formal and non-formal parameter list					
1..2	…					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 51 – Interface					
1..4	…					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 52 – Assignment attempt					
1..2	…					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 53 – Object oriented function block					
1..12	…					

Feat	*Table / Description*				*SFC*	*Notes*
	Table 54 – SFC step					
1a	Step – graphical form with directed links				✓	
1b	Initial step – graphical form with directed link				✓	
2a	Step – textual form without directed links					
2b	Initial step – textual form without directed links					
3a	Step flag – general form ***.X				✓	
3b	Step flag – direct connection of Boolean variable				✓	
4	Step elapsed time – general form ***.T				✓	

Feat	*Table / Description*				*SFC*	*Notes*
	Table 55 – SFC transition and transition condition					
1	Transition condition physically or logically adjacent to the transition using ST language				✓	
2	Transition condition physically or logically adjacent to the transition using LD language				✓	
3	Transition condition physically or logically adjacent to the transition using FBD language					
4	Use of connector					
5	Transition condition: Using LD language				✓	
6	Transition condition: Using FBD language					
7,8	Textual equivalent of feature 1 using ST or IL language					
9	Use of transition name					
10	Transition condition using LD language				✓	
11	Transition condition using FBD language					
12	Transition condition using IL language				✓	
13	Transition condition using ST language				✓	

Feat	*Table / Description*				*SFC*	*Notes*
	Table 56 – SFC declaration of actions					
1	Any Boolean variable declare				✓	
2l	Graphical declaration in LD language				✓	
2s	Inclusion of SFC elements in action				✓	
2f	Graphical declaration in FBD language				✓	
3s	Textual declaration in ST language				✓	
3i	Textual declaration in IL language				✓	

Feat	*Table / Description*				*SFC*	*Notes*
	Table 57 – Step/action association					
1	Action block physically or logically adjacent to the step				✓	
2	Concatenated action blocks physically or logically adjacent to the step				✓	
3	Textual step body					
4	Action block "d" field					

Feat	*Table / Description*				*SFC*	*Notes*
	Table 58 – Action block					
1	"a": Qualifier as per 6.7.4.5				✓	
2	"b": Action name				✓	
3	"c": Boolean "indicator" variables (deprecated)					
4i	IL language				✓	
4s	ST language				✓	
4l	LD language				✓	
4f	FBD language				✓	
5l	Use of action blocks LD					
5f	Use of action blocks in FBD					

Feat	*Table / Description*				*SFC*	*Notes*
	Table 59 – Action qualifiers					
1	None				✓	
2	N				✓	
3	R				✓	
4	S				✓	
5	L					
6	D					
7	P				✓	
8	SD					
9	DS					
10	SL					
11	P1				✓	
12	P0				✓	

Feat	*Table / Description*				*SFC*	*Notes*
	Table 60 – Action control features					
1	With final scan				✓	
2	Without final scan					

Feat	*Table / Description*				*SFC*	*Notes*
	Table 61 – Sequence evolution – graphical					

1	Single sequence				✓	
2a	Divergence of sequence with left to right priority				✓	
2b	Divergence of sequence with numbered branches					
2c	Divergence of sequence with mutual exclusion				✓	
3	Convergence of sequence				✓	
4a	Simultaneous divergence after a single transition				✓	
4b	Simultaneous divergence after conversion				✓	
4c	Simultaneous convergence before one transition				✓	
4d	Simultaneous convergence before a sequence selection				✓	
5	Sequence skip				✓	
6	Sequence loop				✓	
7	Directional arrows				✓	(free form)

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 62 – Configuration and resource declaration					
1..13						GUI only

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 63 - Task					
1..5						GUI only

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 64 - Namespace					
1..5						

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 65 – Nested namespace declaration options					
1..3						

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 66 – Namespace directive USING					
1..3						

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 67 – Parenthesized expression for IL language					
1	Parenthesized expression beginning with explicit operator:					
2	Parenthesized expression (short form)			✓		

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 68 – Instruction list operators					
1	LD			✓		
2	ST			✓		
3	S, R			✓		
4	AND			✓		
5	&			✓		
6	OR			✓		
7	XOR			✓		
8	NOT			✓		
9	ADD			✓		
10	SUB			✓		
11	MUL			✓		

12	DIV			✓		
13	MOD			✓		
14	GT			✓		
15	GE			✓		
16	EQ			✓		
17	NE			✓		
18	LE			✓		
19	LT			✓		
20	JMP			✓		
21	CAL			✓		
22	RET			✓		
23	)			✓		
24	ST?					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 69 – Calls for IL language					
1a	Function block call with non-formal parameter list			✓		
1b	Function block call with formal parameter list					
2	Function block call with load/store of standard input parameters			✓		
3a	Function call with formal parameter list					
3b	Function call with non-formal parameter list			✓		
4a	Method call with formal parameter list					
4b	Method call with non-formal parameter list					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 70 – Standard function block operators for IL language					
1	SR			✓		
2	RS			✓		
3	F/R_TRIG			✓		
4	CTU			✓		
5	CTD			✓		
6	CTUD			✓		
7	TP			✓		
8	TON			✓		
9	TOF			✓		

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 71 – Operators of the ST language					
1	Parentheses (expression)				✓	
2	Evaluation of result of function				✓	
3	Dereference				✓	
4	Negation				✓	
5	Unary plus				✓	
6	Complement				✓	
7	Exponentiation				✓	
8	Multiply				✓	
9	Divide				✓	
10	Modulo				✓	
11	Add				✓	

12	Subtract				✓	
13	Comparison				✓	
14	Equality				✓	
15	Inequality				✓	
16a	Boolean AND				✓	
16b	Boolean AND				✓	
17	Boolean Exclusive OR				✓	
18	Boolean OR OR				✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 72 – ST language statements					
1	Assignment				✓	
1a	Variable and expression of elementary data type				✓	
1b	Variables and expression of different elementary data types with implicit type conversion according Figure 11				✓	Int or real with no loss
1c	Variable and expression of user defined type				✓	
1d	Instances of function block type					
2a	Function call				✓	
2b	Function block call and FB output usage				✓	
2c	Method call					
3	RETURN				✓	
4	IF…				✓	
5	CASE…				✓	
6	FOR…				✓	
7	WHILE…				✓	
8	REPEAT…				✓	
9	CONTINUE					
10	EXIT				✓	
11	Empty statement				✓	

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 73 – Graphic execution control element					
1a	Unconditional jump FBD		✓			
1b	Unconditional jump LD	✓				
2a	Conditional jump FBD		✓			
2b	Conditional jump LD	✓				
3a	Conditional return LD	✓				
3b	Conditional return FBD		✓			
4	Unconditional return	✓	✓			

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 74 – Power rails and link elements					
1	Left power rail	✓	✓			
2	Right power rail	✓	✓			
3	Horizontal link	✓	✓			
4	Vertical link	✓	✓			

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 75 – Contacts					
1	Normally open	✓				
2	Normally closed	✓				

3	Positive transition	✓				
4	Negative transition	✓				
5a,b	Compare contact					

Feat	*Table / Description*	*LD*	*FBD*	*IL*	*ST*	*Notes*
	Table 76 - Coils					
1	Coil	✓				
2	Negated coil	✓				
3	Set coil	✓				
4	Reset coil	✓				
8	Positive transition-sensing coil	✓				
9	Negative transition-sensing coil	✓				